Université de Lille

Mémoire présenté pour obtenir

L'HABILITATION A DIRIGER DES RECHERCHES

Discipline: Mathématiques Appliquées et Applications des Mathématiques

Guillemette MAROT-BRIEND

préparé au laboratoire METRICS (Université de Lille, CHU Lille)
et dans l'équipe-projet Inria MODAL

Contributions méthodologiques en statistique pour l'analyse et l'intégration de données -omiques et cliniques

---

Soutenu le 8 janvier 2021 devant le jury composé de:

Présidente	Marie-Laure Martin-Magniette	Inrae
Rapportrice	Anne Gégout-Petit	Université de Lorraine
Rapporteur	Arthur Tenenhaus	Université Paris-Saclay
Examinateur	Christophe Biernacki	Inria
Examinatrice	Anne-Laure Boulesteix	Université de Munich, Allemagne
Garant	Alain Duhamel	Université de Lille, CHU Lille

après avis également favorable du rapporteur interne Jean-Sébastien Annicotte (Inserm) de l'école doctorale Biologie-Santé.

Remerciements

Tout d'abord, je tiens à remercier chaleureusement Anne Gégout-Petit et Arthur Tenenhaus qui, malgré leurs responsabilités et emplois du temps bien chargés, ont accepté de rapporter ce mémoire. Merci aussi à Marie-Laure Martin-Magniette, qui avait initialement accepté que je propose son nom à l'école doctorale comme rapportrice, avant que cette dernière ne garde que deux noms comme rapporteurs externes. Merci à Anne-Laure Boulesteix d'avoir accepté d'être examinatrice, quitte à prendre sur son temps de vacances pour que je puisse soutenir avant mon départ en congé maternité. C'est un honneur pour moi de vous avoir dans mon jury.

Merci à Christophe Biernacki, mon chef d'équipe-projet Inria MODAL, à la fois pour partager ce moment de science avec moi en participant à mon jury et pour avoir dirigé MODAL avec succès ces dix dernières années. Nous n'avons pas encore eu de collaboration vraiment directe mais cela ne saurait tarder, au regard de certains projets déposés. En tout cas, tu as su créer un environnement favorable pour la recherche et je t'en suis reconnaissante. Je ne peux citer MODAL sans remercier tout particulièrement Alain Celisse, mon voisin de bureau, qui m'a beaucoup apporté à la fois scientifiquement et humainement. C'est un réel plaisir de travailler avec toi et j'espère que nous aurons d'autres collaborations ensemble, même si ton nouveau poste bien mérité nous éloigne géographiquement. Merci aussi à Vincent Vandewalle, avec qui j'ai la chance de collaborer sur plusieurs projets. Merci de m'avoir relayée sur un encadrement d'ingénieure bilille pendant un de mes congés maternité et de co-encadrer avec moi Wilfried. Merci pour ton travail scientifique qui nous font progresser et merci aussi pour tes qualités humaines, qui permettent de relativiser telle ou telle situation. Je te souhaite tout le meilleur pour ta propre soutenance d'HDR et la suite. Merci aux autres collègues de MODAL passés et présents, avec qui j'ai passé de bons moments, en particulier les membres permanents: Sophie Dabo, Pascal Germain, Benjamin Guedj, Serge Iovleff, Julien Jacques, Cristian Preda, Hemant Tyagi sans oublier nos assistantes d'équipe Anne Rejl, Corinne Jamroz et Sandrine Meilen.

Merci à Alain Duhamel d'avoir accepté d'être le garant de ma demande d'habilitation à diriger des recherches. Merci pour ton travail de direction de l'EA2694 et merci de m'avoir soutenue dans mes choix de partager du temps entre la recherche propre en statistique et la direction de la plateforme bilille. Ton investissement pour la plateforme d'aide méthodologique du CHU a été un exemple pour moi et m'a motivée dans ce travail au service de la communauté en biologie-santé. C'est un réel plaisir de travailler ensemble sur le projet PreciNASH. Merci à Benoit Dervaux d'avoir porté le projet scientifique pour renouveler la labellisation de l'ULR2694 METRICS. Merci aux membres investis du conseil de laboratoire METRICS pour les échanges constructifs que nous avons en ces temps particuliers pour notre unité: Jean-Baptiste Beuscart, Emmanuel Chazard, Cyrielle Dumont, Antoine Lamer, Hervé Hubert, Sylvia Pelayo, Renaud Perichon et Damien Subtil. Merci à toutes les personnes localisées au CERIM pour les discussions informelles qui font regretter le temps du présentiel en cette période de confinement, en particulier à certaines personnes que je n'ai pas encore eu l'occasion de citer mais qui rendent cet environnement de recherche et d'enseignement très agréable: Pierre Balayé, Génia Babykina, Grégoire Ficheur, Michaël Genin, Sophie Quenton, Jean-Marie Renard, Mohamed-Salem Ahmed,

Mélanie Steffe et Julien Soula. J'espère avoir plus d'interactions scientifiques avec certains d'entre vous au sein d'une équipe-projet Inria orientée vers la santé.

Merci bien évidemment à toutes les personnes que j'ai encadrées, c'est grâce à vous et votre travail que je peux demander cette habilitation à diriger des recherches. Je commencerai par citer mes doctorants Quentin Grimonprez, Hélène Sarter et Wilfried Heyse mais je salue aussi l'investissement des ingénieurs ou post-doc que j'ai encadrés ou encadre encore: Samuel Blanck, Franck Bonardi, Perrine Boulenger, Maxime Brunin, Estelle Chatelain, Marie Fourcot, Audrey Hulot, Pierre Pericard, Morgane Pierre-Jean, Camille Ternynck, Dorothée Thuillier. Vous êtes trop nombreux pour commencer à raconter des anecdotes ici mais chacun de vous, avec sa personnalité, m'a fait grandir à la fois scientifiquement et humainement.

Merci aussi à mes co-auteurs que j'aurai l'occasion de citer plus loin dans ce mémoire et merci aux co-encadrants de thèse que je n'ai pas encore cités: Christophe Bauters, Corinne Gower et Florence Pinet. Co-encadrer une thèse interdisciplinaire nécessite des compétences complémentaires et c'est une réelle richesse de travailler ensemble.

Merci à Hélène Touzet de m'avoir entraîné dans l'aventure de la plateforme bilille. Après mes voisins de bureau, tu es celle que j'ai le plus cotoyée professionnellement ces dernières années et j'ai beaucoup appris avec toi. Merci de m'avoir permis de vivre de belles aventures inter-disciplinaires, merci pour ton exigence et ta délicatesse dans le management et la communication. Je te souhaite le meilleur pour la suite de ta carrière en dehors de bilille, merci encore pour tout ce que tu as apporté. Merci aussi aux ingénieurs bilille que je n'ai pas cités et les nombreux collaborateurs rencontrés dans ce cadre, que je ne citerai pas non plus, la liste étant trop longue.

Merci aux étudiants qui interagissent en cours et qui renouvellent mon goût pour l'enseignement et le transfert de connaissances, ainsi que tous les collègues avec qui je partage des enseignements et que je n'ai pas encore cités.

Enfin, ma plus grande gratitude va à ma famille, tout d'abord à mon mari Cyril, mes enfants Maximilien, Théophane, Colomban et petit bébé qui devrait poindre le bout de son nez d'ici quelques semaines. C'est vous qui me rappelez chaque jour que la famille est plus importante que le travail. Cyril, merci pour ton amour et merci de m'avoir soutenue tout au long de ces années pour que je puisse m'épanouir dans mon travail même si cela représente régulièrement pour toi de gros sacrifices. Maximilien, Théophane et Colomban, merci pour votre énergie débordante, votre cœur d'enfant et la joie que vous nous apportez. Merci aussi à mes parents, soeur jumelle, belle famille, neveux et nièces, grands-parents et cousins pour les plaisirs partagés en famille, ainsi qu'à toute la famille du Ciel de veiller ainsi sur nous.

Table des Matières

Guillemette Marot-Briend
Née le 1^{er} Septembre 1984
Mariée, 3 enfants (2015, 2017, 2019)

Adresses Professionnelles :

Université de Lille, ULR 2694 METRICS
Pôle recherche de la faculté de médecine, CERIM
1 place de Verdun
59045 Lille Cedex
☎ : +33 (0)3 20 62 68 32
guillemette.marot@univ-lille.fr

Inria MODAL
Parc scientifique de la Haute Borne
40 avenue Halley, Bat. A
59650 Villeneuve d'Ascq
+33 (0)3 59 57 79 77

ACTUELLEMENT :

Depuis sept. 2010	Maître de Conférences, section 26 (mathématiques appliquées et applications des mathématiques) Univ. Lille, CHU Lille, ULR 2694 METRICS & Inria, MODAL
Depuis nov. 2015	Co-responsable de la plateforme de bioinformatique et bioanalyse de Lille *bilille*

Modélisation statistique pour l'analyse de données issues d'expériences à haut débit (génomique, transcriptomique, protéomique, métabolomique, ...)

FORMATION ET EXPÉRIENCES :

2009 - 2010	**Post-doc** INRIA sous la direction de Franck Picard Laboratoire de Biométrie et Biologie Evolutive, Equipe Baobab Université Claude Bernard Lyon 1 - UMR 5558

Classification de courbes et identification de marqueurs moléculaires pour le cancer.

2006 - 2009	**Doctorat** spécialité **mathématiques appliquées**, Agro Paris Tech, école doctorale ABIES (ED 0435)
Co-encadrement	F. Jaffrézic, J.-L. Foulley : INRA Jouy-en-Josas - UMR 1313 Génétique Animale et Biologie Intégrative C.-D. Mayer : Biomathematics and Statistics Scotland (BioSS) - Rowett Institute, Aberdeen (Ecosse)

Modélisation statistique pour la recherche de gènes différentiellement exprimés : modèles de variance-covariance, analyse séquentielle et méta-analyse.

2004-2006	**Ingénieur ENSAI** (Ecole Nationale de la Statistique et de l'Analyse de l'Information, Rennes) après concours grandes écoles MP (Mathématiques Physique). Spécialité " sciences de la vie " en dernière année d'école.

DISTINCTIONS

Depuis 2018	Prime d'Encadrement Doctoral et de Recherche (PEDR) versée par Univ. Lille
2010-2015	Prime d'excellence Scientifique (PES) liée à la Chaire d'Excellence Univ. Lille 2/Inria
2010	Prix de thèse de la Société Française de Biométrie (thèse soutenue le 9 sept. 2009)

ACTIVITÉS D'ENSEIGNEMENT

Université de Lille, Faculté de médecine, depuis 2010

— probabilités, statistique descriptive, statistique inférentielle (Exercices Dirigés Première Année commune aux Etudes de Santé)

— préparation à la certification C2I (certificat informatique et internet) (Travaux Pratiques Deuxième Année des études médicales)

— variables aléatoires, lois usuelles, statistique descriptive, estimation (cours master 1 Biologie Santé)

— algèbre linéaire, analyse en composantes principales, classification, régression logistique (cours et travaux pratiques master 1 Biologie Santé)

— statistique inférentielle, régression linéaire, analyse de la variance, introduction à R (cours et travaux pratiques master 1 Biologie Santé option médecine/sciences)

— analyse de données avec le logiciel R (formation continue et formation doctorale)

— analyses RNA-Seq (formation continue)

— analyse statistique de données -omiques (formation doctorale)

— analyse statistique de données -omiques (Diplôme Universitaire Intelligence Artificielle en santé)

Université de Lille, Faculté des Sciences et Technologies, 2020

— analyse statistique de données -omiques (master Ingénierie Statistique et Numérique)

— régression linéaire, tests multiples, analyse différentielle de données transcriptomiques (master Bioinformatique)

Ecole Polytech'Lille, 2016, 2018, 2020

— classification supervisée (cours et travaux pratiques parcours Génie Informatique et Statistique 4ème année)

Université Claude Bernard Lyon 1 et INSA Lyon, 2009-2010

— mathématiques pour les sciences de la vie (travaux tutorés en Licence 1 Biologie UCBL)

— bioinformatique (travaux pratiques en 4ème année de parcours bioinformatique et modélisation et 5ème année de parcours biochimie biotechnologies INSA)

IUT Paris Descartes (Paris V) Département STID, 2007-2009

— statistique descriptive (travaux dirigés 1ère année)

— séries chronologiques (travaux pratiques 1ère année)

ACTIVITÉS DE RECHERCHE

Mon domaine de recherche est la statistique pour la bioinformatique. Plus précisément, ma recherche concerne :

— la modélisation de la variance dans des analyses différentielles (puces à ADN, RNA-seq)

— la détection de ruptures dans des données de génotypage

— la classification de profils -omiques (génomiques, transcriptomiques, protéomiques)

— l'intégration de données -omiques et cliniques dans un contexte de construction de score

Les méthodes statistiques étudiées pour ces questions sont :

— les approches bayésiennes empiriques

— les modèles mixtes

— les méthodes à noyaux

— les régressions pénalisées

FINANCEMENTS OBTENUS

Contrats industriels

2019	Contrat avec Bonduelle : 6212 € (salaire et fonctionnement)
2017	Contrat avec Florimond Desprez : 24 745 € (salaire et fonctionnement)
2015	Contrat avec Genoscreen : 5000 € (fonctionnement)

Participation à des projets de recherche académiques

En tant que porteur de projet :

2015-2016	projet SIRIC (site de recherche intégrée sur le cancer) MPAGenomics2 : 23 210 € (salaire et fonctionnement)
2012 - 2014	Action de développement technologique Inria MPAGenomics : 93 548 € (salaire)
2010 - 2015	Chaire d'excellence Lille 2 - Inria 75 000 € (salaire et fonctionnement)

En tant que participante :

2020 - 2025	projet européen FAIR (porteur : JC Sirard)
2018 - 2021	projet ANR TheraSCUD2022 (porteur : P. Gosset)
2016 - 2021	projet RHU PreciNASH (porteur : F. Pattou)
2014 - 2017	projet ANR COMeBACK (porteur : C. Gower)
2013 - 2016	bourse de thèse Inria - Direction Générale de l'Armement
2015	projet PEPS BeFast (porteur : A. Celisse)
2013	réseau INRA MIA "segmentation" (porteur : P. Neuvial)

ACTIVITÉS D'ENCADREMENT

Doctorants

— Wilfried Heyse (depuis octobre 2019, co-encadrement avec C. Bauters et V. Vandewalle) : Prise en compte de la structure temporelle dans l'analyse statistique de données protéomiques à haut débit.

— Hélène Sarter (depuis octobre 2016, co-encadrement avec C. Gower) : outils statistiques pour la sélection de variables et l'intégration de données cliniques et -omiques : développement et application au registre EPIMAD

— Quentin Grimonprez (2013-2016, co-encadrement avec A. Celisse) : sélection de groupes de variables corrélées en grande dimension

Stagiaires de master 2

— Wilfried Heyse (2019, 6 mois) : analyse de données protéomiques à haut débit

— Dorothée Thuillier (2011, 6 mois) : détection de pics dans des données de ChIP-on-chip

Ingénieurs jeunes diplômés non permanents

— Marie Fourcot (depuis juin 2018) : analyse de données pour des projets de la plateforme bilille : cytométrie, transcriptomique

— Franck Bonardi (depuis 2018) : analyse de données pour des projets de la plateforme bilille : criblage à haut contenu, transcriptomique

— Audrey Hulot (2016 - 2017, 1 an) : analyse de données pour des projets de la plateforme bilille : méta-analyse de données RNA-Seq, recherche de biomarqueurs

— Quentin Grimonprez (2012-2013, 1 an) : implémentation d'une méthode de sélection de variables en classification supervisée et application aux données de génotypage.

— Morgane Pierre-Jean (2012-2013, 10 mois) : détection de ruptures à partir de méthodes à noyaux pour des données de génotypage.

— Perrine Boulenger (2013, 6 mois) : détection de pics dans des données de ChIP-seq.

Ingénieurs non permanents expérimentés ou post-doc

— Estelle Chatelain (depuis jan. 2020) : classification non supervisée de données cliniques, analyse de données transcriptomiques.

— Pierre Péricard (depuis nov. 2018) : analyse de données de projets de la plateforme bilille (ChIP-Seq et RNA-Seq), interfaçage du package R mixOmics avec Galaxy

— Maxime Brunin (depuis oct. 2017) : analyse de données de projets de la plateforme bilille : recherche de biomarqueurs, classification croisée, intégration de données -omiques.

— Camille Ternynck (depuis oct. 2017) : construction d'un score avec des marqueurs cliniques et biologiques pour la prédiction de NASH (inflammation du foie), intégration de données continues gaussiennes et de données de comptage non gaussiennes.

— Samuel Blanck (2013-2016) :analyse de données d'expression ou de génotypage et interfaçage de la pipeline MPAgenomics avec Galaxy

ANIMATION SCIENTIFIQUE

Organisation de congrès nationaux ou internationaux

— Co-responsable du comité d'organisation des "Journées Ouvertes en Biologie, Informatique et Mathématiques" (JOBIM 2017) https://project.inria.fr/jobim2017/fr/

— Membre du comité d'organisation local des journées "Statistical Methods for Post-Genomic Data" (SMPGD 2016) http://math.univ-lille1.fr/~celisse/SMPGD/

Organisation de journées thématiques

— Innovative bioinformatics for single-cell resolution data (2019)
https://wikis.univ-lille.fr/bilille/singlecell2019

— Analysing data from phenotypic screening (2018)
https://wikis.univ-lille.fr/bilille/phenotypicscreening_2018

— Deuxième journée lilloise de proba-stat (2018) http://cerim.univ-lille2.fr/seminaires-et-evenement
2eme-journee-lilloise-de-proba-stat.html

— Bioinformatics for third generation sequencing (2015)
https://wikis.univ-lille.fr/bilille/animation

— Autour de la Biologie Intégrative (2014)
https://wikis.univ-lille.fr/bilille/animation

— La protéomique à l'ère du tout omique (2014)
https://wikis.univ-lille.fr/bilille/animation

— Bio-informatique structurale : protéines et interactions (2013)
https://wikis.univ-lille.fr/bilille/animation

— Analyse bioinformatique des données de métagénomique (2012) https://wikis.univ-lille.
fr/bilille/animation

— Analyse bioinformatique des données NGS (2011) https://wikis.univ-lille.fr/bilille/
animation

— Fouille de texte pour la biologie (2011)
https://wikis.univ-lille.fr/bilille/animation

COLLECTIVITÉ

Responsabilités et commissions

— Responsable de la plateforme de bioinformatique et bioanalyse lilloise *bilille*, labellisée par les Programmes Investissement d'Avenir (PIA 2) Infrastructures en Biologie-Santé "Institut Français de Bioinformatique" et "France Génomique" (depuis fin 2015)

— Responsable d'Unités d'Enseignement de statistique dans le master bioinformatique de la Faculté des Sciences et Technologies (2020) et le master Biologie-Santé de la Faculté de Médecine (depuis 2013)

— Membre nommée du conseil de laboratoire de l'ULR 2694 METRICS (depuis 2019)
— Membre nommée du groupe de travail recherche de la faculté de médecine, créé pour la réflexion autour de la potentielle fusion des facultés de santé (2019)
— Membre élue de la commission recherche de l'Université de Lille Droit et Santé (2016-2017)
— Membre nommée de la commission des utilisateurs des moyens informatiques Inria (2014-2015)
— Membre élue du comité de centre Inria (2011-2013)

OUTILS LOGICIELS

Packages R :
Les packages disponibles sur le site officiel de R, le CRAN, sont les packages les plus mûrs (hors package orphelin). La forge R permet une disponibilité publique d'un package R en cours de maturation et assure l'intégration continue. Un package orphelin est un package qui a été accepté un jour sur le CRAN mais dont la maintenance n'est plus assurée.

MLGL : Multi-Layer Group-Lasso, sélection de groupes de variables corrélées en grande dimension
Disponibilité CRAN `https://cran.r-project.org/web/packages/MLGL/index.html`
Publication associée : Grimonprez et al., en révision
Contribution : encadrement de Q. Grimonprez (principal contributeur du package) et tests

KernSeg Algorithmes de détection de ruptures permettant de gérer des signaux de grande taille, à l'aide de méthodes à noyaux.
Disponibilité : Forge R `https://r-forge.r-project.org/projects/kernseg/`
Publication associée : Celisse et al., Computational Statistics and Data Analysis 2018
Contribution : tests au travers de simulations pour l'article (contributeur principal du package : G. Rigaill)

metaMA Méta-analyse de données de puces à ADN
Disponibilité : CRAN `https://cran.r-project.org/web/packages/metaMA/index.html`
Publication associée : Marot et al., Bioinformatics 2009
Vignette : Marot and Bruyère (2015) Using metaMA for differential gene expression analysis from multiple studies
Contribution : développement passé (en thèse), création en 2015 d'un tutoriel publié sous forme de vignette dans la dernière version du package, maintenance minimum assurée.

metaRNASeq Méta-analyse de données RNASeq
Disponibilité : CRAN `https://cran.r-project.org/web/packages/metaRNASeq/index.html`
Publication associée : Rau et al., BMC Bioinformatics 2014
Vignette : Marot G., Jaffrézic F., Rau A. (2015) metaRNASeq : Differential meta-analysis of RNA-seq data
Contribution : développement et maintenance minimum assurée.

MPAgenomics Analyse multi-patients de données génomiques
Disponibilité : CRAN (orphelin) `https://cran.r-project.org/src/contrib/Archive/MPAgenomics/`
Publication associée : Grimonprez et al., BMC Bioinformatics 2014
Contribution : encadrement de Q. Grimonprez (principal contributeur du package) et tests

curvclust Classification de courbes
Disponibilité : CRAN (orphelin) `https://cran.r-project.org/src/contrib/Archive/curvclust/`
Publication associée : Giacofci et al., Biometrics 2013
Contribution : développement passé (contribution majeure en post-doc), maintenance laissée à F. Picard

SMVar Analyse différentielle de données de puces à ADN
Disponibilité : CRAN `https://cran.r-project.org/web/packages/SMVar/index.html`
Publication associée : Marot et al., Genetical Research 2007
Contribution : développement passé (thèse), maintenance minimum assurée

Instances Galaxy

Galaxy https://usegalaxy.org/ est une plateforme web d'analyse de données biomédicales. Galaxy offre une interface conviviale et un gestionnaire de workflows largement utilisé dans la communauté des biologistes. Pour faciliter l'utilisation des packages R précédents les plus utilisés par la plateforme de bioinformatique lilloise bilille, deux instances ont été réalisées :

MPAGenomics-Galaxy Analyse multi-patients de données génomiques avec Galaxy.
Disponibilité : Github https://github.com/sblanck/MPAgenomics
Publication associée : aucune (seulement dépôt à l'agence de protection des programmes)
Contribution : encadrement de S. Blanck (développeur de l'instance) et tests

SMAGEXP Méta-analyse de données transcriptomiques
Disponibilité : Github https://github.com/sblanck/smagexp
Publication associée : Blanck et Marot, Gigascience 2019
Contribution : encadrement de S. Blanck (développeur de l'instance) et tests

PUBLICATIONS :

ARTICLES :

Ma recherche étant inter-disciplinaire, j'ai séparé ci-dessous les publications dans des revues de biostatistique ou bioinformatique de celles dans des revues de biologie ou médecine. Les premières représentent un apport méthodologique en statistique pour la bioinformatique important et sont accompagnées d'un outil logiciel mentionné dans la section précédente. La deuxième catégorie reflète des travaux d'analyse de données, utilisant des outils existants ou implémentant des modèles simples ne justifiant pas une publication méthodologique.

— articles dans des revues de biostatistique ou bioinformatique

(: auteurs en ordre alphabétique)*

Grimonprez Q., Blanck S., Celisse A., **Marot G**. MLGL : an R package implementing correlated variable selection by hierarchical clustering and group-Lasso (en révision) Preprint HAL : https://hal.inria.fr/hal-01857242

Blanck, S. and **Marot, G.** (2019) SMAGEXP : a galaxy tool suite for transcriptomics data meta-analysis. *Gigascience* 8(2)

*Celisse, A., **Marot, G.**, Pierre-Jean M., Rigaill G. (2018) New efficient algorithms for multiple change-point detection with reproducing kernels. *Computational Statistics and Data Analysis*. 128, 200-220

Grimonprez Q., Celisse A., Cheok M., Figeac M., **Marot G.** (2014) MPAgenomics : An R package for multi-patients analysis of genomic markers. *BMC Bioinformatics* 15 :394

Rau A., **Marot G.**, Jaffrézic F. (2014) Differential meta-analysis of RNA-seq data from multiple studies. *BMC Bioinformatics* 15 :91

*Giacofci M., Lambert-Lacroix S., **Marot G.**, Picard F. (2013) Wavelet-based clustering for mixed-effects functional models in high dimension. *Biometrics* 69(1) :31-40

Dillies M.-A., Rau A. , Aubert J. , Hennequet-Antier C. , Jeanmougin M., Servant N., Keime C., **Marot G.**, Castel D., Estelle J., Guernec G., Jagla B., Jouneau L., Laloë D., Le Gall C., Schaëffer B., Le Crom S., and Jaffrézic F. (2012) A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis *Briefings in Bioinformatics* 14(6) :671-683.

Marot G., Foulley J.-L., Mayer C.-D. and Jaffrézic F. (2009) Moderated effect size and p-value combinations for microarray meta-analyses. *Bioinformatics.* 25(20) :2692-2699

Marot G. and Mayer C.-D. (2009) Sequential Analysis for Microarray Data Based on Sensitivity and Meta-Analysis. *Statistical Applications in Genetics and Molecular Biology* 83(1), Art. 3.

Marot G., Foulley J.-L. and Jaffrézic F. (2009) A structural mixed model to shrink covariance matrices for time-course differential gene expression studies. *Computational Statistics and Data Analysis* 53(5) :1630-1638

— articles dans des revues de médecine ou biologie

Cuvelliez M, Vandewalle V, Brunin M, Beseme O, Hulot A, de Groote P, Amouyel P, Bauters C, **Marot G**, Pinet F (2019) Circulating proteomic signature of early death in heart failure patients with reduced ejection fraction. *Scientific Reports* 9(1) :19202

Mogilenko DA, Haas JT, L'homme L, Fleury S, Quemener S, Levavasseur M, Becquart C, Wartelle J, Bogomolova A, Pineau P, Molendi-Coste O, Lancel S, Dehondt H, Gheeraert C, Melchior A, Dewas C, Artemii Nikitin A, Samuel Pic S, Rabhi N, Annicotte JS, Oyadomari S, Velasco-Hernandez T, Cammenga J, Foretz M, Viollet B, Vukovic M, Villacreces A, Kranc K, Carmeliet P, **Marot G**, Boulter A, Tavernier S, Berod L, Longhi MP, Paget C, Janssens S, Staumont-Sallé D, Aksoy E, Staels B, Dombrowicz D. (2019) Metabolic and innate immune cues merge into a specific inflammatory response via UPR. *Cell* 177(5) :1201-1216.

Bernardini M, Brossa A, Chinigo G, Guillaume P, Trimaglio G, Allart L, Hulot A, **Marot G**, Genova T, Joshi A, Mattot V, Fromont G, Munaron L, Bussolati B, Prevarskaya N, Pla AF , Gkika D (2019) Transient Receptor Potential Channel Expression Signature in Tumor-derived Endothelial Cells : Functional Roles in Prostate Cancer Angiogenesis. *Cancers* 11(7) :956.

Dhorne-Pollet S., Crisci E, Mach N, Renson P, Jaffrézic F, **Marot G** , Maroilley T. , Moroldo M. , Lecardonnel J. , Blanc F., Bertho N., Bourry O., and Giuffra E. (2019) The miRNA-targeted transcriptome of porcine alveolar macrophages upon infection with Porcine Reproductive and Respiratory Syndrome Virus. *Scientific Reports* 9(1) :3160.

Drullion C, **Marot* G**, Martin N, Deslé J, Saas L, Salazar-Cardozo C, Bouali F, Pourtier A, Abbadie C and Pluquet O. (2018) Pre-malignant transformation by senescence evasion is prevented by the PERK and ATF6alpha branches of the Unfolded Protein Response. *Cancer Letters*. 438 :187-196

Dubois-Chevalier J., Dubois V. , Dehondt H. , Mazrooei P. , Mazuy C. , Sérandour A., Gheeraert C. , Penderia G. , Baugé E. , Derudas B. , Hennuyer N. , Paumelle R. , **Marot G.** , Carroll J. , Lupien M., Staels B. , Lefebvre P., Eeckhoute J. (2017) The logic of transcriptional regulator recruitment architecture at cis-regulatory modules controlling liver functions. *Genome Research*. 27(6) :985-996

Herbaux, C., Bertrand, E., **Marot, G.**, Roumier, C., Poret, N., Soenen, V., Nibourel, O., Roche-Lestienne, C., Broucqsault, N., Galiègue-Zouitina, S., et al. (2016). BACH2 promotes indolent clinical presentation in Waldenström macroglobulinemia. *Oncotarget*. 8(34) :57451-57459.

Poulain S., Roumier C., Venet-Caillault A., Figeac M., Herbaux C., **Marot G.**, Doye E., Bertrand E., Geffroy S., Lepretre F., Nibourel O., Decambron A., Boyle E. M., Renneville A., Tricot S., Daudignon A., Quesnel B., Duthilleul P., Preudhomme C., Leleu X. (2016). Genomic landscape of CXCR4 mutations in Waldenstrom's Macroglobulinemia. *Clinical Cancer Research* 22, 1480-1488

Martin N., Salazar-Cardozo C., Vercamer C., Ott L., **Marot G.**, Slijepcevic P., Abbadie C., Pluquet O. (2014) Identification of a gene signature of a pre-transformation process by senescence evasion in normal human epidermal keratinocytes. *Molecular Cancer* 13 :51

Walker R., Gissot M., Huot L., Alayi T.D., Hot D., **Marot G.**, Schaeffer-Reiss C., Van Dorsselaer A., Kim K., Tomavo S. (2013) Toxoplasma transcription factor TgAP2XI-5 regulates the expression of genes involved in parasite virulence and host invasion. *Journal of Biological Chemistry* 288(43) :31127-38

Valour D., Hue I., Degrelle S.A., Déjean S., **Marot G.**, Dubois O., Germain G., Humblot P., Ponter A., Charpigny G., Grimard B. (2012) Pre- and Post-Partum Mild Underfeeding Influences Gene Expression in the Reproductive Tract of Cyclic Dairy Cows *Reproduction in Domestic Animals.* 48(3) :484-99

Guyonnet B., **Marot G.**, Dacheux J.-L., Lacoste A., Mercat M.-J., Schwob S., Jaffrézic F., Gatti J.-L. (2009) The adult boar testicular and epididymal transcriptomes. *BMC Genomics.* 10 :369-398

de Koning D-J, Jaffrézic F, Lund M S, Watson M, Channing C, Hulsegge I, Pool M, Buitenhuis B, Hedegaard J, Hornshøj H, Jiang L, Sørensen P, **Marot G**, Delmas C, Lê Cao K-A, SanCristobal M, Baron M D, Malinverni R, Stella A, Brunner R, Seyfert H-M, Jensen K, Mouzaki D, Waddington D, Jiménez-Marín A, Pérez Alegre M, Pérez E, Closset R, Detilleux J, Dovc P, Lavric M, Nie H, Janss L. The EADGENE Microarray Data Analysis Workshop. (2007) *Genetics Selection Evolution*, 39(6) :621-31

Watson M, Pérez Alegre M, Baron M D, Delmas C, Dovc P, Duval M, Foulley J-L, Garrido-Pavón J J, Hulsegge B, Jaffrézic F, Jiménez-Marín Á, Lavriè M, Le Cao K-A, **Marot G**, Mouzaki D, Pool M H, Robert-Granie C, San Cristobal M, Tosser-Klopp G, Waddington D, de Koning D-J. Analysis of a simulated microarray dataset : Comparison of methods for data normalization and detection of differential expression. (2007) *Genetics Selection Evolution*, 39(6) :669-83

Jaffrézic F, de Koning D-J, Boettcher P J, Bonnet A, Buitenhuis B, Closset R, Déjean S, Delmas C, Detilleux J C, Dovc P, Duval M, Foulley J-L, Hedegaard J, Hornshøj H, Hulsegge I B., Janss L, Jensen K, Jiang L, Lavric M, Lê Cao K-A, Lund M S, Malinverni, **Marot G**, Nie H, Petzl W, Pool M H, Robert-Granié C, SanCristobal M, van Schothorst E M., Schuberth H-J, Sørensen P, Stella A, Tosser-Klopp G, Waddington D, Watson M, Yang W, Zerbe H, Seyfert H-M. Analysis of the real EADGENE data set : Comparison of methods and guidelines for data normalization and selection of differentially expressed genes. (2007) *Genetics Selection Evolution*, 39(6) :633-50

Jaffrézic F., **Marot G.**, Degrelle S., Hue I. and Foulley, J.-L. (2007) A structural mixed model for variances in differential gene expression studies. *Genetical Research* 89(1) :19-25.

PRÉSENTATIONS ORALES (* orateur)

Vandewalle* V., Ternynck C., **Marot G.** (2019) Linking different kinds of omics data through a model-based clustering approach, *International Federation of Classification Societies*, Thessalonique, Grèce

Sarter* H, Savoye G, Turck D, Vasseur F, **Marot G.**, Pariente B, Colombel JF, Gower-Rousseau C, Fumery M (2019) Une combinaison de facteurs cliniques, sérologiques et génétiques prédit l'évolution vers une forme compliquée dans la maladie de Crohn à début pédiatrique : résultats d'une étude en population générale *Journées Francophones d'hépato-gastroentérologie & d'oncologie digestive*, Paris, France

Sarter* H, Savoye G, Turck D, Vasseur F, **Marot G.**, Pariente B, Sharat Singh, Colombel JF, Gower-Rousseau C, Fumery M (2018) A combination of clinical, serological and genetic factors predicts complicated disease course in paediatric-onset Crohn's disease : results from a population-based study. *13th Congress of ECCO, Inflammatory Bowel Diseases*, Vienne, Autriche

Marot* G. (2017) Biostatistique pour les données omiques, *1ère Journée Lilloise de Proba-stat*, Lille, France

Marot* G. (2016) Enjeux statistiques pour l'intégration de données omiques, *Journées scientifiques de la SFR Cancer*, Lille, France

Blanck* S., **Marot G.** (2016) Meta-analysis of transcriptomic data with Galaxy *Journées Ouvertes en Biologie, Informatique et Mathématiques*, Lyon, France

Grimonprez* Q., Celisse A., **Marot G.** (2016) Sélection de groupes de variables corrélées par classification ascendante hiérarchique et group-lasso *48e Journées de Statistique de la SFDS*, Montpellier, France

Grimonprez* Q., Celisse A., **Marot G.** (2015) Sélection de groupes de variables corrélées par classification ascendante hiérarchique et group-lasso *47e Journées de Statistique de la SFDS*, Lille, France

Grimonprez* Q., Celisse A., **Marot G.** (2014) Analyse multi-patients de données génomiques *46e Journées de Statistique de la SFDS*, Rennes, France

Blanck* S., **Marot G.** Analyse multi-patients de données SNP/CN avec MPAgenomics (2014) *Groupe de travail Statomique*, Paris, France

Marot* G. Que peut apporter la statistique dans le traitement de données génomiques ? (2014) *Institut Pasteur de Lille*, Lille, France

Marot* G., Jaffrézic F., Rau A. (2013) metaRNASeq : un package pour la méta-analyse de données RNA-seq *Rencontres R*, Lyon, France

Pierre-Jean M., **Marot G.**, Rigaill G.*, Celisse A. (2013) Change-point detection with kernel methods *GDR Statistique et Santé*, Paris, France

Pierre-Jean* M., **Marot G.**, Rigaill G., Celisse A. (2013) Change-point detection with kernel methods : application to DNA copy number signals *45e Journées de Statistique de la SFDS* , Toulouse, France

Marot* G. Detection de pics en ChiP-on chip et ChIP-seq (2013) *Groupe de travail Statomique*, Paris, France

Marot* G. Que peut apporter la statistique dans le traitement de données génomiques ? (2013) *Journées Oncolille*, Lille

Herbaux* C., Bertrand E., **Marot G.**, Broucqsault N., Boyle E., Guidez S., Demarquette H., Galiègue-Zouitina S., Roumier C., Tricot S., Poulain S., Leleu X. (2013) Identification de nouveaux gènes cibles pour évaluer et comprendre les différences d'agressivité clinique dans la maladie de Waldenström *Congrès de la société française d'hématologie*, Paris, France

Giacofci M.*, Lambert-Lacroix S., **Marot G.**, Picard F. (2011) Wavelet-based clustering for mixed-effects functional models. *International Biometric Society Channel Network conference*, Bordeaux, France

Marot* G. (2011) Modélisation statistique pour l'analyse de données de puces à ADN, Laboratoire Génétique et Evolution des Populations Végétales, Lille, France

Marot* G. (2011) Présentation de Bioconductor et de son utilisation sur les puces à ADN *séminaire du réseau régional d'ingénieurs en bioinformatique de Lille*, Lille, France

Marot* G., Foulley J.-L., Mayer C.D., Jaffrézic F. (2010) Modélisation statistique pour la recherche de gènes différentiellement exprimés : modèles de variance-covariance, analyse séquentielle et méta-analyse *8ème Journée Jeunes Chercheurs de la Société Française de Biométrie*, Paris, France

Marot* G., Foulley J.-L, Mayer C.-D., Jaffrézic F. (2009) metaMA : an R package implementing meta-analysis approaches for microarrays. *useR ! 2009*, Rennes, France.

Marot* G., Foulley J.-L, Mayer C.-D., Jaffrézic F. (2009) Microarray meta-analysis based on p-value or moderated effect size combinations. *Workshop on Statistical Methods for Post-Genomic Data*, Paris, France.

Marot* G., Jaffrézic F., Foulley J.-L., Mayer C.-D. (2008) Sequential analysis for microarray data based on sensitivity and meta-analysis. *XXIV International Biometric Conference*, Dublin, Irlande.

Marot G., Foulley J.-L., Jaffrézic* F. (2008) A structural mixed model to shrink covariance matrices for time-course differential gene expression studies. *XXIV International Biometric Conference*, Irlande.

Marot* G., Foulley J.-L., Jaffrézic F. (2008) Shrinkage of covariance matrices for time-course differential gene expression studies. *Workshop on Statistical Methods for Post-Genomic Data*, Rennes, France.

Marot* G., Mayer C.-D., Foulley J.-L, Jaffrézic F. (2008) Modélisation statistique pour les données d'expression de gènes. *X séminaire des thésards du département de génétique animale*, Toulouse, France.

Guyonnet* B., Dacheux J.-L., Jaffrézic F., Lacoste A., **Marot G.**, Mercat M.-J., Schwob S., Gatti J.-L. (2008) Le transcriptome épididymaire du verrat : étude de la régionalisation. *Journées Recherche Porcine*,

40, 99-104. Paris, France.

Degrelle* S., Hue I., Campion E., Jaffrézic F., **Marot G.**, Everts R., Ducroix-Crépy C., Vignon X., Heyman Y., Yang X., Lewin H., Renard J.-P. (2007) Embryonic and extraembryonic abnormalities in Day-18 cloned bovine conceptuses. *II International Meeting on Mammalian Embryogenomics*, Paris, France.

Jaffrézic* F., **Marot G.**, Degrelle S., Hue I., Foulley J.-L. (2007) Detection of differentially expressed genes : the importance of variance modelling in the test statistics. *II International Meeting on Mammalian Embryogenomics*, Paris, France.

Marot* G., Foulley J-L., Jaffrézic F. (2006) Variance model comparisons for differential gene expression. *EADGENE Data Analysis Workshop*, Tune, Danemark.

Marot G., Foulley J-L., Jaffrézic* F. (2006) Real data analysis with a structural model for variances for differential gene expression. *EADGENE Data Analysis Workshop*, Tune, Danemark.

POSTERS

Sarter H, Savoye G, Turck D, Vasseur F, **Marot G.**, Pariente B, Singh S, Colombel JF, Gower-Rousseau C, Fumery M (2018) A combination of clinical, serological and genetic factors predicts complicated disease course in paediatric-onset Crohn's disease : results from a population-based study. *Digestive Disease Week*, Washington, USA

Goulas E, **Marot G**, Hulot A, Day A, Chabi M, Aribat S, Neutelings G, Blervacq A-S, Rolando C, Bray F, Fliniaux O, Le Gall H, Gillet F, Domon J-M, Rayon C, Tokarski C, Pelloux J, Mesnard F, Lucau-Danila A, Hawkins S (2018). Towards a better understanding of cell wall dynamics in water-stressed flax by integrative networking. *ELB2018 "Exploring lignocellulosic biomass : challenges and opportunity for bioeconomy"*, Reims, France.

Sarter H, Gower-Rousseau C, **Marot G** (2017) Influence of SNP coding on the analysis of disease risk *Journées Ouvertes en Biologie, Informatique et Mathématiques*, Lille, France

Goulas E, **Marot G**, Hulot A, Day A, Chabi M, Aribat S, Neutelings G, Blervacq A-S, Rolando C, Bray F, Fliniaux O, Le Gall H, Gillet F, Domon J-M, Rayon C, Tokarski C, Pelloux J, Mesnard F, Lucau-Danila A, Hawkins S (2017). Intégration de données multi-omiques : l'exemple du stress hydrique chez le lin. *11es journées du Réseau Français des Parois*, Orléans, France.

Goulas E, **Marot G**, Hulot A, Day A, Chabi M, Aribat S, Neutelings G, Blervacq A-S, Rolando C, Bray F, Fliniaux O, Le Gall H, Gillet F, Domon J-M, Rayon C, Tokarski T, Pelloux J, Mesnard F, Lucau-Danila A, Hawkins S. (2017) The FLAxMixomics project : how to integrate multiple -omics data sets ? *Third general COST meeting COST FA 1306 - The quest for tolerant varieties - Phenotyping at plant and cellular level*, Oeiras, Portugal.

Grimonprez Q., Celisse A., **Marot G.** (2016) Variable selection by exploiting correlation *XXVIIIth International Biometric Conference*, Victoria, Canada

Goulas E, **Marot G**, Day A, Chabi M, Aribat S, Neutelings G, Blervacq AS, Rolando C, Bray F, Fliniaux O, Le Gall H, Gillet F, Domon JM, Rayon C, Tokarski C, Pelloux J, Mesnard F, Lucau-Danila A and Hawkins S (2016) Towards a better understanding of cell wall dynamics in water stressed flax with the help of mixOmics *The third WG2 meeting of COST action FA1306*, Bratislava, République slovaque.

Dhorne-Pollet S., Renson P., Mach N., Jaffrézic F., **Marot G.**, Maroilley T., Moroldo M., Lecardonnel J., Bourry O., Giuffra E. (2016) RNA Silencing? targeted transcriptome of porcine alveolar macrophages upon infection with Porcine Reproductive and Respiratory Syndrome viruses (PRRSV) of different virulence *International Society for Animal Genetics*, Dublin, Irlande.

Dhorne-Pollet S., Lecardonnel J., Jaffrézic F., Moroldo M., **Marot G.**, Giuffra E. (2014) Understanding the complex interaction between the pseudorabies virus (PrV) and its natural host (pig) using RIP-Chip

enrichment analysis *International Society for Animal Genetics*, Xi'an, Chine.

Grimonprez Q., Celisse A., **Marot G.** (2014) Analysis of genomic markers : Make it easy with the R package MPAgenomics *Statistical Methods for Post-Genomic Data*, Paris, France

Herbaux C., **Marot G.**, Bertrand E., Broucqsault N., Zouitna-Galiègue S, Roumier C., Poret N., Soenen V., Nibourel O., Roche-Lestienne C., Renneville A., Boyle E., Fouquet G., Tricot S., Preudhomme C., Quesnel B., Poulain S., Leleu X. (2012) B-Cell-Specific Transcription Factor BACH2 Involved in the Clinical Behavior Heterogeneity of Waldenström Macroglobulinemia, *54th ASH Annual Meeting and Exposition*, Atlanta, Etats-Unis.

Pierre F., Reboul A., Grenier-Boley B., **Marot G.**, Guedj M. , Blervaque R. , Hot D. , Pichon C. , Touzet H. , Pradel E., Sebbane F. (2011) Toward the identification and characterization of the Yersinia pestis RNome produced in vivo, *ASM (American Society for Microbiology) Conference on Regulating with RNA in bacteria*, San Juan, Porto Rico.

Marot G., Mayer C.-D., Foulley J.-L., Jaffrézic F. (2008) Modélisation statistique pour les données d'expression de gènes *Journées ABIES*, Paris, France.

Guyonnet B., Dacheux J.-L., Jaffrézic F., Lacoste A., **Marot G.**, Mercat M.-J., Schwob S., Gatti J.-L. (2008) The Adult Boar Testicular and epididymal transcriptome *Society for the Study of Reproduction* Kailua-Kona, USA

Mayer C.-D. and **Marot G.** (2007) A sequential approach to microarray analyses *IV Annual meeting of NuGO - The European Nutrigenomics Organisation*, Oslo, Norvège

Marot G., Foulley J.-L., Hue I., Mayer C.-D. Jaffrézic F. (2007) Modélisation statistique pour les données d'expression de gènes *IX séminaire des thésards du département de génétique animale*, Jouy en Josas, France

Guyonnet B., Dacheux J.-L., Jaffrézic F., Lacoste A., **Marot G.**, Mercat M.-J., Gatti J.-L. (2007) Recherche et identification de gènes différentiellement exprimés dans l'épididyme de verrat par une approche transcriptomique. *Journées Recherche Porcine*, 39, 297-298. Paris, France

Guyonnet B., Dacheux J.-L., Jaffrézic F., Lacoste A., **Marot G.**, Mercat M.-J., Schwob S., Gatti J.-L. (2007) Le transcriptome épididymaire chez le verrat : Etude de la Régionalisation. *Société Française de Biochimie et de Biologie Moléculaire*, Ile des Embiez, France.

Guyonnet B., Dacheux J.-L., Jaffrézic F., Lacoste A., **Marot G.**, Mercat M.-J., Gatti J.-L. (2006) Analysis of differentially expressed genes along the boar epididymis. *IV International workshop on epididymis*, Châtel-Guyon, France.

Chapter 1

De l'analyse de données -omiques à la construction de scores cliniques

1.1 Introduction

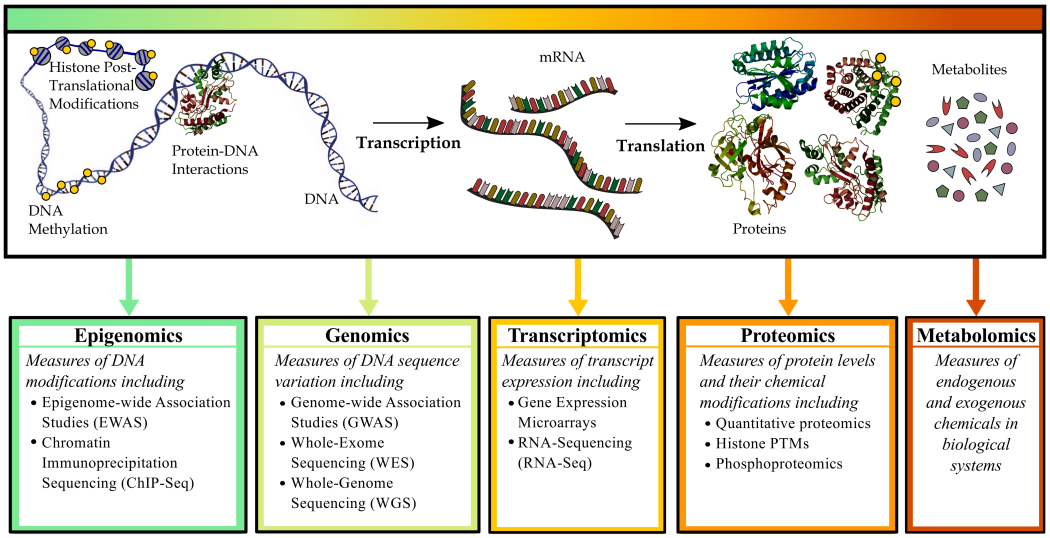
Derrière le terme "-omiques" se cachent beaucoup de réalités différentes, que ce soit sur le plan biologique ou le plan statistique. Dans ce mémoire, le terme "-omiques" désignera les données obtenues par expériences à haut débit : génomiques, transcriptomiques, protéomiques, épigénomiques, métabolomiques, ... Pour les non initiés, quelques lignes décriront ces principaux niveaux -omiques dans la section 1.1.1. Pour le statisticien, ces données partagent toutes un point commun, à savoir que le nombre de variables mesurées (p) est largement supérieur au nombre d'individus (n). Des exemples de transcriptomique sont souvent utilisés pour illustrer des problèmes dits de grande dimension ($p >> n$), où des mesures de puces à ADN ou de séquençage à haut débit permettent de mesurer l'expression de plusieurs milliers de gènes simultanément sur un nombre relativement restreint de patients. La figure 1.1 illustre la variété des technologies à haut débit et donne un aperçu de différentes applications biologiques, de l'épigénomique à la métabolomique. Certaines seront reprises dans les paragraphes suivants.

Derrière chacune de ces applications se posent des questions différentes et l'une des tâches les plus dures du biostatisticien confronté à une nouvelle technologie et à une nouvelle application est de traduire les questions biologiques de l'expérience en question mathématique.

1.1.1 Les données -omiques

Suite à la découverte de la structure moléculaire en double hélice de la molécule d'ADN par Watson et Crick, les chercheurs en biologie ont concentré leurs efforts autour de ce qui est appelé "dogme central de la biologie moléculaire", à savoir la relation entre l'ADN, matériel génétique et les protéines synthétisées dans une cellule. Dans ce dogme, l'ADN est transcrit en ARN messager (ou ARNm), lui-même traduit en protéine. Aujourd'hui, la complexité des mécanismes remet ce dogme et ses extensions en cause [Nau, 2003] et le terme dogme ne doit plus être interprété comme une vérité incontestable mais comme une théorie scientifique. Cette relation reste néanmoins très intéressante pour expliquer simplement l'intérêt des analyses des différents niveaux -omiques.

Figure 1.1: Données -omiques: quelques technologies et applications



Blanca Himes ©Himes Lab

Le niveau ADN est très souvent analysé dans les échantillons tumoraux. En effet, les cancers font suite à une succession d'accidents génétiques, comme l'explique le dossier pédagogique [Curie contributors, 2020]. Si la plus grande majorité des mutations reste sans effet, certaines peuvent avoir des conséquences sur l'ensemble de l'organisme. Les cancers, ainsi que leurs différentes formes, n'ayant pas tous les mêmes anomalies, il est intéressant d'identifier celles qui prédisent des formes agressives de cancer, afin de personnaliser les traitements, en fonction des bénéfices ou risques connus.

L'analyse du niveau transcriptomique permet de repérer quels gènes s'expriment dans quelles cellules. L'expression des gènes correspond à la quantité d'ARN messager (ARNm) mesurée. A ce niveau, il est intéressant de comparer l'expression du même gène dans différentes conditions (par exemple dans des cellules saines ou dans des cellules tumorales). L'analyse des différences de moyennes d'expression entre deux conditions est couramment appelée analyse différentielle et la liste de gènes différentiellement exprimés souvent appelée signature. S'il n'y a pas d'ARNm permettant la synthèse de telle ou telle protéine, cela donne des pistes d'innovation thérapeutique pour réparer des erreurs de mécanisme.

La protéomique est utilisée pour quantifier et caractériser fonctionnellement les protéines produites par l'expression d'un génome, les protéines assurant des fonctions très diverses dans l'organisme.

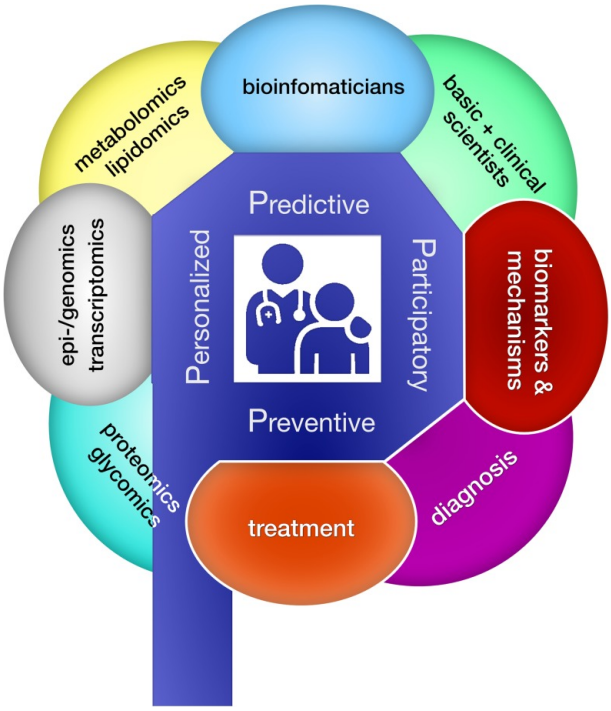
La métabolomique étudie les interactions entre les protéines et l'ensemble des métabolites (sucres, lipides, ...) d'une cellule.

L'épigénomique est l'étude des modifications qui interviennent dans la régulation des gènes. L'étude de ces modifications permet de comprendre pourquoi certains gènes sont activés ou non, comment l'environnement ou un régime alimentaire donné influencent la régulation des gènes. Par exemple, un phénomène de méthylation peut éteindre l'expression des gènes.

1.1.2 Intérêt des données -omiques pour le clinicien

Ce n'est pas toujours évident de savoir quel niveau -omique va être le plus intéressant pour tel ou tel clinicien. Si le niveau ADN est celui le plus analysé en première intention pour les cancers, le niveau transcriptomique reste un niveau dont le coût pour un nombre de sondes analysées élevé reste abordable et qui permet d'avoir une vue déjà intéressante de signatures moléculaires. La révolution technologique en protéomique permet maintenant de réaliser aussi des analyses à grande échelle, invitant de plus en plus de chercheurs à combiner des résultats de transcriptomique et de protéomique. Il est intéressant de remarquer que les plateformes qui génèrent les données -omiques sont en général spécialistes d'un seul niveau -omique alors que le clinicien aimerait des résultats clef en main pour faire de la médecine personnalisée, c'est-à-dire adapter à chaque patient le traitement qui lui est adapté. Ce concept de médecine personnalisée est l'un des 4P de la médecine de précision, décrit dans la figure 1.2.

Figure 1.2: Médecine 4P: prédictive, préventive, personnalisée et participative.



Source: [Hood et al., 2012]

Le terme 4P est apparu en 2012 dans le papier [Hood et al., 2012], reprenant les 4P de prédictive, préventive, personnalisée et participative. La médecine 4P suppose une révolution numérique de la médecine, la collecte et la mise en commun de données de nombreux patients pour pouvoir appréhender la complexité des maladies. Une approche intégrée est nécessaire, faisant appel à différentes compétences dont celles des bioinformaticiens et statisticiens. Il est à noter que le terme anglais "bioinformaticians" inclut les statisticiens spécialistes de l'analyse de données -omiques. L'identification et la compréhension des différentes formes de maladie permet alors de mieux stratifier les patients et de leur proposer des traitements les plus personnalisés possibles.

Comme mentionné dans [van Karnebeek et al., 2018], certaines technologies à haut débit sont plus mûres que d'autres pour une utilisation en clinique. Il n'en reste pas moins que ces données -omiques qui, jusqu'ici étaient étudiées en recherche fondamentale pour pré-sélectionner des facteurs d'intérêt, vont être de plus en plus intégrées dans la construction de scores, pour aider à la décision des cliniciens. Il est donc indispensable de bien les connaître, et de savoir comment les intégrer avec des données cliniques.

1.1.3 Exemples de technologies à haut débit

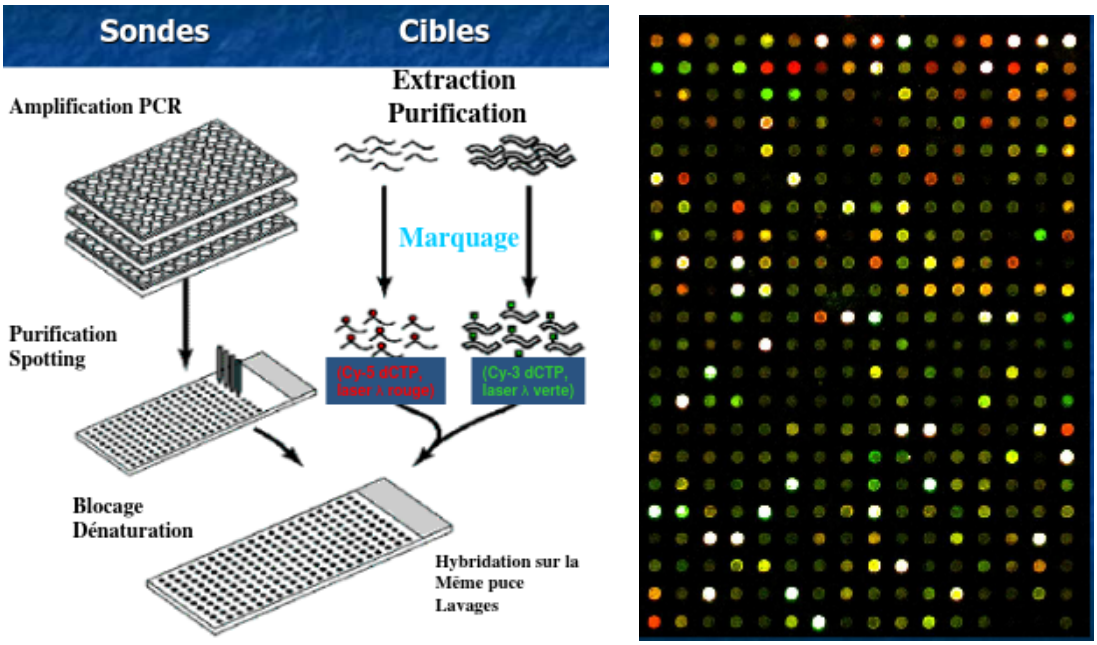
La variété des technologies a été illustrée dans la figure 1.1. Je ne présenterai dans ce paragraphe qu'une sélection de technologies, afin d'introduire les différents articles inclus dans ce mémoire. D'autres exemples seront présentés dans le dernier chapitre dédié aux projets de la plateforme de bioinformatique bilille.

Puces à ADN Le principe des puces à ADN repose sur une propriété de l'ADN et de l'ARN: l'hybridation (reconnaissance et interaction de deux séquences d'ADN ou d'ARN complémentaire. Le principe de ces puces est représenté figure 1.3. Les sondes sont des séquences qui ont été amplifiées (par la technique Polymerase Chain Reaction) puis déposées sur un support solide qui est la puce. La dénaturation permet d'obtenir des "ADN simple brin" qui captureront les cibles. Les cibles sont des ARNm marqués (par flurochrome, par radioactivité) et mis en contact avec les puces. Or, quand un brin d'ARN et un brin d'ADN sont complémentaires, ils se reconnaissent et interagissent (s'hybrident). L'hybridation terminée, la quantité de cibles hybridées est mesurée par scanner. Ces quantités étant proportionnelles à l'expression des gènes correspondants, les puces à ADN permettent ainsi la mesure de l'expression de plusieurs milliers de gènes simultanément. Les puces se différencient par la nature du support, le nombre et le type de sondes à utiliser, et le marquage des cibles. Pour des puces à deux couleurs, la quantité de cibles hybridées est mesurée en détectant la fluorescence émise par l'excitation des fluorophores. Le scanner génère une image en niveaux de gris pour chaque fluorophore. Ces deux images représentent l'intensité de fluorescence lue par le scanner et donnent le niveau d'expression des gènes dans les deux conditions expérimentales. Une image en fausses couleurs (cf. figure 1.3 à droite) sert souvent de représentation à ces deux images. Ces couleurs vont du vert, pour caractériser l'échantillon marqué au Cy3, au rouge pour l'échantillon marqué en Cy5. Le jaune indique que les cibles marquées en Cy5 et Cy3 se sont hybridées en proportion égale.

La technique des puces à ADN peut aussi être utilisée pour étudier des variations du nombre de copies d'ADN, par exemple entre des cellules tumorales et des cellules normales. Dans ce cas-là, ce ne sont pas des ARNm qui sont étudiés mais des segments d'ADN. On parle alors de puces d'hybridation génomique comparative (CGH arrays).

Séquençage à haut débit Le séquençage à haut-débit a pour principe de base la parallélisation de réactions permettant le séquençage de courtes lectures d'une librairie [Audebert et al., 2014]. Le séquençage repose sur plusieurs étapes successives comme illustré dans la figure 1.4: (1) la fragmentation de l'ADN et la ligation des fragments avec des adaptateurs universels, (2) l'amplification moléculaire (pour le séquençage 2ème génération, absente dans la troisième génération), (3) la lecture des fragments qui renvoient des séquences

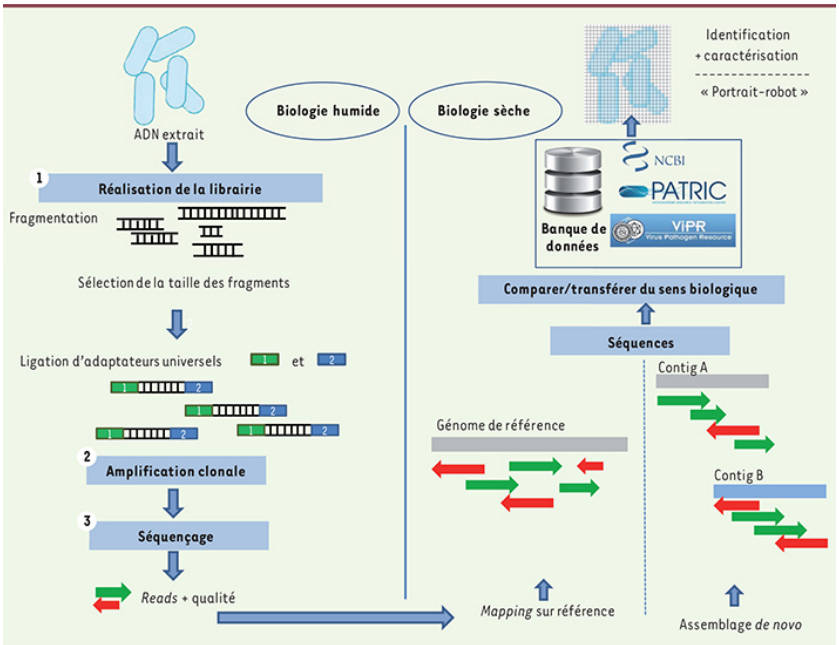
Figure 1.3: Principe des puces à ADN



A. Guillouze F Morel ©Société Française de Toxicologie, 2004

nucléotidiques courtes. Il y a ensuite une étape importante de bioinformatique pour établir une table de comptage qui donne le nombre de lectures à chaque position de la séquence.

Figure 1.4: Principe du séquençage à haut débit

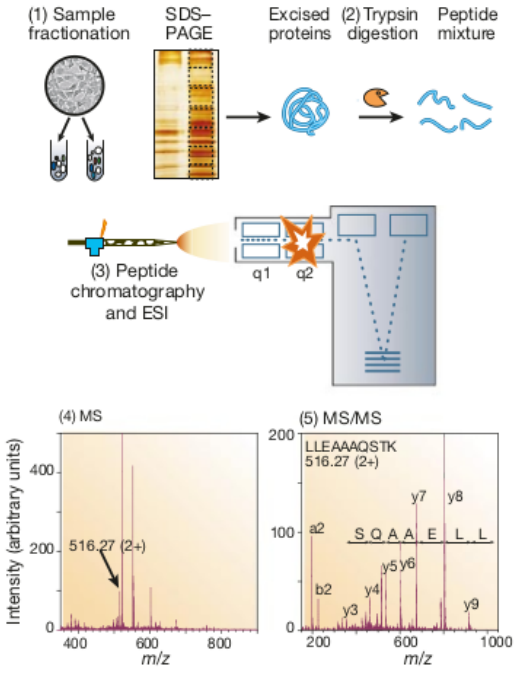


©Med Sci 2014; 30(12): 1144–1151

Comme représenté en figure 1.4, il existe deux grandes catégories d'algorithmes en bioinformatique pour obtenir des séquences plus longues que celle initialement lues: le mapping sur référence, qui consiste à repositionner les lectures sur un génome de référence, et l'assemblage de novo qui n'a pas besoin de référence. Développer des outils dans ces deux grandes catégories qui ne génèrent pas trop d'erreurs et sont rapides en temps de calcul représente un travail important pour les équipes de recherche en bioinformatique. Le travail des statisticiens commence souvent après cette étape, en partant de la table de comptage.

Spectrométrie de masse La spectrométrie de masse est une technique d'analyse physico-chimique permettant de détecter, d'identifier et quantifier des molécules d'intérêt, très utilisée en protéomique [Aebersold and Mann, 2003] et métabolomique [Dettmer et al., 2007]. Le principe d'une expérience de spectrométrie de masse en protéomique est illustré en figure 1.5.

Figure 1.5: Principe d'une expérience de spectrométrie de masse en protéomique



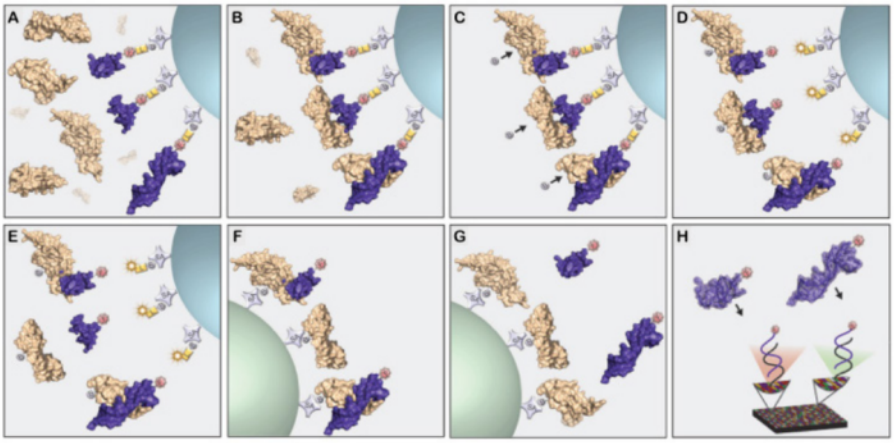
Source: [Aebersold and Mann, 2003]

Les protéines sont d'abord isolées (1) puis dégradées en peptides (2) pour faciliter l'étape d'identification par la masse. En effet, les protéines intactes sont difficilement identifiables par leur masse. Les peptides passent ensuite dans le spectromètre de masse, après une phase de chromatographie qui les sépare (3). Le spectromètre de masse ionise les peptides puis un analyseur sépare les ions en fonction de leur rapport masse/charge (4). Une expérience de spectrométrie de masse en tandem (couplant deux analyseurs) MS/MS permet de fragmenter les ions et les séparer (5).

L'utilisation de la spectrométrie de masse en protéomique permet l'analyse de plusieurs milliers de peptides simultanément. L'identification des protéines à partir des peptides nécessite des comparaisons avec des banques de séquences protéiques connues et reste une étape difficile, avec beaucoup de paramètres ou seuils à fixer manuellement. L'alignement des spectres est aussi une étape cruciale lorsqu'on veut comparer les mesures de plusieurs individus.

Technologie Somascan La technologie SomaScan est une technologie qui permet de doser des milliers de protéines en même temps. C'est une des plus innovantes en termes de haut débit. Le projet de thèse de Wilfried Heyse s'appuie sur un jeu de données mesurant la concentration de 5284 protéines simultanément. Cette technologie offre aussi l'avantage de ne nécessiter que 65 μL de plasma, ce qui est intéressant quand on s'intéresse à des patients gravement malades à qui on ne peut pas prélever trop de sang. La technologie SomaScan utilise des aptamers modifiés appelés Somamer (Slow Off-rate Modified Aptamer). Un aptamer est un oligonucléotide synthétique (ADN simple brin) capable de reconnaître une protéine de manière très spécifique (de la même façon qu'un anticorps). Le principe de cette technologie est illustré en figure 1.6.

Figure 1.6: Principe des puces SomaScan



©Somascan notice

Les protéines (en beige) sont mises au contact des Somamers (en violet) (A) et des complexes spécifiques vont se former (B). Après lavage, une biotine (vitamine) est fixée sur chaque protéine (C). Un flux lumineux est dirigé sur le support afin de casser les liens photoclivables qui retiennent les complexes (D). Les complexes non spécifiques se cassent (E). Les protéines sont fixées à un autre support au niveau de la biotine ajoutée en C et les Somamers libres sont lavés (F). Les somamers sont dissociés des protéines (G), puis sont récupérés et quantifiés par fluorescence (H).

1.2 Contexte, collaborateurs principaux et structuration du mémoire

Post-doctorante dans l'équipe Inria Baobab, j'ai travaillé sur la classification de courbes avec Franck Picard (LBBE: Laboratoire de Biométrie et Biologie Evolutive, Lyon), Madison Giacofci (LJK: Laboratoire Jean Kuntzmann, Grenoble) et Sophie Lambert-Lacroix (TIMC-IMAG Techniques de l'Ingénierie Médicale et de la Complexité - Informatique, Mathématiques, Applications, Grenoble). Notre travail était motivé par des applications génomiques et protéomiques. L'innovation statistique concernait l'introduction d'un effet aléatoire dans la classification de courbes afin de prendre en compte une variabilité individuelle. Bien qu'antérieur à certains travaux décrits dans le prochain chapitre, ce travail de post-doctorat sera présenté dans le chapitre 3 afin de le regrouper avec d'autres travaux sur les variations du nombre de copies d'ADN.

Recrutée en 2010 sur un poste à l'interface entre la faculté de médecine de Lille et une équipe-projet Inria dont la majorité des membres était affiliée au laboratoire de mathématiques Paul Painlevé de la faculté des Sciences et Technologies de Lille, j'ai eu la chance d'être au cœur de projets inter-disciplinaires. Le chapitre 2 décrira des travaux à la fois méthodologiques et de transfert logiciel dans la continuité de mes travaux de thèse sur l'utilisation d'approches bayésiennes empiriques dans les analyses de données transcriptomiques. Ma thèse portait essentiellement sur des applications de données de puces à ADN. Mes travaux lillois ont étendu les résultats de ma thèse à des données de séquençage à haut débit. Ce travail a été effectué en collaboration avec Andrea Rau et Florence Jaffrézic (Laboratoire Génétique Animale et Biologie Intégrative, Jouy-en Josas). Le transfert logiciel des packages R implémentant ces travaux vers Galaxy, une plateforme basée sur les technologies web pour l'analyse de données en recherche biomédicale a été essentiellement réalisé par Samuel Blanck (METRICS, Lille), sous mon encadrement. Le chapitre 3 illustrera le passage d'un projet inter-disciplinaire (MPAGenomics) vers une question de recherche statistique plus ciblée à l'origine de la thèse de Quentin Grimonprez sur la sélection de groupes de variables en grande dimension, co-encadrée par Alain Celisse et Julien Jacques (Laboratoire Paul Painlevé, membres de mon équipe-projet Inria MODAL, Lille). Un des produits dérivés du projet interdisciplinaire MPAGenomics a été le développement d'un nouvel algorithme de détection de ruptures à partir de méthodes à noyaux, en collaboration avec Alain Celisse (Painlevé & MODAL, Lille), Guillem Rigaill (Laboratoire Statistique et Génome et UMR 9213/UMR1403, Evry) et Morgane Pierre-Jean (Laboratoire Statistique et Génome, Evry).

Ayant acquis une réelle expertise de l'analyse de différents niveaux -omiques au cours de ma thèse, mon post-doctorat et mes premières années de maître de conférences, il était tout naturel de souhaiter intégrer ces différents niveaux -omiques. La thèse de Quentin Grimonprez a donc été suivie de la thèse d'Hélène Sarter, co-encadrée par Corinne Gower (Laboratoire Infinite, Lille) sur la sélection de variables dans un contexte d'intégration de données -omiques et cliniques. Ce travail en cours est présenté succintement dans le chapitre 4. Il en est de même pour les pistes de recherche de Wilfried Heyse, qui a commencé sa thèse il y a un an sur la prise en compte de la structure temporelle dans l'analyse statistique de données protéomiques à haut débit. Je co-encadre actuellement Wilfried avec Christophe Bauters, cardiologue de l'équipe de Florence Pinet (Labex U1167 RID-AGE). Pour les aspects statistiques, Wilfried est aussi co-encadré par Vincent Vandewalle,

statisticien appartenant comme moi à l'équipe-projet Inria MODAL et à l'ULR2694 METRICS.
Le point commun de ces trois thèses est la sélection de variables pour la construction de scores. Cette construction de scores est un des thèmes privilégiés de mon équipe universitaire METRICS, dont l'axe 2 du dernier projet HCERES est l'évaluation clinique. Le cadre statistique privilégié est celui des régressions pénalisées pour pallier au problème dit de grande dimension (plus de variables que d'individus).

Par ailleurs, au regard du temps passé pour la communauté en biologie-santé en tant que co-responsable de la plateforme de bioinformatique bilille, j'aborderai dans le chapitre 5 la (re-)construction de cette plateforme ainsi que des travaux plus divers et appliqués que j'ai pu encadrer.

Chapter 2

Méta-analyse de données transcriptomiques

Les analyses différentielles en transcriptomique reposent généralement sur un faible nombre d'individus. Il n'est pas rare de voir moins de dix individus par condition, ce qui nécessite d'utiliser des approches statistiques appropriées. Durant ma thèse, j'avais travaillé sur la modélisation des variances dans des analyses de puces à ADN et implémenté le package R SMVar, disponible sur le CRAN, site officiel de R. La modélisation reposait sur une approche de rétrécissement ("shrinkage" en anglais), indiquant que l'estimateur est un compromis entre deux estimateurs. De manière générale, l'estimateur de rétrécissement $\tilde{\boldsymbol{\theta}}_{\boldsymbol{g}}$ peut s'écrire comme une fonction d'un estimateur gène à gène $\widehat{\boldsymbol{\theta}_{\boldsymbol{g}}}$ et d'un estimateur commun de la population globale $\widehat{\boldsymbol{\theta}_{\boldsymbol{c}}}$:

$$\tilde{\boldsymbol{\theta}}_{\boldsymbol{g}} = \widehat{\boldsymbol{\theta}_{\boldsymbol{c}}} + b(\widehat{\boldsymbol{\theta}_{\boldsymbol{g}}} - \widehat{\boldsymbol{\theta}_{\boldsymbol{c}}}) \tag{2.1}$$

où b est le facteur de rétrécissement. Quand $b = 1$, $\tilde{\boldsymbol{\theta}}_{\boldsymbol{g}} = \widehat{\boldsymbol{\theta}_{\boldsymbol{g}}}$ (estimateur empirique gène à gène). Quand $b = 0$, $\tilde{\boldsymbol{\theta}}_{\boldsymbol{g}} = \widehat{\boldsymbol{\theta}_{\boldsymbol{c}}}$ (estimateur commun). Les approches de rétrécissement diminuent considérablement le nombre de paramètres à estimer tout en gardant une certaine flexibilité avec une valeur par gène.

Quand très peu d'échantillons biologiques sont considérés et que l'analyse est réalisée gène à gène, les tests statistiques manquent de puissance; dans ce contexte, cela veut dire que très peu de gènes différentiellement exprimés peuvent être détectés. Une alternative est de supposer une variance commune à tous les gènes. Cependant, cela conduit souvent à une augmentation du nombre de faux positifs. Dans le cadre de ma thèse, θ de l'équation (2.1) correspondait au log des variances et s'écrivait comme un modèle mixte avec un effet condition fixe et un effet gène aléatoire. Le facteur de rétrécissement était estimé via une approche bayésienne empirique. Les variances modérées estimées ainsi étaient ensuite insérées gène à gène dans les statistiques de Welch pour tester les différences de moyennes entre deux conditions [Jaffrézic et al., 2007]. Cette stratégie est très proche de celle du package Bioconductor limma [Ritchie et al., 2015], maintenant utilisé internationalement en routine pour l'analyse différentielle de puces à ADN. Le package limma implémente lui aussi une approche bayésienne empirique [Smyth, 2004, Phipson et al., 2016]. Le modèle est légèrement différent et suppose une variance commune aux deux conditions alors que SMVar suppose une variance différente entre les deux conditions.

Le package limma ayant été développé au sein d'un environnement de recherche permettant à la fois une maintenance et une évolution du package en incluant régulièrement de nouvelles extensions statistiques, il ne m'a pas paru pertinent de continuer à développer SMVar. J'ai moi-même utilisé limma après ma thèse dans plusieurs analyses de jeux de données réelles [Martin et al., 2014, Herbaux et al., 2016, Poulain et al., 2016, Mogilenko et al., 2019]. J'ai cependant aussi utilisé SMVar quand l'hypothèse de variances homogènes entre les conditions était loin d'être respectée [Valour et al., 2013]. Une autre partie de ma thèse avait concerné la méta-analyse de données de puces à ADN pour augmenter la sensibilité et diminuer le nombre de faux positifs, quand plusieurs études répondant à la même question biologique étaient disponibles. J'avais proposé une approche pour combiner des tailles d'effets modérées [Marot et al., 2009]. Bien que mon approche soit meilleure que d'autres stratégies déjà existantes pour combiner des tailles d'effet, j'avais montré que les combinaisons de p-values de t-tests modérés étaient plus performantes que les autres méthodes de méta-analyse testées en terme de sensibilité. Cela est inhabituel en recherche clinique où on privilégie plutôt les méta-analyses reprenant les données de départ plutôt que celles basées sur des p-values [Haidich, 2010]. Mes résultats suggéraient donc que pour les puces à ADN, l'apport de la modération par approche bayésienne empirique est plus importante que la modélisation de l'effet étude (même si celui-ci doit être pris en compte quand il existe). Les approches testées étant essentiellement relatives aux puces à ADN, des collègues de Jouy-en-Josas m'ont sollicitée au début de mon poste de maître de conférences car elles souhaitaient étendre cette méta-analyse à des données de séquençage à haut débit. Notre collaboration a abouti au papier de BMC Bioinformatics présenté ci-après et j'ai développé le package metaRNASeq à cette occasion [Marot and Bruyère, 2015]. J'en assure encore la maintenance avec l'aide de Samuel Blanck. Contactée régulièrement pour des besoins de méta-analyse de données de puces à ADN ou de séquençage à haut débit, j'ai sollicité les compétences de Samuel pour interfacer les packages R metaMA et metaRNASeq avec Galaxy, une plateforme basée sur les technologies web pour l'analyse de données en recherche biomédicale. Nous avons publié cet outil Galaxy SMAGEXP dans GigaScience [Blanck and Marot, 2019]. Ce papier a été l'occasion de pointer des différences entre les méta-analyses des puces à ADN et celles de séquençage à haut débit. Une des différences très connues est celle des lois statistiques généralement utilisées pour modéliser ces expériences (loi normale pour les puces à ADN, loi binomiale négative pour le séquençage à haut débit). Une autre différence souvent méconnue des utilisateurs, que nous avons pointée dans ce deuxième article présenté dans ce chapitre est la gestion des conflits (par exemple surexpression dans une étude et sous expression dans l'autre).

Rau *et al. BMC Bioinformatics* 2014, **15**:91
http://www.biomedcentral.com/1471-2105/15/91

BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

Differential meta-analysis of RNA-seq data from multiple studies

Andrea Rau[1,2*], Guillemette Marot[3,4] and Florence Jaffrézic[1,2]

Abstract

Background: High-throughput sequencing is now regularly used for studies of the transcriptome (RNA-seq), particularly for comparisons among experimental conditions. For the time being, a limited number of biological replicates are typically considered in such experiments, leading to low detection power for differential expression. As their cost continues to decrease, it is likely that additional follow-up studies will be conducted to re-address the same biological question.

Results: We demonstrate how *p*-value combination techniques previously used for microarray meta-analyses can be used for the differential analysis of RNA-seq data from multiple related studies. These techniques are compared to a negative binomial generalized linear model (GLM) including a fixed study effect on simulated data and real data on human melanoma cell lines. The GLM with fixed study effect performed well for low inter-study variation and small numbers of studies, but was outperformed by the meta-analysis methods for moderate to large inter-study variability and larger numbers of studies.

Conclusions: The *p*-value combination techniques illustrated here are a valuable tool to perform differential meta-analyses of RNA-seq data by appropriately accounting for biological and technical variability within studies as well as additional study-specific effects. An R package `metaRNASeq` is available on the CRAN (http://cran.r-project.org/web/packages/metaRNASeq).

Keywords: Meta-analysis, RNA-seq, Differential expression, *p*-value combination

Background

Studies of gene expression have increasingly come to rely on the use of high-throughput sequencing (HTS) techniques to directly sequence libraries of reads (i.e., nucleotide sequences) arising from the transcriptome (RNA-seq), yielding counts of the number of reads arising from each gene. Due to the cost of HTS experiments, for the time being RNA-seq experiments are typically performed on very few biological replicates, and therefore analyses to detect differential expression between two experimental conditions tend to lack detection power. However, as costs continue to decrease, it is likely that additional follow-up experiments will be conducted to re-address some biological questions, suggesting a future need for methods able to jointly analyze data from multiple studies. In particular, such methods must be able to appropriately account for the biological and technical variability among samples within a given study as well as for the additional variability due to study-specific effects. Such inter-study variability may arise due to technical differences among studies (e.g., sample preparation, library protocols, batch effects) as well as additional biological variability.

In recent years, several methods have been proposed to analyze microarray data arising from multiple independent but related studies; these meta-analysis techniques have the advantage of increasing the available sample size by integrating related datasets, subsequently increasing the power to detect differential expression. Such meta-analyses include, for example, methods to combine *p*-values [1], estimate and combine effect sizes [2], and rank genes within each study [3]; Hu *et al.* [4] and Hong and Breitling [5] provide a review and comparison of such

*Correspondence: andrea.rau@jouy.inra.fr
[1]INRA, UMR1313 Génétique animale et biologie intégrative, 78352 Jouy-en-Josas, France
[2]AgroParisTech, UMR1313 Génétique animale et biologie intégrative, 75231 Paris 05, France
Full list of author information is available at the end of the article

methods, and Tseng *et al.* [6] present a recent literature review and discussion of statistical considerations for microarray meta-analysis. In particular, Marot *et al.* [1] showed that the inverse normal p-value combination technique outperformed effect size combination methods or moderated t-tests [7] obtained from a linear model with a fixed study effect on several criteria, including sensitivity, area under the Receiver Operating Characteristic (ROC) curve, and gene ranking.

In many cases the meta-analysis techniques previously used for microarray data are not directly applicable for RNA-seq data. In particular, differential analyses of microarray data, whether for one or multiple studies, typically make use of a standard or moderated t-test [7,8], as such data are continuous and may be roughly approximated by a Gaussian distribution after log-transformation. On the other hand, the growing body of work concerning the differential analysis of RNA-seq data has primarily focused on the use of overdispersed Poisson [9] or negative binomial models [10,11] in order to account for their highly heterogeneous and discrete nature. Under these models, the calculation and interpretation of effect sizes is not straightforward. Kulinskaya *et al.* [12] recently proposed an effect size combination method for Poisson-distributed data, based on an Anscombe transformation, but this method is not well-adapted to RNA-seq data due to the presence of over-dispersion among biological replicates as well as zero-inflation. To our knowledge, no other transformation has been proposed to obtain effect sizes for over-dispersed Poisson or negative binomial data.

In this paper, we consider several methods for the integrated analysis of RNA-seq data arising from multiple related studies, including two p-value combination methods as well as a model fitted over the full data with a fixed study effect. We first demonstrate how the inverse normal and Fisher p-value combination methods can be adapted to the differential meta-analysis of RNA-seq data. Then we compare these two methods to the results of independent per-study analyses and a negative binomial generalized linear model (GLM) with a fixed study effect as implemented in the `DESeq` Bioconductor package [10]. All methods are compared on real data from two related studies on human melanoma cell lines, as well as in an extensive set of simulations varying the inter-study variability, number of studies, and biological replicates per study.

Finally, we note that our focus is on RNA-seq data arising from two or more studies in which all experimental conditions under consideration are included in every study (with potentially different numbers of biological replicates); differential analyses among conditions that are not studied in the same experiment are typically limited, or even compromised, due to the confounding of condition and study effects.

Methods

Let y_{gcrs} be the observed count for gene g ($g = 1, \ldots, G$), condition c ($c = 1, 2$), biological replicate r ($r = 1, \ldots R_{cs}$), and study s ($s = 1, \ldots, S$). Note that the number of biological replicates R_{cs} may vary between conditions and among studies. We use dot notation to indicate summations in various directions, e.g., $y_{gc\cdot s} = \sum_r y_{gcrs}$, $y_{g\cdot\cdot s} = \sum_c \sum_r y_{gcrs}$, and so on. Let μ_{gcs} be the mean expression level for gene g in condition c and study s. For an integrated differential analysis of gene expression across all studies, two approaches can be envisaged: the combination of p-values from per-study differential analyses, and a global differential analysis. We illustrate both using the default methods and parameters of the `DESeq` (v1.10.1) analysis pipeline [10], although other popular methods, e.g., `edgeR` [11], could also be used; we note that the recent extensive comparison of Soneson and Delorenzi [13] provides a helpful guide to choosing an appropriate method and software package to use in practice.

P-value combination from independent analyses

For the differential analysis of gene expression within a given study s, we assume that gene counts y_{gcrs} follow a negative binomial distribution parameterized by its mean $\eta_{gcrs} = \ell_{crs}\mu_{gcs}$ and dispersion ϕ_{gs}, where ℓ_{crs} is a normalization factor to correct for differences in library size. A comparison of different methods to estimate ℓ_{crs} may be found in Dillies *et al.* [14].

After obtaining per-gene mean and dispersion parameter estimates in each study independently, a parametric gamma regression is used to obtain fitted dispersion estimates by pooling information from genes with similar expression strengths. Subsequently, for each gene in each study, the null hypothesis to be tested is that there is no difference in the relative proportion of read counts attributed to each condition, or in other words, that the gene is non-differentially expressed. Per-gene and per-study p-values p_{gs} are computed using a conditioned test analogous to Fisher's exact test, where the p-value of a pair of observed count sums $(y_{g1\cdot s}, y_{g2\cdot s})$ is calculated as the sum of all probabilities less than $p(y_{g1\cdot s}, y_{g2\cdot s})$ given the overall sum $y_{g\cdot\cdot s}$:

$$p_{gs} = \frac{\sum\limits_{\substack{a,b\geq 0 \\ a+b=y_{g\cdot\cdot s} \\ p(a,b)\leq p(y_{g1\cdot s},y_{g2\cdot s})}} p(a,b)}{\sum\limits_{\substack{a,b\geq 0 \\ a+b=y_{g\cdot\cdot s}}} p(a,b)}$$

where it is assumed that $p(a, b) = p(a)p(b)$, and $p(a)$ and $p(b)$ represent the probability of a and b counts in the first

and second conditions, respectively. These probabilities are calculated using the negative binomial distributions parameterized by the corresponding estimated mean and dispersion parameters, μ_{gcs} and ϕ_{gs}.

Additional details are described by Anders and Huber [10] and in the `DESeq` package vignette. Once these vectors of raw p-values have been obtained for each study, we consider two possible approaches to combine them: the inverse normal and the Fisher combination methods. We note that both of these approaches assume that under the null hypothesis, each vector of p-values is assumed to be uniformly distributed.

Inverse normal method

For each gene g, we define

$$N_g = \sum_{s=1}^{S} w_s \Phi^{-1}(1 - p_{gs}) \quad (1)$$

where p_{gs} corresponds to the raw p-value obtained for gene g in a differential analysis for study s, Φ the cumulative distribution function of the standard normal distribution, and w_s a set of weights [15,16]. We propose here to define the study-specific weights w_s, as described by Marot and Mayer [17]:

$$w_s = \sqrt{\frac{\sum_c R_{cs}}{\sum_\ell \sum_c R_{c\ell}}},$$

where $\sum_c R_{cs}$ is the total number of biological replicates in study s. This allows studies with large numbers of biological replicates to be attributed a larger weight than smaller studies. We note that other weights may also be defined by the user depending on the quality of the data in each study, if this information is available.

Under the null hypothesis, the test statistic N_g in Equation (1) follows a $\mathcal{N}(0,1)$ distribution. A unilateral test on the right-hand tail of the distribution may then be performed, and classical procedures for the correction of multiple testing such as the approach of Benjamini and Hochberg [18] may subsequently be applied to the obtained p-values to control the false discovery rate at a desired level α.

Fisher combination method

For the Fisher combination method [19], the test statistic for each gene g may be defined as

$$F_g = -2 \sum_{s=1}^{S} \ln\left(p_{gs}\right), \quad (2)$$

where as before p_{gs} corresponds to the raw p-value obtained for gene g in a differential analysis for study s. Under the null hypothesis, the test statistic F_g in Equation (2) follows a χ^2 distribution with $2S$ degrees of freedom. As with the inverse normal p-value combination method, classical procedures for the correction of multiple testing [18] may be applied to the obtained p-values to control the false discovery rate at a desired level α.

Additional considerations for p-value combination

We note that the implementation of the previously described p-value combination techniques requires two additional considerations to be taken into account when dealing with RNA-seq data.

First, a crucial underlying assumption for the statistics defined in Equations (1) and (2) is that p-values for all genes arising from the per-study differential analyses are uniformly distributed under the null hypothesis. This assumption is, however, not always satisfied for RNA-seq data; in particular, a peak is often observed for p-values close to 1 due to the discretization of p-values for very low counts. To circumvent this first difficulty, as is commonly done for differential analyses in practice, we propose to filter the weakly expressed genes in each study, using the `HTSFilter` Bioconductor package [20] as described in the Additional file 1. We note that in so doing, it is possible for a gene to be filtered from one study and not from another. As will be seen in the following, this approach appears to effectively filter those genes contributing to a peak of large p-values, resulting in p-values that are roughly uniformly distributed under the null hypothesis.

Second, for the two p-value combination methods described above, unlike microarray data, under- and over-expressed genes are analyzed together for RNA-seq data. As such, some care must be taken to identify genes exhibiting conflicting expression patterns (i.e., under-expression when comparing one condition to another in one study, and over-expression for the same comparison in another study). In the case of microarray data, Marot *et al.* [1] suggested the use of one-tailed p-values for each study to avoid directional conflicts; as the inverse normal combination method was used in their work, the combined statistic thus follows a normal distribution, which is symmetric. Because under- and over-expressed genes may be found in the left and right tail, respectively, of the corresponding normal distribution, it is thus possible to use a two-tailed test to simultaneously study over and under-expressed genes. Note that Pearson [21] and Owen [22] proposed another alternative to handle conflicting differential expression if the Fisher combination method is used instead. However, in the case of RNA-seq data, the use of the conditioned test described above does not enable the separation of over- and under-expressed genes in distribution tails; this implies that it is not possible to use the approaches proposed by Marot *et al.* [1] or Owen [22]. We thus suggest that genes exhibiting differential expression conflicts among studies be identified post hoc, and removed from the list of differentially expressed genes; this step to remove genes with conflicting differential

expression from the final list of differentially expressed genes may be performed automatically within the associated R package `metaRNASeq`.

Global differential analysis

For a global analysis of RNA-seq data arising from multiple studies, we assume that gene counts y_{gcrs} follow a negative binomial distribution parameterized by mean $\eta_{gcrs} = \ell_{crs}\mu_{gcs}$ and dispersion ϕ_g, where ℓ_{crs} is the library size normalization factor. In order to estimate a possible effect due to study, a full and reduced model are fitted for each gene using negative binomial generalized linear models (GLM); the full model regresses gene expression on fixed effects for the experimental condition and study, while the reduced model regresses gene expression only on a fixed effect for the study.

Specifically, the full model is $\log\left(\eta_{gcrs}\right) = \beta_g + \lambda_{gc} + \delta_{gs} + \log(\ell_{crs})$, where β_g is an intercept, λ_{gc} is a fixed condition effect, δ_{gs} a fixed study effect, and $\lambda_{g1} = \delta_{g1} = 0$, with the choice of the condition and study to be used as references being arbitrary. The reduced model is $\log\left(\eta_{gcrs}\right) = \beta_g + \delta_{gs} + \log(\ell_{crs})$. Per-gene dispersion parameters are estimated as before, where a parametric gamma regression is used to obtain fitted dispersion estimates by pooling information from genes with similar expression strengths across all studies.

We are now interested in testing the global per-gene null hypothesis

$$H_{0,g} : \forall c,\ \lambda_{gc} = 0 \text{ vs } H_{1,g} : \exists c \mid \lambda_{gc} \neq 0.$$

Following parameter estimation, the two models are compared using a χ^2 likelihood ratio test (with degrees of freedom equal to the number of conditions minus one) to determine whether including the experimental condition significantly improves the model fit. Note that for the global differential analysis we use the `HTSFilter` Bioconductor package [20] to filter the full set of data across studies prior to calculating p-values, resulting in a single vector of raw filtered per-gene p-values that may be corrected for multiple testing using classical procedures [18] to control the false discovery rate at a desired level α. Additional details may be found in the `DESeq` package vignette.

Results and discussion

Application to real data

Presentation of the data

The negative binomial GLM and p-value combination methods were applied to a pair of real RNA-seq studies performed to compare two human melanoma cell lines [23]. Each study compares gene expression in a melanoma cell line expressing the Microphtalmia Transcription Factor (MiTF) to one in which small interfering RNAs (siRNAs) were used to repress MiTF, with three biological replicates per cell line in the first (hereafter referred to as Study A) and two per cell line in the second (Study B). The raw read counts and phenotype tables for Study A are available in the Supplementary Materials of Dillies et al. [14], and the data from Study B from Strub et al. [23].

The characteristics of the data from these two studies are summarized in Additional file 1: Table S1. In particular, we note that the data from Study A tend to have larger total library sizes and a smaller number of unique reads (i.e. reads that appear once in the reference genome) than those from Study B; in addition, Study A appears to exhibit larger overall per-gene variability than does Study B (Additional file 1: Figure S8). These two points indicate that in this pair of studies, a considerable amount of inter-study heterogeneity appears to be present (Additional file 1: Figure S9).

Results

After performing individual differential analysis for each study using the negative binomial model and exact test as described in the previous section, we obtained per-gene p-values for each study (Figure 1, histograms in background). As previously stated, an important underlying assumption of the p-value combination methods is that the p-values are uniformly distributed under the null hypothesis; we note that this is not the case here, especially for the second study, due to a large peak of values close to 1 resulting from the discretization of p-values. In order to remove the weakly expressed genes contributing to this peak in each study, we filtered the data from each study as proposed in Rau *et al.* [20], resulting in a distribution of raw p-values from each study that appears to satisfy the uniformity assumption under the null hypothesis (Figure 1, histograms in foreground).

The per-study filtered p-values were combined using the test statistics defined in Equations (1) and (2), and the corresponding results were compared to those of the intersection of independent per-study analyses and the global analysis using a negative binomial GLM with a fixed study effect as previously described. We note that for the independent per-study differential analyses, a gene is declared to be differentially expressed if identified in both studies with no differential expression conflict. For the Inverse normal and Fisher methods, a total of 310 (6.8% of differentially expressed genes) and 439 (9.0% of differentially expressed genes) genes were respectively found to exhibit conflicting expression between the two studies, and were subsequently removed from the final list of differentially expressed genes. Unsurprisingly, these genes tended to be those with relatively large p-values in both studies (Additional file 1: Figure S11).

In addition, we also investigated whether genes identified as differentially expressed by the Inverse normal and Fisher methods tended to be disproportionately

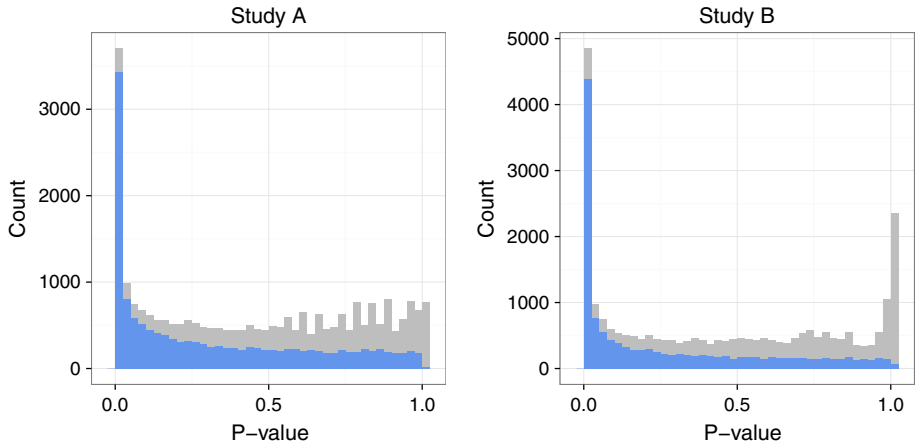


Figure 1 Raw *p*-value histograms from per-study differential analyses of real data. Histograms of raw *p*-values obtained from per-study differential analyses in the real data from Study A (left) and Study B (right): unfiltered (in grey) and filtered (in blue) using the method of [20]. Figure made using the `ggplot2` package [24].

dominated by one study over the other, i.e. very small *p*-values in only one study (Additional file 1: Figure S12). Although Study B appears to have slightly more genes with very small *p*-values, for the most part, *p*-values for differentially expressed genes tend to be well-balanced between the two studies.

The Venn diagram presented in Figure 2 compares the lists of differentially expressed genes found for all methods considered. It may immediately be noticed that the independent per-study analysis approach is very conservative, and both of the *p*-value combination approaches (Fisher and inverse normal) considerably increase the detection power. In addition, a large number of genes are found in common among the *p*-value combination methods and the global analysis (3578 compared to only 1583 from the intersection of individual studies). In order to determine whether the genes uniquely identified by a particular method appear to be biologically pertinent, an Ingenuity Pathways Analysis (Ingenuity® Systems, www.ingenuity.com) was performed to identify functional annotation for the genes uniquely identified by the Fisher *p*-value combination method with respect to the global analysis, and vice versa. We note that the sets of genes uniquely identified by the Fisher method or the global analysis (Additional file 1: Tables S2 and S3), as well as the set of genes found in common (Additional file 1: Table S4), all appear to include genes of potential interest related to cancer or melanoma, which was the main focus of this set of studies. As such, for this pair of studies it appears that the union of genes identified by the two approaches may be of biological interest; to further study the effect of number of studies and inter-study variability on the performance of each method, we investigate an extensive set of simulated data in the following section.

Simulation study

Data were simulated according to a negative binomial distribution,

$$Y_{gcrs} \sim \mathcal{NB}\left(\mu_{gcs}, \phi_{gs}\right)$$

where μ_{gcs} and ϕ_{gs} represent the mean and dispersion, respectively, for gene g, condition c and study s, and the mean-variance relationship is defined by

$$\mathrm{Var}\left(Y_{gcrs}\right) = \mu_{gcs} + \frac{\mu_{gcs}^2}{\phi_{gs}}.$$

In order to incorporate inter-study variability, we consider the following situation for the mean parameter μ_{gcs}:

$$\log\left(\mu_{gcs}\right) = \theta_{gc} + \varepsilon_{gcs}, \text{ and } \varepsilon_{gcs} \sim \mathcal{N}\left(0, \sigma^2\right),$$

where θ_{gc} represents the mean for gene g in condition c, ε_{gcs} the variability around these means due to a study- and condition-specific random effect, and σ^2 the size of the inter-study variability. Note that as ε_{gcs} affects μ_{gcs} through a log link, the value of $\exp\left(\varepsilon_{gcs}\right)$ has a multiplicative effect on the mean.

Parameters for simulations

To fix realistic values for the parameters $\left\{\theta_{gc}, \phi_{gs}, \sigma\right\}$, we first performed individual per-study differential analyses by fitting a negative binomial model with the default methods and parameters of the `DESeq` package on the unfiltered human data presented above. The per-study false discovery rate was subsequently controlled at the $\alpha = 0.05$ level [18]. For the genes identified as differentially expressed in both studies, θ_{g1} and θ_{g2} were fixed to be the values of the empirical means (after normalization for library size differences) for each condition across studies. For the remaining genes, we set $\theta_{g1} = \theta_{g2} = \theta_g$ to be the overall empirical mean (after normalization for library

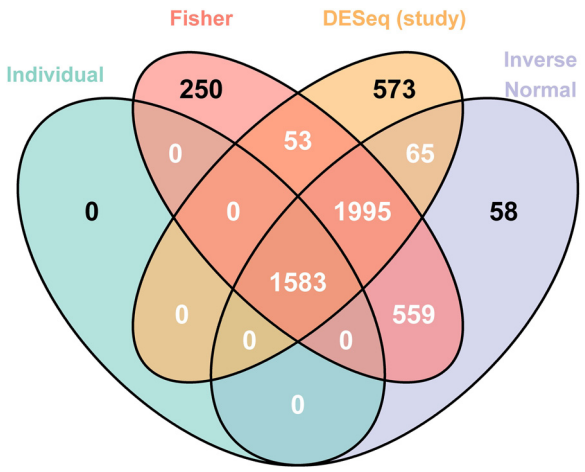


Figure 2 Comparison of results from differential analyses of real data. Venn diagram presenting the results of the differential analysis for the real data for the two meta-analysis methods (Fisher and inverse normal), the global analysis (DESeq (study)), and the intersection of individual per-study analyses (Individual). Figure made using the `VennDiagram` package [25].

size differences) for gene g across both conditions and studies. Using the gamma-family GLM fitted to the per-gene mean and dispersion parameter estimates for each study (Additional file 1: Figure S8), we fixed the dispersion parameter ϕ_{gs} to be equal to the fitted values

$$\phi_{gs}^{-1} = \hat{\gamma}_{0s} + \frac{\hat{\gamma}_{1s}}{\theta_g},$$

where $\hat{\gamma}_{0s}$ and $\hat{\gamma}_{1s}$ are the estimated coefficients from the gamma-family GLM for study s, and θ_g is the overall empirical mean for gene g. For weakly expressed genes, it has been observed that little overdispersion is present as biological variation is dominated by shot noise (i.e., the variation inherent to a counting process); for genes with $\theta_g < 10$, the dispersion parameter is therefore fixed to be $\phi_{gs} = 10^{10}$, which corresponds to nearly zero overdispersion (i.e., mean nearly equal to the variance).

Finally, the parameter σ is chosen to represent a range of values for the amount of inter-study variability. The observed human data exhibit a considerable amount of inter-study variability, corresponding to a value of roughly $\sigma = 0.5$ (see Additional file 1: Figure S9). In the following simulations, four values are considered for the parameter σ: $\{0, 0.15, 0.3, 0.5\}$, representing zero, small, medium, and large inter-study variability, respectively. Finally, we note that for genes simulated to be non-differentially expressed, we set $\varepsilon_{g1s} = \varepsilon_{g2s} = \varepsilon_{gs} \sim \mathcal{N}(0, \sigma^2)$.

The simulation settings used for the number of studies and number of replicates per condition in each study are presented in Table 1 and were chosen to reflect the size of real RNA-seq experiments. When more than two studies were simulated, the same simulation parameters were used as for the first two, as determined from the real data. For simplicity, the same number of replicates was simulated in each condition for all studies.

Methods and criteria for comparison

In addition to the intersection of independent per-study analyses (where genes were declared to be differentially expressed if identified in more than half of the studies with no differential conflict), the Fisher and inverse normal p-value combination techniques, and the global analysis with fixed study effect, we also considered a global analysis with no study effect. For each simulation setting and level of inter-study variability σ, 300 independent datasets were simulated, and the filtering method of Rau *et al.* [20] was applied, either independently to each study (for the independent per-study analyses and p-value combination techniques) or to the full set of data (for the global analysis).

For each method, performance was assessed using the sensitivity, false discovery rate (FDR) and area under the receiver operating characteristic (ROC) curve (AUC). In addition, we also considered a criterion to assess the "value added" for the p-value combination methods with

Table 1 Parameter settings for the simulations, including the number of studies and the number of replicates per condition in each study

Setting	# of studies	Replicates/study
1	2	(2,3)
2	3	(2,2,3)
3	5	(2,2,3,3,3)

Parameter settings for the simulations, including the number of studies and the number of replicates per condition in each study.

respect to the global analysis, and vice versa: the proportion of true positives among those uniquely identified by a given method (e.g., the Fisher approach) as compared to another (e.g., the global analysis).

Results

The different methods were first compared with ROC curves, presented in Figure 3 for low and high inter-study variability (results for zero and moderate inter-study variability are shown in Additional file 1: Figure S5). We note that for clarity, the inverse normal method is not represented on these plots as its performance was found to be equivalent to the Fisher method. It can first be noted that for no or small inter-study variability ($\sigma = 0$ or $\sigma = 0.15$), no practical difference may be observed among the methods. On the other hand, for moderate to large inter-study variability ($\sigma = 0.3$ or $\sigma = 0.5$) differences among the methods become more apparent; this pattern is observed for any number of studies. As expected, including a study effect in the global analysis improves the performance over a naive global analysis without such an effect. We note that the two proposed meta-analysis methods (inverse normal and Fisher *p*-value combination) were found to perform very similarly and were able, in the case of large inter-study variability, to outperform the global analysis in terms of AUC (Additional file 1: Figure S1). In particular, in the presence of large inter-study variability, the naive global analysis without a study effect unsurprisingly has the lowest AUC, and the two meta-analysis methods yield a larger AUC than the global analysis with a study effect.

Considering the sensitivity (Figure 4 and Additional file 1: Figure S6), the meta-analyses appear to lead to similar, and in some settings considerably higher, detection power compared to the other methods. We note that in all settings, using the intersection of independent analyses leads to much lower sensitivity, even for low or zero inter-study variability. As for the AUC, the sensitivity was found to be considerably improved for the global analysis when including a study effect in the GLM model, particularly for medium to large inter-study variability. The two meta-analysis methods were found to lead to significant improvements in sensitivity as compared to the global analysis in the presence of moderate to large inter-study variability when three or more studies were considered. However, for the setting that most resembles our real

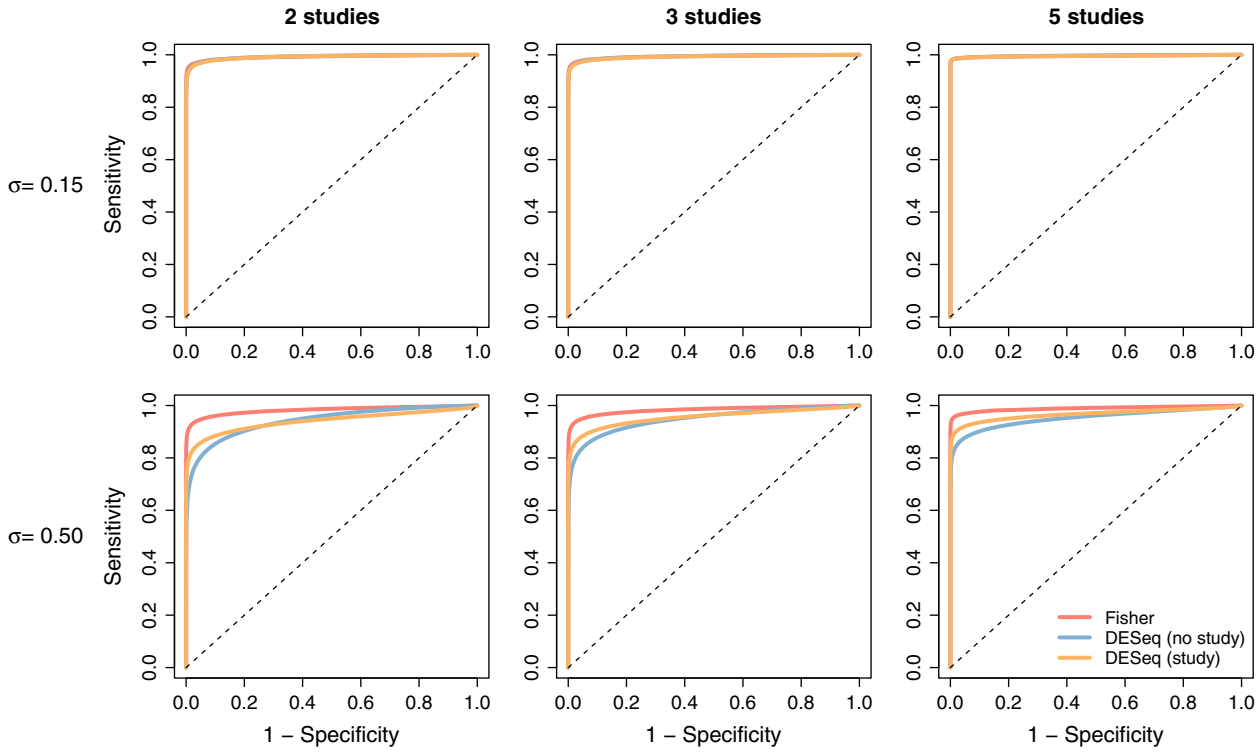


Figure 3 Receiver operating characteristic curves for low and high inter-study variability. Receiver Operating Characteristic (ROC) curves, averaged over 300 datasets. Each plot represents the results of a particular setting, with columns corresponding (from left to right) to simulations including 2 studies, 3 studies, and 5 studies, and rows corresponding (from top to bottom) to simulations with inter-study variability set to $\sigma = 0.15$ and $\sigma = 0.50$ (small to large inter-study variability). Within each plot: Fisher (red lines), global analysis with no fixed study effect (DESeq (no study), blue lines), and global analysis with a fixed study effect (DESeq (study), orange lines). The dotted black line represents the diagonal.

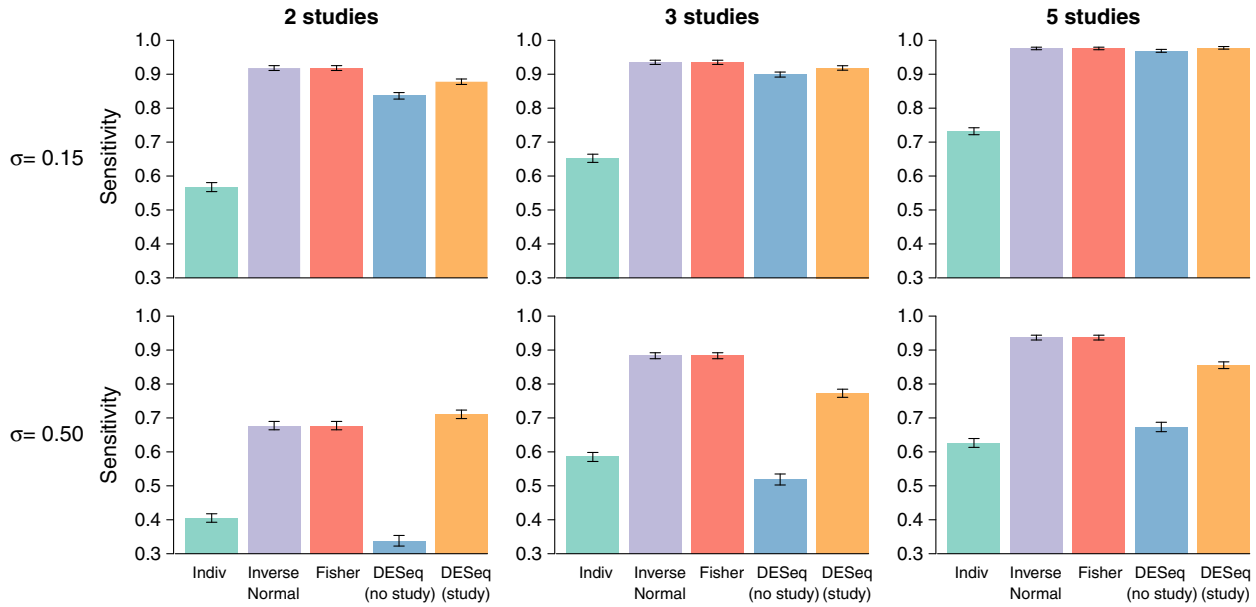


Figure 4 Sensitivity for low and high inter-study variability. Sensitivity, for the simulation settings corresponding to $\sigma = 0.15$ and $\sigma = 0.50$. Each barplot represents the results of a particular setting, with columns corresponding (from left to right) to simulations including 2 studies, 3 studies, and 5 studies, and rows corresponding (from top to bottom) to simulations with inter-study variability set to $\sigma = 0.15$ and $\sigma = 0.50$ (low inter-study variability to large inter-study variability). Within each barplot, from left to right: Individual per-study analyses (green bars), inverse normal (purple bars), Fisher (red bars), global DESeq with no study effect (blue bars), and global DESeq with a fixed study effect (orange bars).

data analysis (2 studies, $\sigma = 0.50$), the global analysis with study effect and meta-analyses appear to have similar detection power. Finally, we also note that for all methods the FDR was well controlled below 5% (Additional file 1: Figure S2).

Based on these criteria, the two proposed meta-analysis methods (inverse normal and Fisher) seem to perform very similarly. In order to more thoroughly investigate the differences between p-value combination methods and the global analysis including a study effect, we calculated the proportion of true positives uniquely detected by the Fisher method as compared to the global analysis with study effect, and vice versa (Figure 5 and Additional file 1: Figure S7). In the setting closest to the real data analysis presented above (two studies and large inter-study variability), the proportion of true positives found uniquely by either the Fisher approach or the global analysis with fixed study effect are very large (around 80% for both methods). This seems to suggest that the additional genes uniquely found either by the global analysis or Fisher p-value combination method in the real data application may indeed be of great biological interest. For more than two studies, however, as the inter-study variability increases the proportion of truly differentially expressed genes uniquely found by the Fisher method increases compared to the global analysis. For example, for three studies with large inter-study variability ($\sigma = 0.5$), the proportion of truly DE genes uniquely found with the Fisher method was equal to more than 80%, whereas it was only around 40% for the global analysis with a study effect.

Conclusions

The aim of this paper was to present and compare different strategies for the differential meta-analysis of RNA-seq data arising from multiple, related studies. As expected, naive analyses such as the overlap of lists of differentially expressed genes found by individual studies or a global analysis not accounting for a study effect perform very poorly. On the other hand, the two proposed meta-analysis methods seem to have very similar performances. For low inter-study variability, the results are very close to those of a global GLM analysis including a study effect. When the inter-study variability increases, however, the gains in performance in terms of AUC, sensitivity, and proportion of true positives among uniquely identified genes for the meta-analysis techniques are significant as compared to the global analysis, particularly for the analysis of data from more than two studies. We note that both of the proposed p-value combination methods are implemented in an R package called `metaRNASeq`, available on the CRAN; a package vignette describing the use of `metaRNASeq` may be found in Additional file 2 as well as by calling `vignette("metaRNASeq")` after loading the package in R.

Our focus in this work is on differential analyses between two experimental conditions, but can readily be

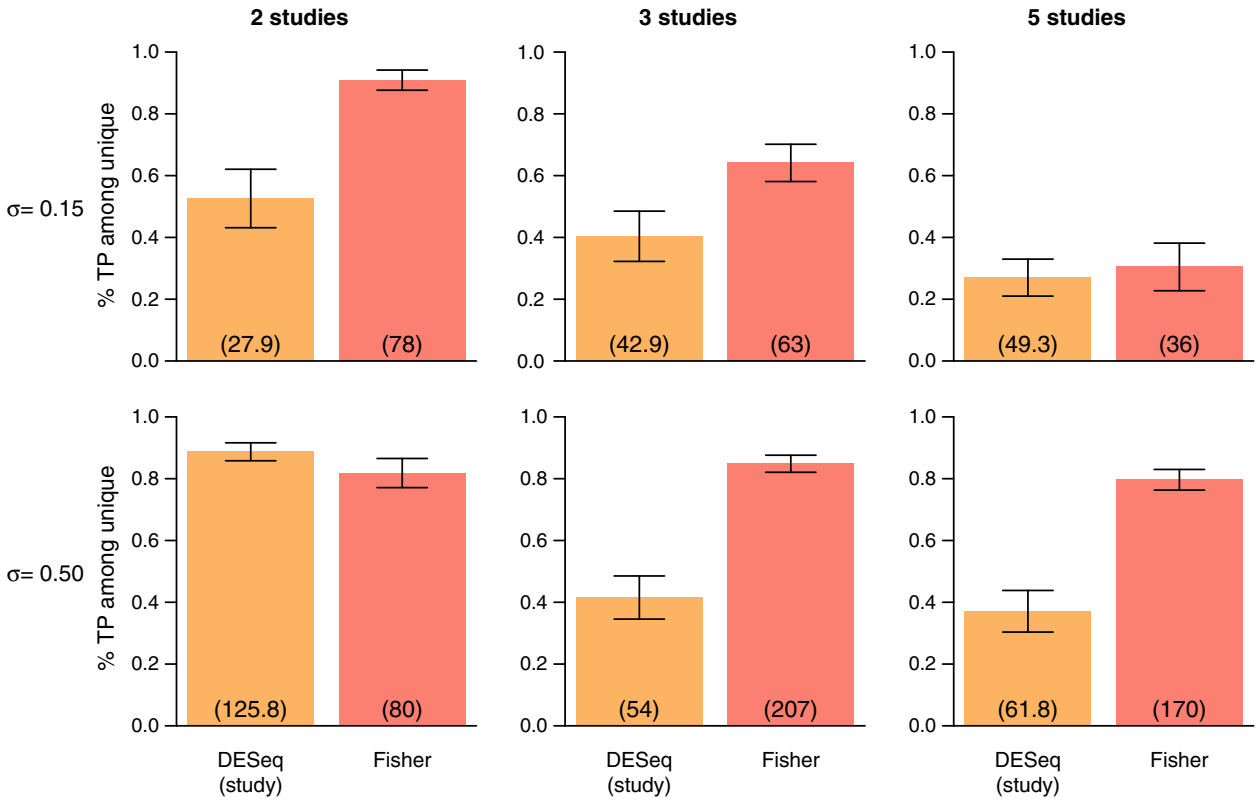


Figure 5 Proportion of true positives among unique discoveries. Proportion of true positives among unique discoveries for DESeq with a fixed study effect (orange bars) and Fisher (red bars). Each barplot represents the results of a particular setting, with columns corresponding (from left to right) to simulations including 2 studies, 3 studies, and 5 studies, and rows corresponding (from top to bottom) to simulations with inter-study variability set to $\sigma = 0.15$ and $\sigma = 0.50$ (low inter-study variability to large inter-study variability). Error bars represent one standard deviation, and numbers in parentheses represent the mean total number of unique discoveries for DESeq with study effect as compared to Fisher and vice versa, respectively.

extended to multi-group comparisons. However, as previously noted, the methods presented here are intended for the analysis of data in which all experimental conditions under consideration are included in every study, thus avoiding problems due to the confounding of condition and study effects. As with all meta-analyses, the p-value combination techniques presented here must overcome differences in experimental objectives, design, and populations of interest, as well as differences in sequencing technology, library preparation, and laboratory-specific effects.

The differential meta-analyses presented here concern expression studies based on RNA-seq data. However, other genomic data are generated by high-throughput sequencing techniques, including chromatin immunoprecipitation sequencing (CHIP-seq) and DNA methylation sequencing (methyl-seq), and the proposed techniques could potentially be extended to these other kinds of data. However, in order to be biologically relevant, the p-value combination methods rely on the fact that the same test statistics, or in the case of RNA-seq data conditioned tests, are used to obtain p-values for each study. An important challenge for the future will be to propose methods able to jointly analyze related heterogeneous data, such as microarray and RNA-seq data, or other kinds of genomic data. This is not straightforward in a meta-analysis framework and remains an open research question.

Additional files

Additional file 1: Supplementary materials. This document contains a discussion concerning the filtering of RNA-seq data, supplementary information about the characteristics of the data and the Ingenuity Pathways Analysis discussed in the real data analysis, and supplementary figures.

Additional file 2: `metaRNASeq` vignette. This document contains a vignette describing in greater detail the `metaRNASeq` R package. It can also be obtained by running the following commands in the R console:

```
> library(metaRNASeq)
> vignette("metaRNASeq")
```

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
AR participated in the design of the study, performed simulations and data analyses, and helped draft the manuscript. GM designed the study, wrote the associated R package, and helped draft the manuscript. FJ conceived of the study, participated in its design, and drafted the manuscript. All authors read and approved the final manuscript.

Acknowledgements
We thank Thomas Strub, Irwin Davidson, Céline Keime and the IGBMC sequencing platform for providing the RNA-seq data, as well as two anonymous reviewers for their helpful comments and suggestions.

Author details
[1]INRA, UMR1313 Génétique animale et biologie intégrative, 78352 Jouy-en-Josas, France. [2]AgroParisTech, UMR1313 Génétique animale et biologie intégrative, 75231 Paris 05, France. [3]Université Lille Nord de France, UDSL, EA2694 Biostatistics, Lille, France. [4]Inria Lille Nord Europe, MODAL, Lille, France.

Received: 13 June 2013 Accepted: 21 March 2014
Published: 29 March 2014

References

1. Marot G, Foulley JL, Mayer CD, Jaffrézic F: **Moderated effect size and P-value combinations for microarray meta-analyses.** *Bioinformatics* 2009, **25**(20):2692–2699. doi:10.1093/bioinformatics/btp444.
2. Choi JK, Yu U, Kim S, Yoo OJ: **Combining multiple microarray studies and model interstudy variation.** *Bioinformatics* 2003, **19**(Suppl 1):84–90.
3. Breitling R, Armengaud P, Amtmann A, Herzyk P: **Rank products: a simple yet powerful new method to detect differential regulated genes in replicated microarray experiments.** *FEBS Lett* 2004, **573:**83–92.
4. Hu P, Greenwood CM, Beyene J: **Statistical methods for meta-analysis of microarray data: a comparative study.** *Inf Syst Front* 2006, **8:**9–20.
5. Hong F, Breitling R: **A comparison of meta-analysis methods for detecting differentially expressed genes in microarray experiments.** *Bioinformatics* 2008, **24**(3):374–382.
6. Tseng GC, Ghosh D, Feingold E: **Comprehensive literature review and statistical considerations for microarray meta-analysis.** *Nucleic Acids Res* 2012, **40**(9):3785–3799.
7. Smyth GK: **Linear models and empirical Bayes methods for assessing differential expression in microarray experiments.** *Stat Appl Genet Mol Biol* 2004, **3**(Article 3). http://dx.doi.org/10.2202/1544-6115.1027.
8. Jaffrézic F, Marot G, Degrelle S, Hue I, Foulley JL: **A structural mixed model for variances in differential gene expression studies.** *Genet Res* 2007, **89:**19–25.
9. Auer P, Doerge R: **A two-stage Poisson model for testing RNA-seq data.** *Stat Appl Genet Mol Biol* 2011, **10**(26):1–26.
10. Anders S, Huber W: **Differential expression analysis for sequence count data.** *Genome Biol* 2010, **11**(R106). doi:10.1186/gb-2010-11-10-r106.
11. Robinson MD, McCarthy DJ, Smyth GK: **edgeR: a Bioconductor package for differential expression analysis of digital gene expression data.** *Bioinformatics* 2010, **26:**139–140.
12. Kulinskaya E, Morgenthaler S, Staudte RG: *Meta Analysis: a guide to calibrating and combining statistical evidence, Volume Volume 756 of Wiley Series in Probability and Statistics*. West Sussex, England: John Wiley & Sons; 2008.
13. Soneson C, Delorenzi M: **A comparison of methods for differential expression analysis of RNA-seq data.** *BMC Bioinformatics* 2013, **14:**91.
14. Dillies MA, Rau A, Aubert J, Hennequet-Antier C, Jeanmougin M, Servant N, Keime C, Marot G, Castel D, Estelle J, Guernec G, Jagla B, Jouneau L, Laloë D, Le Gall C, Schaëffer B, Le Crom S, Jaffrézic F: **A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis.** *Brief Bioinform* 2012. [doi:10.1093/bib/bbs046].
15. Stouffer S, Suchman E, DeVinney L, Star S, Williams JRM: *The American soldier. Adjustment during Army life*. Princeton, NJ: Princeton University Press; 1949.
16. Liptak T: **On the combination of independent tests.** *Magyar Tudomanyos. Akademia Matematikai Kutato Intezetenek Kozlemenyei* 1958, **3:**171–197.
17. Marot G, Mayer CD: **Sequential analysis for microarray data based on sensitivity and meta-analysis.** *Stat Appl Genet Mol Biol* 2009, **8**(Article 3). [doi:10.2202/1544-6115.1368].
18. Benjamini Y, Hochberg Y: **Controlling the false discovery rate: a practical and powerful approach to multiple testing.** *J Roy Statist Soc Ser B* 1995, **57:**289–300.
19. Fisher RA: *Statistical Methods for Research Workers*. Edinburgh: Oliver and Boyd; 1932.
20. Rau A, Gallopin M, Celeux G, Jaffrézic F: **Data-based filtering for replicated high-throughput transcriptome sequencing experiments.** *Bioinformatics* 2013, **29**(17):2146–52.
21. Pearson K: **On a new method of determining "goodness of fit".** *Biometrika* 1934, **26:**425–442.
22. Owen AB: **Karl Pearson's meta-analysis revisited.** *Annals of Statistics* 2009, **37**(6B):3867–3892.
23. Strub T, Giuliano S, Ye T, Bonet C, Keime C, Kobi D, Gras SL, Cormont M, Ballotti R, Bertolotto C, Davidson I: **Essential role of microphthalmia transcription factor for DNA replication, mitosis and genomic stability in melanoma.** *Oncogene* 2011, **30:**2319–2332.
24. Wickham H: *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer; 2009.
25. Chen H, Boutros PC: **VennDiagram: a package for the generation of highly-customizable Venn and Euler diagrams in R.** *BMC Bioinformatics* 2011, **12:**35.

doi:10.1186/1471-2105-15-91
Cite this article as: Rau *et al.*: Differential meta-analysis of RNA-seq data from multiple studies. *BMC Bioinformatics* 2014 **15**:91.

GigaScience, 8, 2019, 1–7

doi: 10.1093/gigascience/giy167
Advance Access Publication Date: 29 January 2019
Technical Note

TECHNICAL NOTE

SMAGEXP: a galaxy tool suite for transcriptomics data meta-analysis

Samuel Blanck [1,*] and Guillemette Marot [1,2]

[1]Univ. Lille, CHU Lille , EA 2694 CERIM, 1 place de Verdun, F-59000 Lille, France and [2]Inria Lille-Nord Europe, MODAL, 40 avenue Halley, 59650 Villeneuve d'Ascq , France

*Correspondence address. Samuel Blanck, CERIM, 1 place de Verdun, 59000 Lille, France E-mail: samuel.blanck@univ-lille.fr http://orcid.org/0000-0002-7868-2844

Abstract

Background: With the proliferation of available microarray and high-throughput sequencing experiments in the public domain, the use of meta-analysis methods increases. In these experiments, where the sample size is often limited, meta-analysis offers the possibility to considerably enhance the statistical power and give more accurate results. For those purposes, it combines either effect sizes or results of single studies in an appropriate manner. R packages metaMA and metaRNASeq perform meta-analysis on microarray and next generation sequencing (NGS) data, respectively. They are not interchangeable as they rely on statistical modeling specific to each technology. **Results:** SMAGEXP (Statistical Meta-Analysis for Gene EXPression) integrates metaMA and metaRNAseq packages into Galaxy. We aim to propose a unified way to carry out meta-analysis of gene expression data, while taking care of their specificities. We have developed this tool suite to analyze microarray data from the Gene Expression Omnibus database or custom data from Affymetrix© microarrays. These data are then combined to carry out meta-analysis using metaMA package. SMAGEXP also offers to combine raw read counts from NGS experiments using DESeq2 and metaRNASeq package. In both cases, key values, independent from the technology type, are reported to judge the quality of the meta-analysis. These tools are available on the Galaxy main tool shed. A dockerized instance of galaxy containing SMAGEXP and its dependencies is available on Docker hub. Source code, help, and installation instructions are available on GitHub. **Conclusion:** The use of Galaxy offers an easy-to-use gene expression meta-analysis tool suite based on the metaMA and metaRNASeq packages.

Keywords: galaxy; transcriptomics; microarray; RNA-seq; meta-analysis

Background

Meta-analyses are widely used in medicine and health policy to increase statistical power in studies suffering from small sample sizes. Gene expression experiments are a typical example of such designs. The R packages metaMA and metaRNASeq are dedicated to gene expression microarray and next-generation sequencing (NGS) meta-analysis, respectively. While metaMA and metaRNASeq are open source and available on CRAN, they require coding skills in R to perform meta-analysis. Thus, to facilitate the use and the dissemination of these packages, we developed Galaxy wrappers. Galaxy [1–3] is an open, web-based platform for data-intensive biomedical research. It keeps tracks of history, and all analyses can be rerun. The Galaxy community is very active, and numerous bioinformatics tools are included in Galaxy thanks to a modular system based on XML wrappers. These integrated tools can be shared via the Galaxy toolshed, which serves as an app store.

Methods

Overview of R packages integrated into Galaxy

metaMA

Gene expression microarray data meta-analysis can be performed thanks to the metaMA [4] R package. It proposes meth-

Received: 1 March 2018; **Revised:** 27 June 2018; **Accepted:** 20 December 2018

Table 1: Summary of tool inputs and outputs

Tool	Input	Output
GEOQuery	Gene Expression Omnibus database ID	rdata object and .cond file
QCNormalization	Raw .CEL Affymetrix© files	rdata object and plots
Import custom data	Expression data in tabular text format	rdata object and plots
Limma analysis	rdata object from GEOQuery or QCNormalization or Import custom data and .cond file	rdata Object, HTML report and results text file
Microarray data meta-analysis	rdata objects from Limma analyses	HTML report
Recount	Recount accession ID	One count file per sample
RNA-seq data meta-analysis	Results text files from galaxy DESeq2 tool	HTML report

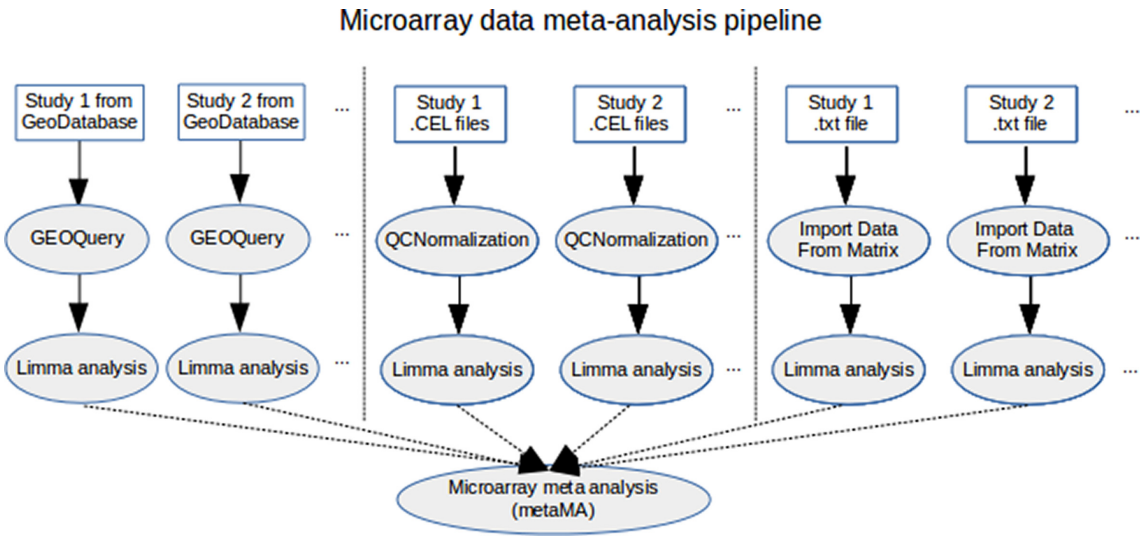


Figure 1: Overview of the tools from microarray data meta-analysis pipeline integrated within Galaxy.

ods to combine either *P* values or moderated effect sizes from different studies to find differentially expressed (DE) genes. In our pipeline we only keep the inverse normal method [5] to combine the *P* values calculated by limma [6] for each single study.

metaRNAseq

RNA sequencing (RNA-seq) data meta-analysis can be performed thanks to the metaRNASeq [7] R package. It implements two *P* value combination techniques: the inverse normal and Fisher methods [8]. Single study *P* values are computed with DESeq2 [9].

Differences between metaMA and metaRNASeq

Main differences come from the statistical distributions used to model data and from the manner to treat the genes exhibiting conflicting expression patterns (i.e., under-expression when comparing one condition to another in one study, and over-expression for the same comparison in another study). Usually, microarray data are modeled by Gaussian distributions, while NGS data are modeled by negative binomial distributions. As explained in [4] and [7], the trick to use one-tailed *P* values for each single study before combination in metaMA avoids directional conflicts. In metaRNASeq, this trick cannot be used, which necessitates a *post hoc* identification of conflicts, a step that is also proposed in metaRNASeq.

Description of Galaxy tools

The SMAGEXP tool suite offers two distinct gene expression meta-analysis functionalities: one dedicated to microarray data meta-analysis and one dedicated to RNA-seq data meta-analysis (see Table 1 and Fig. 1).

Microarray data meta-analysis

GEOQuery tool. GEOQuery tool fetches microarray data directly from Gene Expression Omnibus (GEO) database [10], based on the GEOQuery [11] bioconductor [12] R package. Given a GSE accession ID, it returns an rdata object containing the data and a text file (.cond file, see Fig. 2) summarizing the conditions of the experiment. The .cond file is a text file containing one line per sample in the experiment. Each line is made of 3 columns:

- Sample ID
- Condition of the biological sample
- Description of the biological sample

Column names are optional, and only the columns order matters. As the GEO dataset should already have been normalized, the GEOQuery tool does not perform any normalization method, apart from an optional log2 transformation.

QCNormalization tool. It is possible to analyze .CEL files from Affymetrix© gene expression microarray. The QCnormalization tool offers to ensure the quality of the data and to normalize them. Several normalization methods are available:

```
GSM342582.CEL   tumor   GSM342582_Tongue_040
GSM342583.CEL   normal  GSM342583_Tongue_041
GSM342584.CEL   tumor   GSM342584_Tongue_041
GSM342585.CEL   normal  GSM342585_Tongue_042
GSM342586.CEL   tumor   GSM342586_Tongue_042
GSM342587.CEL   normal  GSM342587_Tongue_043
```

Figure 2: Example of .cond file.

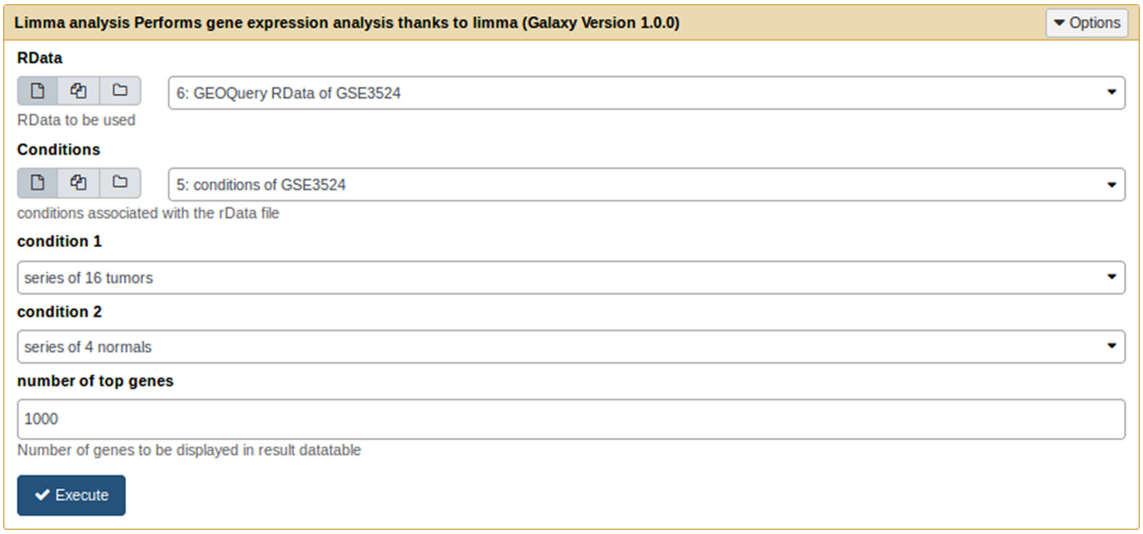


Figure 3: limma analysis tool form.

- rma normalization
- quantile normalization + log2
- background correction + log2
- log2 only

This tool generates several quality figures: microarray images, box plots, and MA plots. It also outputs an rdata object containing the normalized data for further analysis with the limma analysis tool.

Import custom data tool. This tool imports data stored in a tabular text file. A few normalization methods are proposed, but it is possible to skip the normalization step by choosing "none" in the normalization methods options. Therefore, this tool is of special interest when the input dataset has been previously normalized. This tool also generates box plots and MA plots and outputs an rdata object containing the data for further analysis with the limma analysis tool.

Limma analysis tool. The Limma analysis tool performs single analysis either of data previously retrieved from the GEO database or normalized Affymetrix© .CEL files data. Given a .cond file, it runs a standard limma differential expression analysis. The user choose two conditions extracted from the .cond file (see Fig. 3). It generates box plots for rough quality control of normalization, *P* value histograms to ensure that statistical hypotheses are not violated, and a volcano plot to quickly identify the most meaningful changes. This tool also outputs a table summarizing the DE genes and their annotations. Genes are sorted by ascending Benjamini-Hochberg adjusted *P* value, and annotations are retrieved via GEO database. This list of genes can be exported to excel or to csv format. This table is sortable and requestable. Furthermore, it is possible to expand each row to display extended annotation information, including hypertext links to the National Center for Biotechnology Information (NCBI) gene database. Finally, this tool outputs an rdata object to perform further meta-analysis and a text file containing annotated results of the differential analysis.

Microarray data meta-analysis tool. The meta-analysis relies on the metaMA R package. Prior to the meta-analysis itself, a preprocessing is made in order to ensure compatibility between several sources of data. In fact, data could come from different types of microarrays. First, we list the Entrez gene ID corresponding to each probe of each dataset. Next, we keep the probes corresponding to the genes that are shared by all the experiments of the meta-analysis. Then, for each dataset, we merge the microarray probes originating from the same Entrez gene ID by computing their mean. Note that the merging of different technologies induces a loss of information and might generate several conflicts as probes do not necessarily reflect the same biological reality. Finally, the *P* value combination method of metaMA is run on the merged dataset. It generates a Venn diagram (if the number of studies is lower than 3) or a UpSet diagram [13] (if the number of studies is greater than 4) summarizing the results of the meta-analysis, and a list of indicators to evaluate the quality of the performance of the meta-analysis:

- DE (differentially expressed): number of DE genes
- IDD (integration-driven discoveries): number of genes that are declared DE in the meta-analysis that were not identified in any of the single studies alone
- Loss: number of genes that are identified DE in single studies but not in meta-analysis

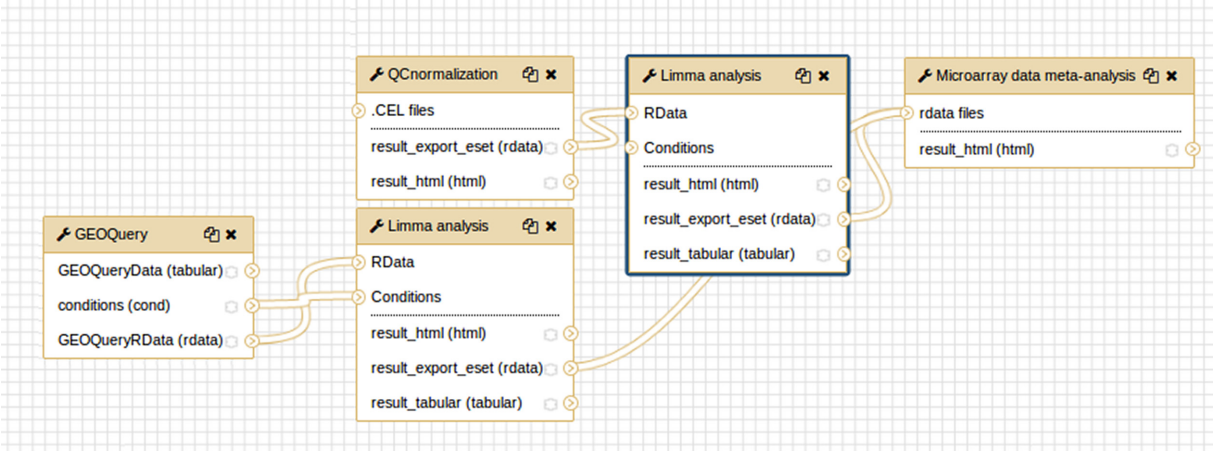


Figure 4: Exemple of a galaxy workflow for microarray meta-analysis.

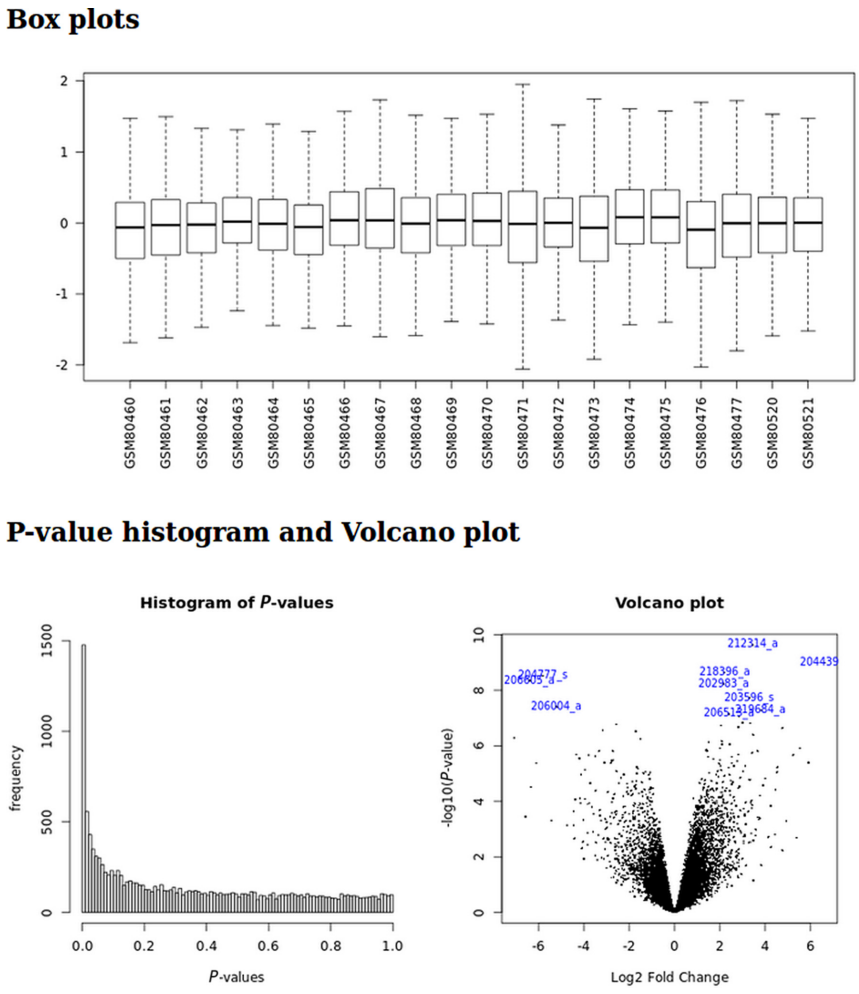


Figure 5: limma analysis tool output plots.

- IDR (integration-driven discovery rate): corresponding proportion of IDD
- IRR (integration-driven revision): corresponding proportion of loss

It also outputs a fully sortable and requestable table, with gene annotations and hypertext links to NCBI gene database.

RNA-seq data meta-analysis

Recount tool. The recount tool fetches data from the recount2 project database [14]. The recount Galaxy tool relies on the bio-

Copy CSV Excel Search:

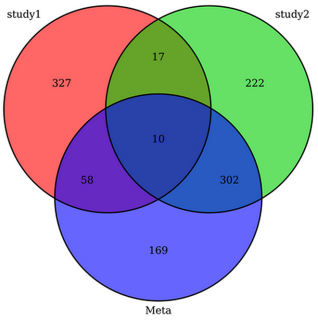
	ID	adj_P_Val	P_Value	t	B	logFC	Gene_symbol	Gene_title	Gene_ID	Chromosome_annota...	GO_Function_ID
⊕	212314_at	3.3e-06	2.3e-10	-11.19	13.3288	-3.46	SEL1L3	sel-1 suppressor of lin...	23231	Chromosome 4, NC_00...	
⊕	204439_at	7.1e-06	1.0e-09	-10.31	12.0465	-6.64	IFI44L	interferon-induced pr...	10964	Chromosome 1, NC_00...	
⊕	218396_at	1.0e-05	2.3e-09	-9.85	11.3259	-2.20	VPS13C	vacuolar protein sorti...	54832	Chromosome 15, NC_0...	
⊖	204777_s_at	1.0e-05	2.9e-09	9.71	11.1038	5.81	MAL	mal, T-cell differentiat...	4118	Chromosome 2, NC_00...	GO:0015267///GO:000...
	Gene Symbol: MAL										
	Gene Title: mal, T-cell differentiation protein										
	GO Function ID: GO:0015267, GO:0008289, GO:0016505, GO:0005515, GO:0019911,										
⊕	206605_at	1.3e-05	4.5e-09	9.47	10.7170	6.39	ENDOU	endonuclease, polyU-s...	8909	Chromosome 12, NC_0...	GO:0003723///GO:000...
⊕	202983_at	1.4e-05	6.0e-09	-9.32	10.4674	-2.17	HLTF	helicase-like transcrip...	6596	Chromosome 3, NC_00...	GO:0005524///GO:001...
⊕	203596_s_at	3.9e-05	1.9e-08	-8.70	9.4094	-3.31	IFIT5	interferon-induced pr...	24138	Chromosome 10, NC_0...	GO:0003723///GO:004...
⊕	206004_at	7.0e-05	4.0e-08	8.33	8.7509	5.21	TGM3	transglutaminase 3	7053	Chromosome 20, NC_0...	GO:0005509///GO:000...
⊕	219684_at	8.3e-05	5.3e-08	-8.18	8.4819	-3.80	RTP4	receptor (chemosenso...	64108	Chromosome 3, NC_00...	GO:0005515
⊕	206513_at	1.0e-04	7.1e-08	-8.04	8.2131	-2.39	AIM2	absent in melanoma 2	9447	Chromosome 1, NC_00...	GO:0003690///GO:004...

Show 10 entries

Showing 1 to 10 of 1,000 entries Previous 1 2 3 4 5 ... 100 Next

Figure 6: limma analysis tool: table of top 10 genes for GSE3524 dataset.

Venn diagram



Summary

DE	IDD	Loss	IDR	IRR
539	169	566	31.35	60.47

DE : Number of differentially expressed genes
IDD (Integration Driven discoveries) : number of genes that are declared DE in the meta-analysis that were not identified in any of the individual studies alone
Loss : Number of genes that are identified DE in individual studies but not in meta-analysis
IDR (Integration-driven Discovery Rate) : corresponding proportions of IDD
IRR (Integration-driven Revision) : corresponding proportions of Loss

Figure 7: Venn diagram and summary of microarray data meta-analysis tool results.

conductor R package recount. Given the accession ID of an experiment, it generates one count file per sample of the experiment. Then these files can be analyzed by the Galaxy DESeq2 tool.

RNA-seq data meta-analysis tool. The RNA-seq data meta-analysis tool relies on the DESeq2 galaxy tool analysis results. Given several text files resulting from the DESeq2 [9] tool, the metaRNAseq tool performs a meta-analysis, generates the list of DE genes, and outputs the DE, IDD, loss, IDR, and IRR indicators.

Application

Microarray meta-analysis example

SMAGEXP was applied to two GEO datasets identified with the following IDs: GSE3524 [15] and GSE13601 [16]. These two datasets contain human oral squamous cell carcinoma (SCC) data. See Fig. 4 for an overview of the worfklow of this analysis.

First, we fetch data from the GSE3524 using the GEOQuery tool (with parameter "log2 transformation" = auto). Then, we launch the limma analysis, using the output from the GEOquery tool. It generates an rdata output that will be useful for the meta-analysis. Results can be seen in Figs. 5 and 6

Secondly, the same kind of analysis is run from raw .CEL files. We choose to keep six .CEL files from the GSE13601 dataset (IDs from GSM342582 to GSM342587). Quality control and normalization are done thanks to the QCnormalization tool. Then, as previously, the limma analysis tool is run to generate an HTML report and an rdata output.

Run a metaMA analysis

To run the microarray meta-analysis tool, we only need the rdata output of each single study, generated by the limma analysis tool. It generates a Venn diagram or an UpSet plot (when the number of studies is greater than 3) to compare the results of each study with the meta-analysis. It also outputs several indicators as described in the description of the tool (see Fig. 7). As for the limma tool, annotated expressed genes are displayed in a table that can be ordered and requested.

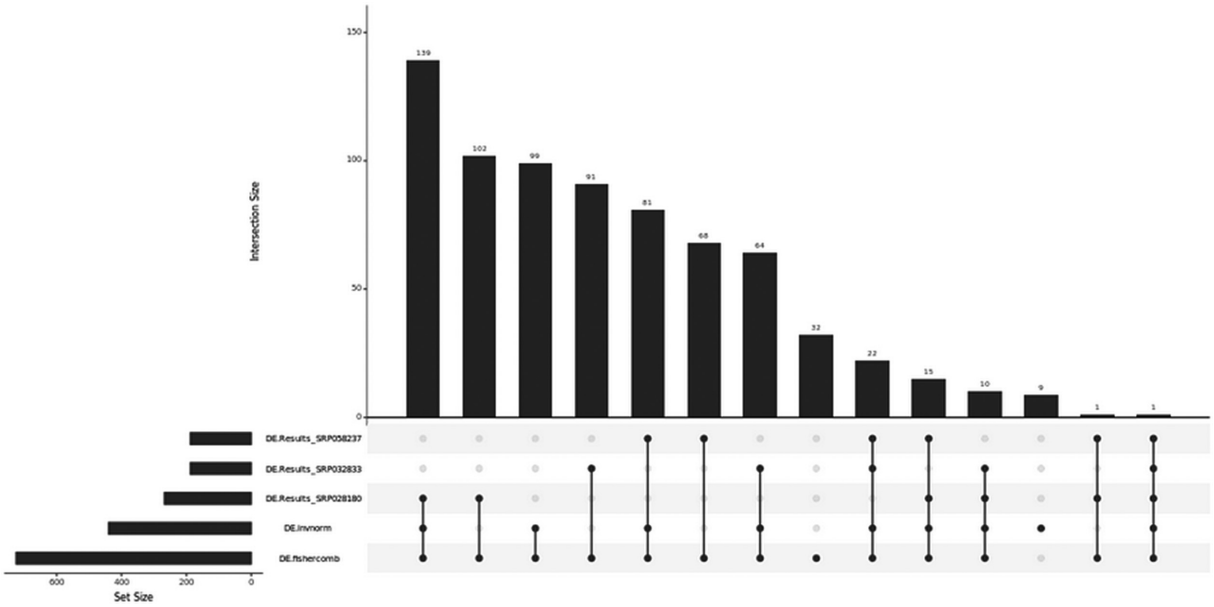


Figure 8: UpSet plot for the RNA-seq datasets SRP032833, SRP028180, and SRP058237.

1	2	3	4	5	6	7	8	9	10
"ID"	"DE.DE.Results_SRP058237"	"DE.DE.Results_SRP028180"	"DE.DE.Results_SRP032833"	"DE.DE.fishercomb"	"DE.DE.invnorm"	"FC.Results_SRP058237"	"FC.Results_SRP028180"	"FC.Results_SRP032833"	"signFC"
"ENSG00000000003.14"	0	0	0	0	0	0.455146164961458	1.33913280471823	-0.0888398117666611	0
"ENSG00000000005.5"	NA	NA	0	0	0	NA	NA	0.536693942861224	NA
"ENSG00000000419.12"	0	0	0	0	0	-0.48309992748253	2.17636688954897	-0.507227286832749	0
"ENSG00000000457.13"	0	0	0	0	0	0.242290545493709	0.442632029805506	0.198830840644203	1
"ENSG00000000460.16"	0	0	0	0	0	0.451776441815323	0.0664236691132769	0.5526767606192	1
"ENSG00000000938.12"	0	0	1	1	1	-1.03666017111835	3.41945634684131	-1.75654716798619	0
"ENSG00000000971.15"	0	0	NA	0	0	0.697621595674714	-0.467897685456305	-0.134173901933097	0
"ENSG00000001036.13"	0	0	0	0	0	0.23017159092455	0.224430876523283	-0.384266125346025	0
"ENSG00000001084.10"	0	0	0	0	0	0.147430824479954	-0.0924241682878958	-0.199143617451887	0
"ENSG00000001167.14"	0	0	0	0	0	-0.557029894743996	1.89195667645183	-0.285766330239775	0

Figure 9: Header of a metaRNAseq results file.

RNA-seq data meta-analysis example

SMAGEXP was applied to three Recount2 datasets identified with the following IDs: SRP032833 [17], SRP028180 [18], and SRP058237 [19]. These three datasets contain human lung SCC data. We first fetch data from these datasets with the recount galaxy tool. Then, thanks to the Galaxy DESeq2 tool, we launch differential analysis on the following contrasts: invasive vs normal for SRP032833 dataset, tumor vs normal for SRP028180 dataset, and tumor vs adjacent for SRP058237 dataset.

Run a metaRNAseq analysis

The RNA-seq data meta-analysis tool relies on DESeq2 results

It outputs a Venn diagram or an UpSet plot (if the number of studies is greater than 3, see Fig. 8) and the same indicators as in the microarray data analysis tool for both Fisher and inverse normal *P* values combinations. It also generates a text file containing summarization of the results of each single analysis and meta-analysis. Potential conflicts between single analysis are indicated by zero values in the "signFC" column (see Fig. 9).

Conclusion

We developed SMAGEXP, a tool suite dedicated to gene-expression data meta-analysis. This tool suite proposes quality controls, single analyses, and meta-analyses of microarray and RNA-seq data, suggesting appropriate pipelines for each type of data. It delivers fully annotated results of differentially DE genes, exportable in several usual formats. Integrated into Galaxy, SMAGEXP is easy to use for biologists and life scientists. R packages metaMA and metaRNAseq thus inherit reproducibility and accessibility support from Galaxy. Furthermore, thanks to Docker, we made these Galaxy tools and their dependencies easy to deploy.

Availability of source code and requirements

- Project name: SMAGEXP
- Project home page: https://github.com/sblanck/smagexp [20]
- Operating system(s): Linux (Galaxy); platform independent for Galaxy's browser-based user interface.
- Programming language: R
- Other requirements: Galaxy, Docker [21]
- License: MIT license
- Any restrictions to use by non-academics: None
- SciCrunch.org RRID:SCR_016360

SMAGEXP is available on the Galaxy main toolshed [22]. Furthermore, a fully dockerized instance of Galaxy containing SMAGEXP and DESeq2 is available at: https://hub.docker.com/r/sblanck/galaxy-smagexp/.

Availability of supporting data

The datasets supporting the microarray meta-analysis example presented here are available in the GEO database. Their acces-

sion IDs are GSE3524 and GSE13601. The datasets supporting the RNA-seq meta-analysis example presented here are available on Recount2. Their accession IDs are SRP032833, SRP028180, and SRP058237

Documentation, step-by-step tutorials, examples, galaxy histories, and workflow presented here are available on GitHub: https://github.com/sblanck/smagexp/tree/master/examples.

Code snapshots and input data are available from the GigaScience GigaDB repository [23].

Abbreviations

DE, differentially expressed; GEO, Gene Expression Omnibus; IDD, integration-driven discoveries; IDR, integration-driven discovery rate; IRR, integration-driven revision; NCBI, National Center for Biotechnology Information; NGS, next-generation sequencing; RNA-seq, RNA sequencing; SCC, squamous cell carcinoma; SMAGEXP, Statistical Meta-Analysis for Gene EXPression.

Competing interests

The authors declare that they have no competing interests.

Author contributions

The project was initiated by G.M. who developed metaMA and metaRNASeq R packages. The galaxy tools were developed, installed, and documented by S.B. and tested by S.B. and G.M. The article was written by S.B. and G.M. Both authors read and approved the final manuscript.

Acknowledgement

This project was supported by University of Lille and Inria Lille-Nord Europe and by CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015-2020

References

1. Goecks J, Nekrutenko A, Taylor J, et al. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome Biol 2010; **11**(8): R86.
2. Blankenberg D, Kuster GV, Coraor N, et al. Galaxy: a web-based genome analysis tool for experimentalists. Current Protocols in Molecular Biology 2010; 19–10.
3. Giardine B, Riemer C, Hardison RC, et al. Galaxy: a platform for interactive large-scale genome analysis. Genome Research 2005; **15**(10): 1451–1455.
4. Marot G, Foulley JL, Mayer CD, et al. Moderated effect size and P-value combinations for microarray meta-analyses. Bioinformatics 2009; **25**(20): 2692–2699.
5. Hedges L, Olkin I. Statistical Methods for Meta-Analysis. London: Academic Press; 1985.
6. Ritchie ME, Phipson B, Wu D, et al. limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Research 2015; **43**(7): e47.
7. Rau A, Marot G, Jaffrézic F.. Differential meta-analysis of RNA-seq data from multiple studies. BMC Bioinformatics 2014; **15**(1): 1–10.
8. Fisher RA. Statistical Methods for Research Workers. Edinburgh: Oliver and Boyd; 1932.
9. Love MI, Huber W, Anders S.. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology 2014; **15**: 550.
10. Edgar R, Domrachev M, Lash AE.. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Res 2002; **30**(1): 207–210.
11. Davis S, Meltzer P. GEOquery: a bridge between the Gene Expression Omnibus (GEO) and BioConductor. Bioinformatics 2007; **14**: 1846–1847.
12. Huber W, Carey JV, Gentleman R, et al. Orchestrating high-throughput genomic analysis with Bioconductor. Nature Methods 2015; **12**(2): 115–121.
13. Conway JR, Lex A, Gehlenborg N. UpSetR: an R package for the visualization of intersecting sets and their properties. Bioinformatics 2017; **33**(18): 2938–2940.
14. Collado-Torres L, Nellore A, Kammers K, et al. Reproducible RNA-seq analysis using recount2. Nature Biotechnology 2017, 35(2), 319–321, doi:10.1038/nbt.3838.html.
15. Toruner GA, Ulger C, Alkan M, et al. Association between gene expression profile and tumor invasion in oral squamous cell carcinoma. Cancer Genet Cytogenet 2004; **154**(1): 27–35.
16. Estilo CL, O-charoenrat P, Talbot S, et al. Oral tongue cancer gene expression profiling: identification of novel potential prognosticators by oligonucleotide microarray analysis. BMC Cancer 2009; **9**: 11.
17. Morton ML, Bai X, Merry CR, et al. Identification of mRNAs and lincRNAs associated with lung cancer progression using next-generation RNA sequencing from laser micro-dissected archival FFPE tissue specimens. Lung Cancer 2014; **85**(1): 31–39.
18. Ooi AT, Gower AC, Zhang KX, et al. Molecular profiling of premalignant lesions in lung squamous cell carcinomas identifies mechanisms involved in stepwise carcinogenesis. Cancer Prev Res (Phila) 2014; **7**(5): 487–495.
19. Durrans A, Gao D, Gupta R, et al. Identification of reprogrammed myeloid cell transcriptomes in NSCLC. PloS One 2015; **10**(6): 1–22.
20. SMAGEXP; https://github.com/sblanck/smagexp
21. Galaxy; https://galaxyproject.org/, Access date 17/01/2019.
22. Blankenberg D, Von Kuster G, Bouvier E, et al. Dissemination of scientific software with Galaxy ToolShed. Genome Biology 2014; **15**(2): 1–3.
23. Blanck S, Marot G. Supporting data for “SMAGEXP: a galaxy tool suite for transcriptomics data meta-analysis.” GigaScience Database. 2018. http://dx.doi.org/10.5524/100541

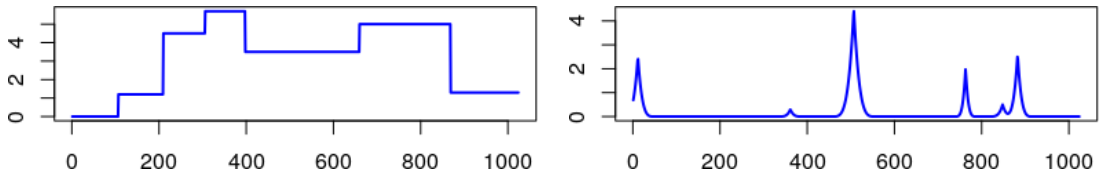
Chapter 3

De la classification non supervisée de profils génomiques à la classification supervisée de patients avec sélection de variables

3.1 Classification non supervisée de courbes

Le travail statistique décrit dans l'article de Biometrics ci-après a d'abord été motivé par l'analyse de profils génomiques pour le cancer. L'idée était de faire des sous-groupes de patients présentant les mêmes marqueurs moléculaires, afin de personnaliser les traitements en fonction des différentes formes. Devant le nombre de mesures disponibles le long du génome, il est possible de considérer chaque profil génomique comme une courbe et d'utiliser des techniques d'analyse fonctionnelle, où les unités de base sont des courbes. L'originalité de l'article [Giacofci et al., 2013] vient de l'inclusion d'un effet aléatoire dans la classification de courbes pour modéliser une variabilité individuelle. Ma principale contribution a été l'implémentation de cette nouvelle approche de classification et la mise en place de simulations. J'ai essentiellement travaillé sur deux des quatre types de profils mentionnés dans la table 1 de l'article, à savoir "blocks" et "bumps", décrits initialement dans [Donoho and Johnstone, 1994] et représentés dans la figure 3.1.

Figure 3.1: Exemples de profils étudiés



$\mu^{\texttt{Blocks}}(t) = \sum_r \frac{h_r}{2}\left(1 + sgn(t - v_r)\right)$ $\quad$ $\mu^{\texttt{Bumps}}(t) = \sum_r h_r / \left(1 + \frac{|t-v_r|}{w_r}\right)^4$

avec t la position dans le signal ($t \in [0, 1]$), v_r la localisation des points de rupture, h_r les hauteurs des sauts et w_r les largeurs des pics.

Biometrics 69, 31–40
March 2013
DOI: 10.1111/j.1541-0420.2012.01828.x

Wavelet-Based Clustering for Mixed-Effects Functional Models in High Dimension

M. Giacofci,[1,*] S. Lambert-Lacroix,[2,] G. Marot,[3,4,5,***] and F. Picard[6,****]**

[1]Laboratoire LJK, BP 53, Université de Grenoble et CNRS, 38041 Grenoble cedex 9, France
[2]UJF-Grenoble 1/CNRS/UPMF/TIMC-IMAG UMR 5525, Grenoble, F-38041, France
[3]Projet BAMBOO, INRIA Rhône-Alpes, F-38330 Montbonnot Saint-Martin, France
[4]Biostatistics, EA 2694, UDSL, Université Lille Nord de France
[5]MODAL, INRIA Lille Nord Europe, F-59650 Villeneuve d'Ascq, France
[6]LBBE, UMR CNRS 5558 Université Lyon 1, F-69622, Villeurbanne, France
[*]*email:* madison.giacofci@imag.fr
[**]*email:* sophie.lambert@imag.fr
[***]*email:* guillemette.marot@univ-lille2.fr
[****]*email:* franck.picard@univ-lyon1.fr

Summary. We propose a method for high-dimensional curve clustering in the presence of interindividual variability. Curve clustering has longly been studied especially using splines to account for functional random effects. However, splines are not appropriate when dealing with high-dimensional data and can not be used to model irregular curves such as peak-like data. Our method is based on a wavelet decomposition of the signal for both fixed and random effects. We propose an efficient dimension reduction step based on wavelet thresholding adapted to multiple curves and using an appropriate structure for the random effect variance, we ensure that both fixed and random effects lie in the same functional space even when dealing with irregular functions that belong to Besov spaces. In the wavelet domain our model resumes to a linear mixed-effects model that can be used for a model-based clustering algorithm and for which we develop an EM-algorithm for maximum likelihood estimation. The properties of the overall procedure are validated by an extensive simulation study. Then, we illustrate our method on mass spectrometry data and we propose an original application of functional data analysis on microarray comparative genomic hybridization (CGH) data. Our procedure is available through the R package `curvclust` which is the first publicly available package that performs curve clustering with random effects in the high dimensional framework (available on the CRAN).

Key words: Clustering; Functional data; Mixed models; Wavelets.

1. Introduction

Functional data analysis has gained increased attention in the past years, in particular in high-throughput biology with the use of mass spectrometry. This method is used to characterize the protein content of biological samples by separating compounds according to their mass to charge ratio (m/z). Among different technologies matrix assisted laser desorption and ionization, time-of-flight (MALDI-TOF) mass spectrometry is one the most used and has become standard to improve proteomic profiling of diseases as well as clinical diagnosis.

Dedicated methods have been developed to analyze such data for differential analysis, supervised classification and clustering (Hilario et al. 2006). Up to now the functional setting has mostly been developed for differential analysis (Morris et al. 2008). One central element is the modeling of the interindividual variability by using functional random effects, because subject-specific fluctuations are known to be the largest source of variability in mass-spec data (Eckel-Passow et al. 2009). In this article, we focus on the nonsupervised task which consists in finding groups of individuals whose proteomic landscape is similar. Surprisingly the clustering task received less attention, and is mainly based on hierarchical clustering on the set of peaks detected across spectra (Bensmail et al. 2005; Morris et al. 2010). However, such method is known to depend heavily on the peak detection method and has the strong disadvantage to neglect the interindividual variability whereas this information should be central for subgroup discovery. Thus, our main focus in this article is modeling and clustering curves of this type in a functional mixed model framework.

When dealing with curve clustering in the presence of individual variability, a pioneer work is based on a spline decomposition of the signal (James and Sugar 2003) which resumes to a linear mixed effect model on which clustering and low-dimensional representation can be performed. However, splines show two main drawbacks: (*i*) they are inappropriate when dealing with functions that show peaks and irregularities, (*ii*) they require heavy computational efforts and so are not adapted to high dimensional data. On the contrary,

wavelet representations appear to be a natural framework to consider such irregularities through the sequence space of (usually sparse) Besov representation. Recent works have been done about estimation and inference in the functional mixed effects framework based on a wavelet decomposition approach. A fully Bayesian version has been proposed by Morris and Carroll (2006), with nonparametric estimates of fixed and random effects as well as between and within-curve covariance matrix estimates to accomodate a wide variety of correlation structures. In addition, Antoniadis and Sapatinas (2007) propose a study of both estimation and inference in a frequentist framework. In this article, we use a wavelet representation for both fixed and random effects to perform model-based clustering. Such strategy has been considered by Antoniadis, Bigot, and von Sachs (2008) and by Ray and Mallick (2006) without random effects for image clustering and for the analysis of time course experiments respectively. We use a similar approach and we extend it by adding functional random effects. Interindividual variability in the wavelet domain is modeled using results of Antoniadis and Sapatinas (2007) but accomodates a broader range of correlation structure. In particular we allow within curve correlation to vary over groups and positions. Then we propose a two-step procedure which involves a dimension reduction step and a clustering step based on the EM-algorithm. We also propose a model-selection criterion that accounts for the interindividual variability, and we define a rigorous simulation framework for curve clustering. Our method is implemented within the `R` package **`curvclust`** (available on the CRAN), which is the first available software dedicated to this task. In a first application, we illustrate our method on the mass spectrometry data first published in Petricoin et al. (2002).

Then our last contribution is to extend the use of functional models to another type of high throughput data which are comparative genomic hybridization (CGH) data. The CGH array technology is used to map copy number imbalances between genomes by hybridizing differentially labeled genomic DNAs on a chip. Fluorescence ratios are usually analyzed using change-point models to detect segments that correspond to homogeneous regions on the genome in terms of copy number. Clustering patients based on their CGH profiles is very promising and has been successfully used to identify molecular subtypes of cancer. However, clustering CGH profiles based on a segmentation has the same drawbacks that clustering mass spectra based on detected peaks: results depend on the segmentation methods. Moreover the interindividual variability has never been investigated in this type of data, whereas it is likely to represent an important part of the variability of the data especially for cancer profiles. We use the breast cancer data of Fridlyand et al. (2006) that have already been analyzed for nonsupervised clustering by Van Wieringen et al. (2008). We show the interest of functional random effects for these type of data and we discuss the impacts in terms of analysis and design for copy number studies.

2. Functional Clustering Modeling using Wavelets

2.1 *Presentation of the Model*

We observe N curves $Y_i(t)$ over M equally spaced time points $\mathbf{t} = (t_1, \ldots, t_M)$ with $t_j \in [0,1]$ for $j \in [1, M]$, and $M = 2^J$ for some integer J. In the functional clustering setting we suppose that individuals are spread among L unknown clusters of prior size π_ℓ, $\ell = 1, \ldots, L$, and we denote by $\zeta_{i\ell}$ the indicator variable that equals 1 if the ith individual is in the ℓth group. Then, we consider the linear functional model such that given $\{\zeta_{i\ell} = 1\}$, $Y_i(t) = \mu_\ell(t) + E_i(t)$, where $\mu_\ell(t)$ is the principal functional fixed effect that characterizes cluster ℓ, $E_i(t)$ is a zero mean Gaussian process with covariance kernel $\mathrm{cov}(E_i(t), E_i(t')) = \sigma_E^2 \delta_{tt'}$, where $\delta_{tt'}$ stands for the kronecker product. In the following, we will use notations $\mathbf{Y}_i(\mathbf{t}) = (Y_i(t_1), \ldots, Y_i(t_M))^T$, $\mu_\ell(\mathbf{t}) = (\mu_\ell(t_1), \ldots, \mu_\ell(t_M))^T$ and $\mathbf{E}_i(\mathbf{t}) = (E_i(t_1), \ldots, E_i(t_M))^T$. To handle subject-specific random deviations from the cluster average curve we introduce random functions $U_i(t)$ that are modeled as centered Gaussian processes with kernel $K_\ell(t,t') = \mathrm{cov}(U_i(t), U_i(t'))$ (given $\{\zeta_{i\ell} = 1\}$), not necessarily stationary, but independent from $E_i(t)$. Then given $\{\zeta_{i\ell} = 1\}$, the previous model becomes $Y_i(t) = \mu_\ell(t) + U_i(t) + E_i(t)$ (2.1). Once defined in the functional domain, a classical approach is to convert the original infinite-dimensional clustering problem into a finite-dimensional problem using a functional basis representation of the model. At this step James and Sugar (2003) propose a spline-based representation of model (2.1) with individuals observed at sparse sets of time points like in longitudinal data. Our procedure is more adapted to high dimensional data thanks to the computational efficiency of wavelets, unlike splines that require matrix inversions whose complexity increases with the density of the design. Moreover, as we will see below, the wavelet representation allows us to account for a wider range of functional shapes than splines, thanks to their connection with Besov spaces. Using a wavelet representation of this model allows us to characterize different types of smoothness conditions assumed on the response curves $Y_i(t)$ by the mean of their wavelet coefficients. Moreover, wavelet representations are sparse for a wide variety of functional spaces, which is crucial when dealing with high dimensional data. This property will be central while performing dimension reduction. Briefly, we are working with a dyadic orthonormal wavelet basis $\{\phi_{j_0 k}(t), k = 0, 1, \ldots 2^{j_0} - 1; \psi_{jk}(t), j \geq j_0, k = 0, \ldots, 2^j - 1\}$ generated from a father wavelet ϕ and a mother wavelet ψ of regularity r, $(r \geq 0)$. In this basis $Y_i(t)$ has the following decomposition: $Y_i(t) = \sum_{k=0}^{2^{j_0}-1} c^*_{i,j_0 k} \phi_{j_0 k}(t) + \sum_{j \geq j_0} \sum_{k=0}^{2^j - 1} d^*_{i,jk} \psi_{jk}(t)$. In practice we use the discrete wavelet transform (DWT) which can be performed thanks to Mallat's fast algorithm with $\mathcal{O}(M)$ operations only. We denote by $\mathbf{W}$ the $[M \times M]$-matrix containing filters of the chosen wavelet basis. The resulting scaling and wavelet coefficients $\mathbf{c}_i = (c_{i,j_0 k})_{k=0\ldots 2^{j_0}-1}$ and $\mathbf{d}_i = (d_{i,jk})_{j=j_0\ldots J-1}^{k=0\ldots 2^j-1}$ of the individual curves are empirical discrete coefficients. They are related to their theoretical continuous counterparts $c^*_{i,j_0 k}$ and $d^*_{i,jk}$ by: $c_{i,j_0 k} \approx \sqrt{M} c^*_{i,j_0 k}$ and $d_{i,jk} \approx \sqrt{M} d^*_{i,jk}$. In the following, we denote by $\alpha_\ell = (\alpha_{\ell,j_0 k})_{k=0\ldots 2^{j_0}-1}$ and $\beta_\ell = (\beta_{\ell,jk})_{j=j_0\ldots J-1}^{k=0\ldots 2^j-1}$ the $[2^{j_0} \times 1]$ and $[(M - 2^{j_0}) \times 1]$ vectors of scaling and wavelet coefficients of $\mu_\ell(\mathbf{t})$, and we denote by $\nu_i = (\nu_{i,j_0 k})_{k=0\ldots 2^{j_0}-1}$ and $\theta_i = (\theta_{i,jk})_{j=j_0\ldots J-1}^{k=0\ldots 2^j-1}$ the $[2^{j_0} \times 1]$ and $[(M - 2^{j_0}) \times 1]$ vectors of scaling and wavelet random coefficients of $\mathbf{U}_i(\mathbf{t}) = (U_i(t_1), \ldots, U_i(t_M))^T$. We apply the DWT to model (2.1) such

that $\mathbf{WY}_i(\mathbf{t}) = \mathbf{W}\mu_\ell(\mathbf{t}) + \mathbf{WU}_i(\mathbf{t}) + \mathbf{WE}_i(\mathbf{t})$, and in the coefficients domain our model resumes to a linear mixed-effect model, such that given $\{\zeta_{i\ell} = 1\}$, $(\mathbf{c}_i^T, \mathbf{d}_i^T)^T = (\alpha_\ell^T, \beta_\ell^T)^T + (\nu_i^T, \theta_i^T)^T + (\varepsilon_{\mathbf{c}_i}^T, \varepsilon_{\mathbf{d}_i}^T)^T$. $(\varepsilon_{\mathbf{c}_i}^T, \varepsilon_{\mathbf{d}_i}^T)^T$ stands for the vector of errors on scaling and wavelet coefficients, distributed as $\mathcal{N}(0_M, \sigma_\varepsilon^2 \mathbf{I}_M)$ with 0_M the vector of zeros and $\mathbf{I}_M$ the identity matrix of size M, and $\sigma_\varepsilon^2 = \sigma_E^2$. Then we suppose that $(\nu_i^T, \theta_i^T)^T \sim \mathcal{N}(0_M, \mathbf{G} = \mathrm{Diag}(\mathbf{G}_\nu, \mathbf{G}_\theta))$, with, $\mathbf{G}_\nu$ and $\mathbf{G}_\theta$ the covariance matrices of ν_i and θ_i, respectively. We further suppose that these random coefficients are independent from the errors and that matrix $\mathbf{G}$ is diagonal, thanks to the whitening property of wavelets (Zhang and Walter 1994). Without loss in generality, we will assume that $j_0 = 0$ in the following.

2.2 *Besov Spaces and Specification of the Variance of Random Effects*

The strength of the wavelet representation is that it allows us to handle very diverse shapes of curves among which curves with irregularities that lie in particular Besov spaces. Besov spaces consist of functions that have a specific degree of smoothness. Roughly speaking, for a Besov space $B_{p,q}^s[0,1]$, parameter s indicates the number of function's derivatives, where their existence is required in a L^p-sense, q allowing finer control of the function's regularity. For a detailed study of Besov spaces, we refer to Donoho and Johnstone (1998). When dealing with functional mixed models the difficulty lies in the control of the regularity of random functions U_i, so that if the fixed function μ_ℓ is supposed to belong to some Besov space, U_i belongs to the same functional space. Following Antoniadis and Sapatinas (2007), this goal is achieved by controlling the exponential decrease of the variances of the random wavelet coefficients such that $\mathbb{V}(\theta_{i,jk}) = 2^{-j\eta}\gamma_\theta^2$ with parameter η being associated with the regularity of process U_i. Indeed, Abramovich, Sapatinas, and Silverman (1998) state that given a mother wavelet ψ of regularity r, where $\max(0, \frac{1}{p} - \frac{1}{2}) < s < r$ and given that $\mu_\ell(t) \in B_{p,q}^s[0,1]$, then,

$$U_i(t) \in B_{p,q}^s[0,1] \text{ a.s.} \quad \Longleftrightarrow \quad \begin{cases} s + \frac{1}{2} - \frac{\eta}{2} = 0 & \text{if } 1 \le p < \infty \text{ and } q = \infty, \\ s + \frac{1}{2} - \frac{\eta}{2} < 0 & \text{otherwise.} \end{cases}$$

We further allow γ_θ^2 to depend on scale and position ($\gamma_{\theta,jk}^2$) as proposed by Morris and Carroll (2006) or on cluster ($\gamma_{\theta,\ell}^2$) or on both ($\gamma_{\theta,\ell jk}^2$). As mentioned by Antoniadis and Sapatinas (2007), even if the model restricts matrix $\mathbf{G}$ to the class of matrices diagonalisable by the DWT, modeling $\mathbb{V}(\theta_{i,jk})$ as a function of scale and position allows us to account for dependencies and nonstationarities in the functional domain.

2.3 *Dimensionality Reduction*

Wavelet representations are sparse for a wide class of functional spaces which makes their use very efficient when dealing with high dimensional data. In the case of a single curve, shrinkage estimation and hard thresholding have been developed by Donoho and Johnstone (1994). Both methods present the double advantage to reduce dimensionality and to ensure good reconstruction properties. In the framework of curve clustering, our goal is to reduce the dimensionality of the problem to handle heavy datasets and not to find the optimal reconstruction rule. With this in mind we follow the strategy proposed by Antoniadis et al. (2008) and we propose a dimension reduction procedure that proceeds in two steps,

(1) We first perform individual denoising to keep coefficients which contain individual-specific information. This is done by applying nonlinear wavelet hard thresholding of coefficients $\mathbf{d}_i$ via an universal threshold as described in Donoho and Johnstone (1994). For recall, it consists in setting to zero coefficients $d_{i,jk}$ whose absolute value are below the universal threshold $\sigma\sqrt{2\log M}$. A traditional way to estimate σ is to take the average of the N robust individual noise variance estimates defined by the median absolute deviation ($\widehat{\sigma}_{\mathrm{MAD}}$) of empirical wavelet coefficients at the finest resolution level $J-1$ divided by 0.6745. In our setting this quantity provides a robust estimation of the variance level at the finest resolution level, i.e., $\mathbb{V}(d_{i,J-1,k}) = 2^{-(J-1)\eta}\gamma_\theta^2 + \sigma_\varepsilon^2$.

(2) In a second part, we take the union set of wavelet coefficients that survived thresholding. This has the advantage to remove wavelet coefficients that are zero for all individuals, and hence which are not informative regarding to the clustering goal.

As a first remark, we can point that a mixed-model specific thresholding rule could be applied by taking an estimate of the global variance of the observations which is given by $\mathbb{V}(d_{i,jk}) = 2^{-j\eta}\gamma_\theta^2 + \sigma_\varepsilon^2$. Such a level dependent thresholding would lead to greater variance estimate and hence to a greater dimensionality reduction. Nevertheless its estimation would require estimates of both parameters σ_ε^2 and γ_θ^2. This can be easily done when the individual labels are known. Otherwise, this estimation is a difficult task when individual labels are unknown because it leads to estimate variance from samples with different and unknown means. Moreover, simulations showed that the difference was negligible (not shown). Finally note that we do not use the third reduction step proposed by Antoniadis et al. (2008) which is dedicated to image segmentation.

3. Parameter Estimation and Model Selection

3.1 *An EM Algorithm for Maximum Likelihood Estimation*

Once projected in the wavelet domain, the clustering model resumes to a standard clustering model with additional random effects whose variance is of particular form. Thus, parameters are estimated by maximum likelihood using the EM algorithm. Both label variables ζ and random effects (ν, θ) are unobserved and the complete data log-likelihood can be written such that $\log \mathcal{L}(\mathbf{c}, \mathbf{d}, \nu, \theta, \zeta; \pi, \alpha, \beta, \mathbf{G}, \sigma_\varepsilon^2) = \log \mathcal{L}(\mathbf{c}, \mathbf{d} | \nu, \theta, \zeta; \pi, \alpha, \beta, \sigma_\varepsilon^2) + \log \mathcal{L}(\nu, \theta | \zeta; \mathbf{G}) + \log \mathcal{L}(\zeta; \pi)$. This likelihood can be easily computed thanks to the properties of mixed linear models: $((\mathbf{c}_i^T, \mathbf{d}_i^T)^T | (\nu_i^T, \theta_i^T)^T, \{\zeta_{i\ell} = 1\}) \sim \mathcal{N}((\alpha_\ell^T + \nu_i^T, \beta_\ell^T + \theta_i^T)^T, \sigma_\varepsilon^2 \mathbf{I}_M)$. The E-step consists in replacing the unobserved variables by their conditional expectation. Hence, cluster labels predictors $\widehat{\zeta}_{i\ell}$ are up-dated using *posterior* probabilities $\tau_{i\ell}$ such that,

$$\widehat{\zeta}_{i\ell}^{[h+1]} = \tau_{i\ell}^{[h+1]} = \frac{\pi_\ell^{[h]} f\left(\mathbf{c}_i,\ \mathbf{d}_i;\ \alpha_\ell^{[h]},\ \beta_\ell^{[h]},\ \mathbf{G}^{[h]} + \sigma_\varepsilon^{2[h]} \mathbf{I}_M\right)}{\sum_p \pi_p^{[h]} f\left(\mathbf{c}_i,\ \mathbf{d}_i;\ \alpha_p^{[h]},\ \beta_p^{[h]},\ \mathbf{G}^{[h]} + \sigma_\varepsilon^{2[h]} \mathbf{I}_M\right)},$$

with $f(.)$ the probability density function of the Gaussian distribution. Then, using notation $\widehat{\nu}_{i\ell} = \mathbb{E}(\nu_i | \mathbf{c}_i, \zeta_{i\ell} = 1) = (\widehat{\nu}_{i,j_0 k\ell})_{k=0,\dots,2^{j_0}-1}$ and $\widehat{\theta}_{i\ell} = \mathbb{E}(\theta_i | \mathbf{d}_i, \zeta_{i\ell} = 1) = (\widehat{\theta}_{i,jk\ell})_{j=j_0,\dots J-1}^{k=0,\dots,2^j-1}$, we apply the Henderson's trick (Robinson

1991) to get the following updates of the best linear unbiased predictors (BLUPs) of random effects: $\widehat{\nu}_{i\ell}^{[h+1]} = (\mathbf{c}_i - \alpha_\ell^{[h]})/(1+\lambda_\nu^{[h]})$, and $\widehat{\theta}_{i\ell}^{[h+1]} = (\mathbf{d}_i - \beta_\ell^{[h]})/(1+2^{j\eta}\lambda_\theta^{[h]})$, with $(\lambda_\nu, \lambda_\theta) = (\sigma_\varepsilon^2/\gamma_\nu^2, \sigma_\varepsilon^2/\gamma_\theta^2)$. As for the maximization part, it provides the estimators of the mean curve coefficients $\alpha_\ell^{[h+1]} = \sum_{i=1}^n \widehat{\zeta}_{i\ell}^{[h+1]}(\mathbf{c}_i - \widehat{\nu}_{i\ell}^{[h+1]})/\widehat{N}_\ell^{[h+1]}$, and $\beta_\ell^{[h+1]} = \sum_{i=1}^n \widehat{\zeta}_{i\ell}^{[h+1]}(\mathbf{d}_i - \widehat{\theta}_{i\ell}^{[h+1]})/\widehat{N}_\ell^{[h+1]}$, with $\widehat{N}_\ell^{[h+1]} = \sum_i \widehat{\zeta}_{i\ell}^{[h+1]}$, and $\pi_\ell^{[h+1]} = \widehat{N}_\ell^{[h+1]}/N$. Moreover, the EM algorithm provides a ML estimator of the variances of the model (using $j_0 = 0$): $N(M-1)\gamma_\theta^{2[h+1]} = \sum_{ijk\ell} 2^{j\eta}\widehat{\zeta}_{i\ell}^{[h+1]}(\widehat{\theta}_{i,jk\ell}^{2[h+1]} + \frac{\sigma_\varepsilon^{2[h]}}{1+2^{j\eta}\lambda_\theta^{[h]}})$, $N\gamma_\nu^{2[h+1]} = \sum_{i\ell}\widehat{\zeta}_{i\ell}^{[h+1]}(\widehat{\nu}_{i,00\ell}^{2[h+1]} + \frac{\sigma_\varepsilon^{2[h]}}{1+\lambda_\nu^{[h]}})$, and $MN\sigma_\varepsilon^{2[h+1]} = \sum_{i\ell}\widehat{\zeta}_{i\ell}^{[h+1]}\{\sum_{jk}[(d_{i,jk} - \widehat{\beta}_{\ell,jk}^{[h+1]} - \widehat{\theta}_{i,jk\ell}^{[h+1]})^2 + \frac{\sigma_\varepsilon^{2[h]}}{1+\lambda_\theta^{[h]}2^{j\eta}}] + (c_{i,00} - \widehat{\alpha}_{\ell,00}^{[h+1]} - \widehat{\nu}_{i,00\ell}^{[h+1]})^2 + \frac{\sigma_\varepsilon^{2[h]}}{1+\lambda_\nu^{[h]}}\}$. Note that we use SEM, a stochastic version of EM to avoid random initializations (Celeux and Diebolt 1986). Hard clustering can also be performed using the Maximum a posteriori (MAP) rule based on *posterior* probabilities ($\tau_{i\ell}$). As last point, we mention that η can be estimated by maximization of the likelihood using the golden search section algorithm (Kiefer 1953).

3.2 *Choosing the Number of Clusters*

We propose to choose the number of clusters using the framework of penalized likelihoods. In the following, we use notations $\mathbf{m}_L[\gamma^2], \mathbf{m}_L[\gamma_\ell^2]$ for clustering models with L groups with constant and group-dependent variances, respectively. We first use the Bayesian Information Criterion and we select the dimension that maximizes

$$\mathrm{BIC}(\mathbf{m}_L[\gamma^2]) = \log\mathcal{L}\big(\mathbf{c},\mathbf{d};\widehat{\pi},\widehat{\alpha},\widehat{\beta},\widehat{\mathbf{G}},\widehat{\sigma}_\varepsilon^2,\mathbf{m}_L[\gamma^2]\big) - \frac{|\mathbf{m}_L[\gamma^2]|}{2}\times\log(N).$$

This classical criterion is a penalized version of the observed-data log-likelihood where $|\mathbf{m}_L[\gamma^2]| = (M+1)L + |\mathbf{G}|$ is the number of free parameters of a model with L clusters, the dimension of $\mathbf{G}$ (denoted by $|\mathbf{G}|$ here) depending on the variance structure of the random effects. When considering mixed models, it is likely that the prediction of the random effects provides information regarding the number of clusters to select. To use information from hidden variables we propose to derive an integrated classification likelihood criterion in the spirit of Biernacki, Celeux, and Govaert (2000). The ICL criterion is based on the integrated likelihood of the complete data: $\log\mathcal{L}(\mathbf{c},\mathbf{d},\nu,\theta,\zeta|\mathbf{m}_L[\gamma_\ell^2]) = \log\mathcal{L}(\mathbf{c},\mathbf{d}|\nu,\theta,\zeta,\mathbf{m}_L[\gamma_\ell^2]) + \log\mathcal{L}(\nu,\theta|\zeta,\mathbf{m}_L[\gamma_\ell^2]) + \log\mathcal{L}(\zeta|\mathbf{m}_L[\gamma_\ell^2])$. For the first term we use a BIC-like approximation such that $-2\log\mathcal{L}(\mathbf{c},\mathbf{d}|\nu,\theta,\zeta,\mathbf{m}_L[\gamma_\ell^2]) \simeq NM\log\mathrm{RSS}(\mathbf{c},\mathbf{d}|\nu,\theta) + (ML+1)\times\log(N)$, with $\mathrm{RSS}(\mathbf{c},\mathbf{d}|\nu,\theta,\zeta)$ the residual sum of squares defined such that $\mathrm{RSS}(\mathbf{c},\mathbf{d}|\nu,\theta,\zeta) = \sum_{i\ell}\zeta_{i\ell}\|\mathbf{c}_i - \widehat{\alpha}_\ell - \nu_{i\ell}\|^2 + \sum_{i\ell}\zeta_{i\ell}\|\mathbf{d}_i - \widehat{\beta}_\ell - \theta_{i\ell}\|^2$. Then we derive the integrated log-likelihood of the random effects. We assume a noninformative Jeffrey prior for the variance parameters such that $g(\gamma_{\nu,\ell}^2|\zeta,\mathbf{m}_L[\gamma_\ell^2]) \propto 1/\gamma_{\nu,\ell}^2$. Using notations $N_\ell = \sum_{i=1}^N \zeta_{i\ell}$ and $\mathrm{RSS}_\ell(\nu,\zeta) = \sum_{i=1}^N \zeta_{i\ell}\nu_{i,00\ell}^2$, we get,

$$-2\log\mathcal{L}(\nu|\zeta,\mathbf{m}_L[\gamma_\ell^2]) \simeq \sum_\ell N_\ell\log\mathrm{RSS}_\ell(\nu,\zeta) - 2\sum_\ell\log\Gamma(N_\ell/2).$$

Similarly for the detail coefficients we get,

$$-2\log\mathcal{L}(\theta|\zeta,\mathbf{m}_L[\gamma_\ell^2]) \simeq (M-1)\sum_\ell N_\ell\log\mathrm{RSS}_\ell(\theta,\zeta) - 2\sum_\ell\log\Gamma(N_\ell(M-1)/2).$$

Finally for the classification term a Dirichlet *prior* is assumed for π and the corresponding integrated likelihood is approximated such as,

$$\log\mathcal{L}(\zeta|\mathbf{m}_L[\gamma_\ell^2]) \simeq \sum_{\ell=1}^L N_\ell\log\left(\frac{N_\ell}{N}\right) - \frac{(L-1)}{2}\log(N).$$

The last step of this derivation is to replace hidden variables by their predictions provided by the EM algorithm. Random effects (ν,θ) are replaced by their BLUP $(\widehat{\nu},\widehat{\theta})$, and label variables ζ are replaced by their conditional expectation τ. Put together we obtain the following integrated classification likelihood criterion (ICL), such that $-2\times\mathrm{ICL}(\mathbf{m}_L[\gamma_\ell^2])/N$ equals

$$\begin{aligned} &M\log\mathrm{RSS}(\mathbf{c},\mathbf{d}|\widehat{\nu},\widehat{\theta},\tau) + \sum_\ell\widehat{\pi}_\ell\big[\log\mathrm{RSS}_\ell(\widehat{\nu},\tau) \\ &\quad + (M-1)\log\mathrm{RSS}_\ell(\widehat{\theta},\tau)\big] \\ &\quad - \frac{2}{N}\sum_\ell\left[\log\Gamma\left(\frac{\widehat{N}_\ell}{2}\right) + \log\Gamma\left(\frac{\widehat{N}_\ell(M-1)}{2}\right)\right] \\ &\quad - 2\sum_{\ell=1}^L\widehat{\pi}_\ell\log(\widehat{\pi}_\ell) + \frac{(M+1)L}{N}\times\log(N). \end{aligned}$$

Those criteria will be compared in the simulation study.

4. Simulations and Comparison of Methods

4.1 *Definition of a General Simulation Framework*

In this section, we propose to define a unified framework for synthetic data generation for functional mixed models and functional clustering models (FCMs). Using this unified strategy different methods can be fairly compared based on appropriately simulated data. First we properly define the signal-to-noise ratio (SNR) in the functional domain. The SNR is defined as the ratio of signal power to the power of the measurement noise corrupting the signal. In our case, the power of the signal is defined such as,

$$\begin{aligned} &\lim_{T\to\infty}\frac{1}{T}\int_{\frac{T}{2}}^{-\frac{T}{2}}\sum_\ell\pi_\ell\mathbb{E}\big(|\mu_\ell(t)+U_i(t)|\big)^2\mathrm{d}t \\ &= \frac{1}{M}\sum_{\ell=1}^L\pi_\ell\left(\sum_{k=0}^{2^{j_0}-1}\alpha_{\ell,j_0k}^2 + \sum_{j\geq j_0}\sum_{k=0}^{2^j-1}\beta_{\ell,jk}^2\right) \\ &\quad + 2^{j_0}\gamma_\nu^2 + \frac{2^{j_0(1-\eta)}\gamma_\theta^2}{1-2^{(1-\eta)}}. \end{aligned}$$

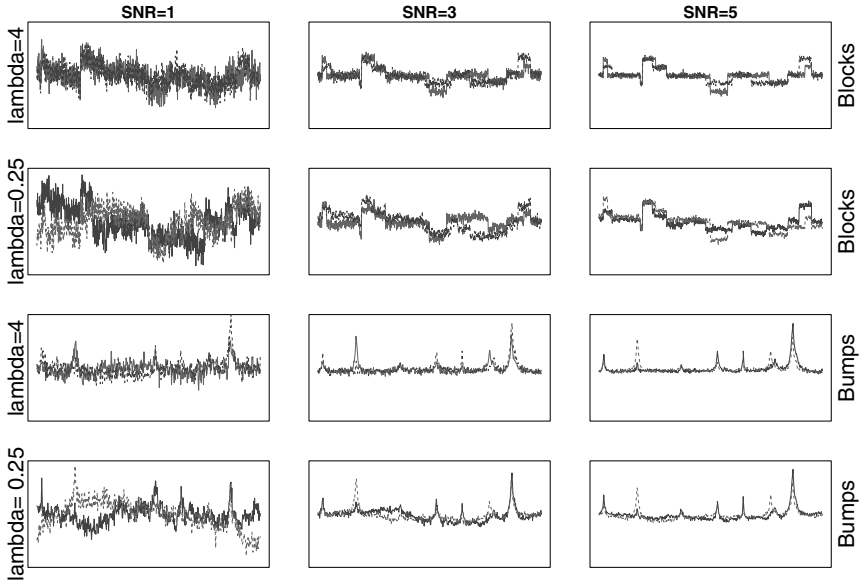


Figure 1. Example of simulated curves with varying SNR_μ and λ_{U} (one curve per cluster).

The derivation of such formula is given in the Web Supplementary Material. Hence we need to control two terms: SNR_μ that accounts for the power of the fixed effects and λ_{U} for the power of the random effect using an analogy with the λ parameter used in the EM algorithm. For this purpose we introduce parameters,

$$\mathrm{SNR}_\mu^2 = \frac{1}{M\sigma_E^2}\sum_{\ell=1}^{L}\pi_\ell\left(\sum_{k=0}^{2^{j_0}-1}\alpha_{\ell,j_0k}^2 + \sum_{j\geq j_0}\sum_{k=0}^{2^j-1}\beta_{\ell,jk}^2\right),$$
$$\lambda_{\mathrm{U}} = \sigma_E^2 / \left(\gamma_\nu^2 + \frac{\gamma_\theta^2}{1-2^{(1-\eta)}}\right).$$

When performing simulations, SNR_μ usually lies in $\{0.1, 1, 3, 5, 7\}$ and λ_{U} varies in $\{1/4, 1, 4\}$ such that small values of λ_{U} indicate an important variance for the random effects. In practice, we also choose $\gamma_\nu^2 = \gamma_\theta^2$.

To build fixed effects for simulations we generalize the approach described in Amato and Sapatinas (2005) which uses the well-known synthetic functions `Blocks`, `Bumps`, `Heavisine` and `Doppler` originally proposed by Donoho and Johnstone (1994). We choose L fixed effects for each synthetic function classes using expressions given in the Supplementary Material. Once parameters $(\mathrm{SNR}_\mu, \lambda_{\mathrm{U}}, \{\mu_\ell(t)\}_\ell)$ have been chosen (i.e., values for $\sigma_E^2, \gamma_\nu^2, \gamma_\theta^2$, and α_ℓ, β_ℓ are deduced), our simulation procedure is performed in the wavelet domain such that realizations of centered Gaussian distribution with variance $2^{-j\eta}\gamma_\theta^2$ are added to the fixed effect empirical wavelet coefficients to account for interindividual variability. Then Gaussian noise with variance σ_ε^2 is added to account for measurement errors. This unified method ensures that both fixed and random effects lie in the same Besov space, as mentioned earlier, and observed signals $\mathbf{Y}_i(\mathbf{t})$ can be recovered using the inverse DWT. An example of such simulated data is given in Figure 1.

4.2 *Simulation Design and Indicators of Performance*

Because too many configurations could be explored using simulations, we propose to fix the number of individuals at $N = 50$, the number of groups at $L = 2, 4$, the length of the signals at $M = 512$, and parameter η is set to 2. Then the simulation design explores the following configurations: $\mathrm{SNR}_\mu \in \{0.1, 1, 3, 5, 7\}$, $\lambda_{\mathrm{U}} \in \{1/4, 1, 4\}$, $\pi \in \{0.1, 0.25, 0.5\}$ ($\pi = 1/4$ when $L = 4$), each simulation being repeated 50 times. In terms of methods, we compete functional clustering models with or without mixed effects (FCMM/FCM, Functional Clustering Mixed Model/Functional Clustering Model), and we consider (or not) the dimension reduction method based on the union of coefficients. We compare these four methods to the functional clustering mixed model based on splines as proposed by James and Sugar (2003) whose `R` code is available on the web page of the authors (`http://www-bcf.usc.edu/gareth/`). Our purpose is to highlight the benefit of using wavelets when dealing with high dimensional data.

The performance of the clustering procedures are compared using the empirical error rate (EER) defined by $\mathrm{EER} = 1N\sum_{i=1}^{N}\sum_{\ell}^{L}\mathbb{I}\{\widehat{\zeta}_{i\ell}^{\mathrm{MAP}} \neq \zeta_{i\ell}\}$, where $\widehat{\zeta}_{i\ell}^{MAP}$ is the predicted class for individual i using the MAP rule, and $\zeta_{i\ell}$ is the true class. This criteria ranges from 0, for which no classification error is made to 1 which means that all individuals are misclassified. We finally consider the speed of execution of each procedure.

4.3 *Simulation Results*

4.3.1 *Clustering results.* Figure 2 presents the variations of the Empirical Error Rates according to SNR_μ and to the strength of the random effect (a small λ_{U} indicates a strong random effect). A general comment is that the functional clustering mixed model (FCMM) outperforms all methods in terms of EER compared with the FCM and Splines. This result is true even for unbalanced clusters and with an increasing number of groups (see Supplementary Material). FCMM

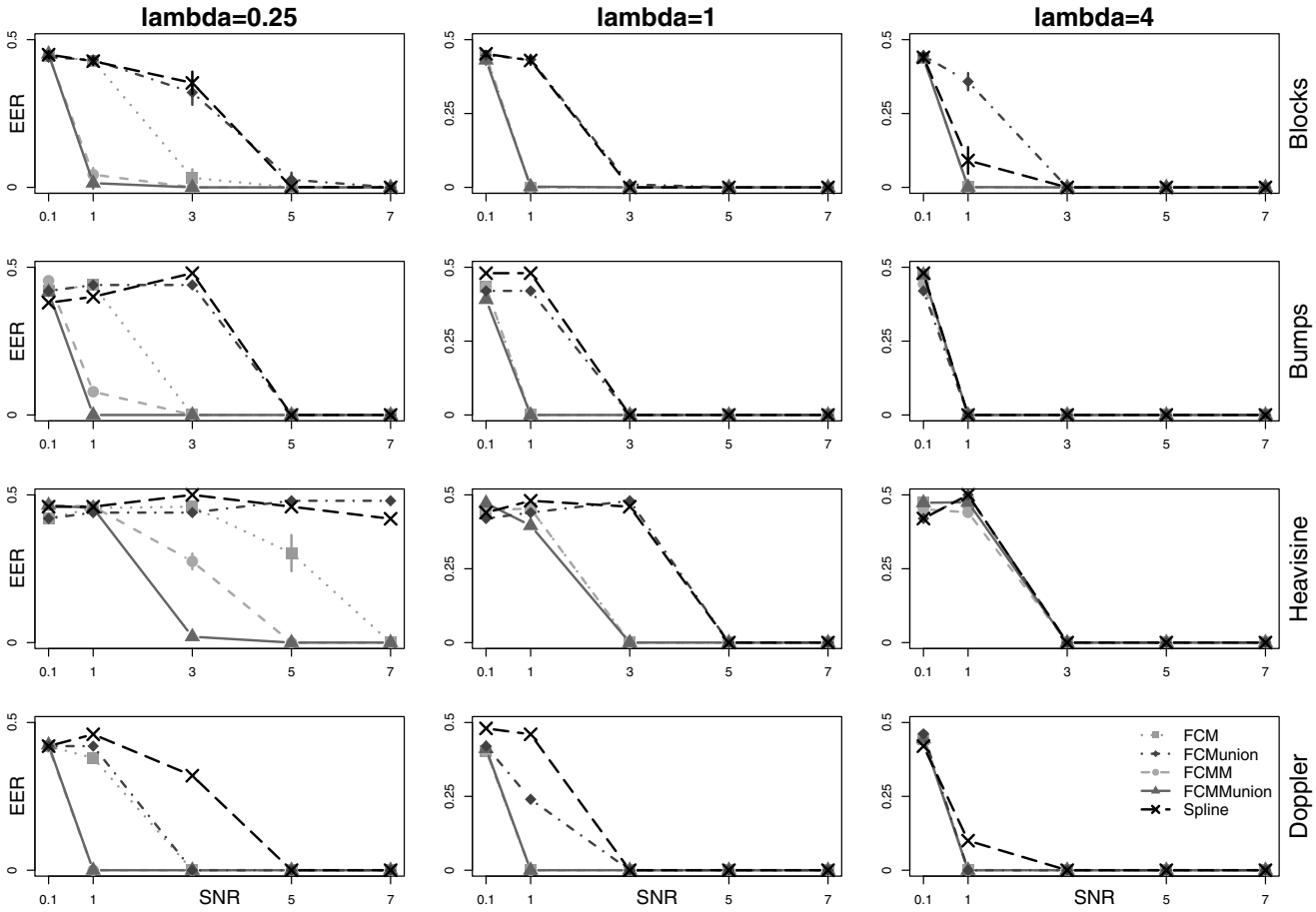


Figure 2. Variation of the empirical error rate (EER) for different estimation methods: functional clustering mixed model (FCMM), functional clustering model (FCM), with or without dimension reduction ("union"), and Splines. In columns different intensities for the variance of the random effect are considered: $\lambda_U = 0.25/1/4$ for a strong/mild/small random effect. In rows are considered different shapes for the mean curve of each group (Blocks, Bumps, Heavisine, Doppler). Results correspond to $L = 2$ clusters with balanced proportions 0.5/0.5

has two main advantages. First the modeling of functional random effects leads to a better identification of the informative structures in terms of clustering. Table 1 clearly shows that FCMM is the best method to estimate the variance of the residuals contrary to FCM that provides over-estimates (which leads to poor clustering performance).

Then dimension reduction increases the performance of FCMM by removing coefficients that are not informative with respect to clustering. This is not true for the FCM for which dimension reduction increases the EER. This trend can be explained by the bad estimation of the error's variance when random effects are not considered in the model. The selection of the coefficients that all survived thresholding leads to worst estimators in the case of FCM but the impact is moderate on the FCMM (Table 1). In the Supplementary Material we also illustrate the performance of the dimension reduction procedure. This table was not provided by Antoniadis et al. (2008) when they first proposed the union-set method. Our results show that taking the union of coefficients that survived thresholding keeps less than 10% of the coefficients. Among those coefficients, we show that a high proportion should have been thresholded whereas they are not. This means that the procedure is sensitive but not very specific, as expected when considering a union-based strategy. However, because our objective is not functional reconstruction, we consider that keeping too many coefficients is not a major issue.

Our last point concerns the time of execution of each method. When dealing with high dimensional data, it is crucial to propose methods that show reasonable computational time. Table 1 clearly shows that using wavelet-based FCMs gives the best execution times, and even when random effects are considered, time of execution remains moderate (less than 10 minutes for $N = 50$ individuals and $M = 512$ positions). Splines are known to be poorly efficient in terms of computational efficiency. This issue becomes critical when dealing with functional models with many individuals. The size of our simulated datasets was the upper limit that could be analyzed by Splines, in particular due to memory constraints. To this extent, our `R` package `curvclust` is the only freely

Table 1

Relative bias of the estimator of the error variance: $(\sigma^2 - \widehat{\sigma}^2)/\sigma^2$, and average time of execution (TOE) in minutes for different models on simulated data ($N = 50$ individuals, $M = 512$ positions). FCM, functional clustering model, FCMM, functional clustering mixed model. FCMu/FCMMu, functional clustering (mixed) models based on the union of coefficients for dimension reduction. Programs were run on a cluster of 2 octo-bicore Opteron 2.8 Ghz and 2 octo-quadcore Opteron 2.3 GHz

		Bias					TOE				
SNR^2_μ		0.1	1	3	5	7	0.1	1	3	5	7
FCM	Blocks	−2.57	−2.66	−2.96	−3.02	−2.99	2.3	2.4	2.3	2.4	2.3
	Bumps	−2.50	−2.69	−2.93	−2.93	−2.93	2.6	2.5	2.6	2.5	2.5
	Heavisine	−2.15	−2.17	−3.22	−4.30	−2.50	2.8	2.7	2.7	2.7	2.8
	Doppler	−2.73	−3.07	−3.32	−3.33	−3.33	2.9	3.2	3.1	3.2	3.2
FCMu	Blocks	−12.93	−11.33	−9.42	−9.38	−8.89	0.4	0.4	0.5	0.5	0.5
	Bumps	−12.98	−11.11	−13.46	−11.98	−11.93	0.5	0.5	0.5	0.5	0.5
	Heavisine	−11.62	−10.20	−10.07	−12.05	−15.68	0.5	0.5	0.5	0.5	0.5
	Doppler	−14.75	−13.14	−11.33	−8.59	−7.87	0.5	0.5	0.5	0.6	0.6
FCMM	Blocks	0.11	0.05	−0.01	−0.01	−0.00	16.0	16.1	15.6	15.8	16.0
	Bumps	0.09	0.04	0.01	0.01	0.01	16.1	16.3	15.2	15.3	15.4
	Heavisine	0.10	0.09	0.08	0.03	0.02	16.4	16.2	16.0	16.4	15.9
	Doppler	0.08	0.01	−0.02	−0.02	−0.01	17.5	17.4	17.5	16.4	17.0
FCMMu	Blocks	−0.11	−0.06	0.03	0.06	0.05	6.9	7.1	7.6	7.6	7.6
	Bumps	−0.10	−0.04	−0.08	−0.08	−0.05	6.7	6.7	6.8	6.7	6.7
	Heavisine	−0.10	−0.10	−0.18	−0.21	−0.19	7.1	7.3	6.8	6.8	6.8
	Doppler	−0.18	−0.06	−0.04	−0.16	−0.11	7.3	7.1	7.3	7.8	7.9
Spline	Blocks	.	.	.	.	.	25.5	26.2	23.0	23.6	22.3
	Bumps	.	.	.	.	.	23.3	26.6	22.0	21.2	21.7
	Heavisine	.	.	.	.	.	24.2	21.6	21.8	22.4	22.3
	Doppler	.	.	.	.	.	33.2	32.4	24.2	24.8	24.2

available software that performs curve clustering with functional random effects within a reduced amount of time in high dimension.

4.3.2 *Model selection results.* The model selection criteria are compared using the same simulation design with four groups (Figure 3). The BIC selects four clusters even when the SNR is low (except for Heavisine), contrary to ICL which is more stringent. Their behavior differ slightly with respect to the strength of the random effect, with ICL penalizing more when the random effect is strong whereas BIC gives similar results with respect to the strength of the random effect. Overall, differences between criteria are mild.

5. Applications

5.1 *Mass Spectrometry Data*

We first consider a SELDI-TOF mass spectrometry dataset issued from a study on ovarian cancer (Petricoin et al. 2002). The sample set includes serum profiles of 162 subjects with ovarian cancer and 91 non-cancer control subjects. Each serum profile consists of 15,154 recorded intensities corresponding to distinct m/z values. This dataset was produced by the Ciphergen WCX2 protein chip. It is available through the Clinical Proteomics Programs Databank (`http://home.ccr.cancer.gov/ncifdaproteomics/ppatter-ns.asp`, ovarian dataset 8-7-02). Before clustering, raw data are background corrected using a quantile regression procedure, and spectra are aligned using a procedure based on wavelets zero crossings (Antoniadis et al. 2007). Then the ovarian cancer dataset is made of 8192 intensities within the range of m/z ratio [1500, 14,000], ratios below 1500 being discarded due to the effects of matrix. We compete wavelet-based FCMs on these data considering different random effect structures. Procedures are applied in a nonsupervised framework to retrieve the known labels (cancer/control) and comparisons are based on empirical error rate estimates (EER, Table 2). Note that the spline-based procedure of James and Sugar (2003) could not be applied on these data because of their too high dimensionality.

The first result is that empirical error rates are high for all methods and that the introduction of random effects slightly decreases the EER whatever the random effect structure (from 38% to $\sim 25\%$). To investigate the origins of such modest performance, we also performed clustering based on group-wise aligned spectra instead of global alignment (which should be done in the unsupervised context). Results are striking: when spectra are aligned according to known labels, model $\mathbf{m}_2[\gamma^2_{jk}]$ (for which the variance of random effects depends on scale and position) results in one mismatch only (EER=0.4%). This result leads to the following conclusions. First spectra alignment is a challenge when performing subgroup discovery, and the task is much more difficult compared with supervised clustering for which labels are known. Indeed inaccuracy in spectra alignment could lead to artificial differences in individual serum profiles which decreases the performance of clustering. A promising (but challenging) perspective would be to perform clustering and alignment simultaneously. Moreover as wavelets have been shown to perform best for peak-detection/alignment (Yang et al. 2009),

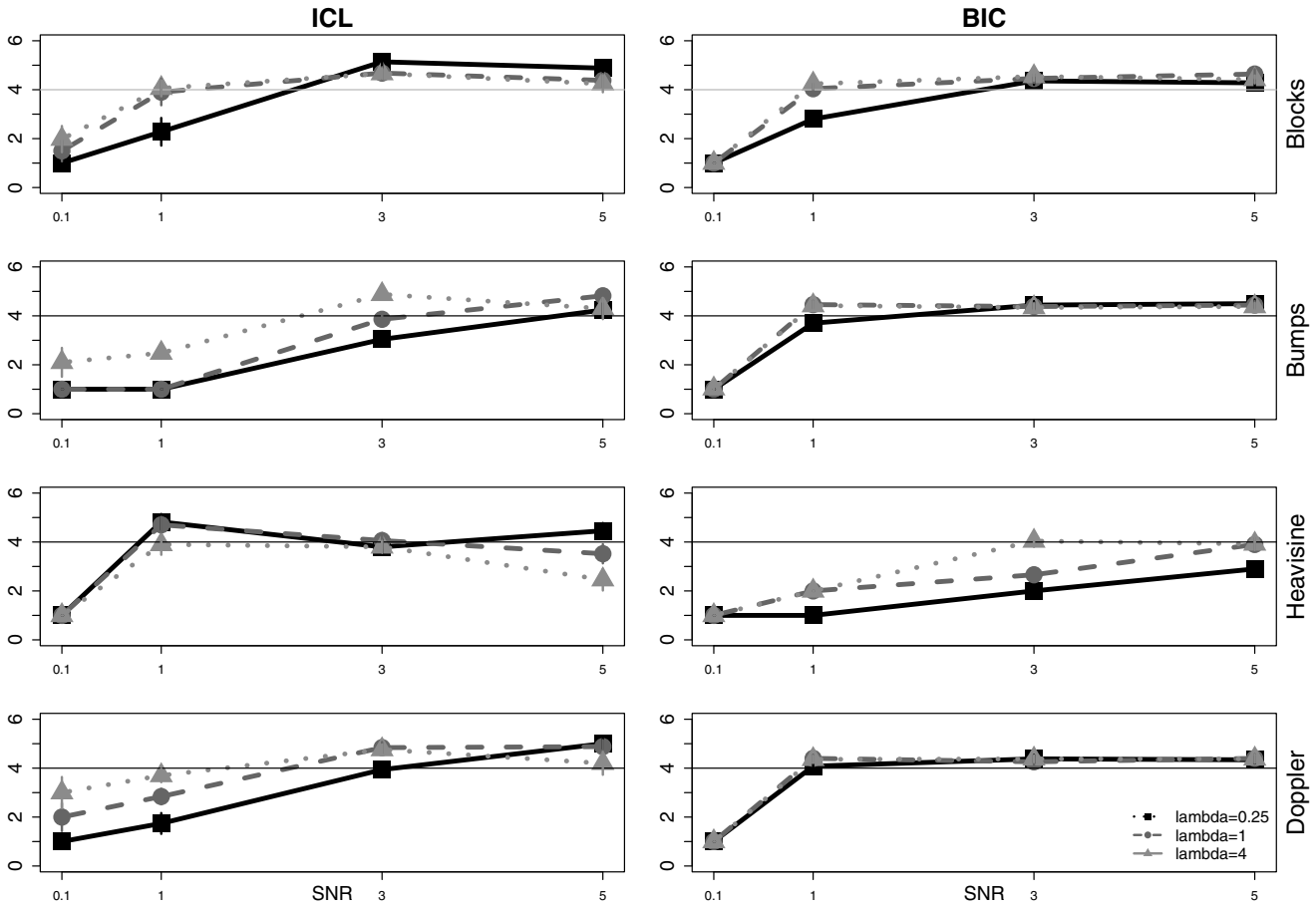


Figure 3. Estimated number of clusters using ICL and BIC when the simulated number of clusters is four (with balanced cluster sizes).

Table 2
Empirical error rates (in percent) for the Petricoin et al. (2002) *data for different models: functional clustering without random effects, two groups ($\mathbf{m}_2$), functional clustering with random effect with different variance structures for the random effect: constant $\mathbf{m}_2[\gamma^2]$, group $\mathbf{m}_2[\gamma^2_\ell]$, scale-position $\mathbf{m}_2[\gamma^2_{jk}]$, or group-scale-position dependent $\mathbf{m}_2[\gamma^2_{\ell,jk}]$*

	$\mathbf{m}_2$	$\mathbf{m}_2[\gamma^2]$	$\mathbf{m}_2[\gamma^2_\ell]$	$\mathbf{m}_2[\gamma^2_{jk}]$	$\mathbf{m}_2[\gamma^2_{\ell,jk}]$
Global alignment	38	24	24	23	23
Group alignment	20	21	22	0.4	36

our wavelet-based procedure for clustering would be a good starting point to integrate both strategies.

Then a second result is that best clustering performance are provided by a functional clustering mixed model for which the random effect has a covariance structure that depends on both scale and location. This implies that interindividual variations occur at specific ranges of m/z values, which reinforces the importance of correct spectra alignment. Interestingly, an important proportion of variance terms are close to zeros which would make the BLUPs sparse if dimension reduction was performed on random effects. Unfortunately, the task is difficult in the nonsupervised setting because BLUPs can not be computed without the knowledge of group-specific means (which would be possible in the supervised setting). Thus dimension reduction for clustering using mixed functional model remains challenging and still needs to be investigated.

5.2 *Comparative Genomic Hybridization Data*

In this last application we consider the clustering of breast-cancer tumors based on their copy number aberration profiles measured by array-based Comparative Genomic Hydridization (Fridlyand et al. 2006). Array CGH is a widely used technology that enables the characterization of genome-wide chromosomal aberrations using the microarray technology. Many statistical methods have been developed to analyze these data (van de Wiel et al. 2011). They are mainly based on segmentation methods to retrieve segments of homogeneous copy number along the genome.

Clustering individuals based on their CGH profiles is a very challenging issue and has already been considered to identify new subtypes of tumors (Chin et al. 2007). For now, subgroup discovery is mainly performed using hierarchical clustering based on segmentation results (Van Wieringen et al. 2008). However, the interindividual variability has never been

Table 3
Estimated SNR_μ^2 *and* λ_U *for the breast tumor dataset of Fridlyand et al. (2006)*

	Complete dataset	
Cluster ID	$\widehat{\mathrm{SNR}}_\mu^2$	$\widehat{\lambda}_\mathrm{U}$
1	2.1e-4	3.9e-04
2	2.3e-3	3.8e-05
3	1.3e-3	6.4e-04
4 (1q/16p)	1.5e-3	1.3e-04
5	9.3e-4	4.3e-05
	ER+ dataset	
1	2.1e-3	2.2e-04
2	7.8e-3	1.9e-05
3	1.1e-2	3.8e-05
4 (1q/16p)	4.4e-3	4.4e-04

quantified in these data, contrary to mass spectrometry for instance. Thus using our method for clustering with the Haar basis (piece-wise constant basis) is a way to perform subgroup discovery by considering random effects. In the Fridlyand et al. (2006) article, the authors analyzed the genomic profiles of 62 samples using P1/BAC CGH arrays (2464 genomic clones). We used the 55 profiles for which additional clinical information were available (the raw data can be downloaded as a supplementary material of the Fridlyand et al. 2006 article). The authors identified three main subtypes of breast cancer that differ with respect to level of genomic instability. Interestingly, Van Wieringen et al. (2008) re-analyzed the data and do not mention much correspondance between the two clustering results. Moreover, they discovered much more subgroups and noticed that "the samples in the study could be more heterogeneous than previously implied."

We also find more subgroups than the original study, with five clusters selected by ICL (two by the BIC). First, this shows the power which is gained when considering the random effect in the selection step. Then we were able to identify the 1q/16p subtype on the complete dataset (with one mismatch). This subtype was identified in the first study (Fridlyand et al. 2006) but not by other clustering methods (Van Wieringen et al. 2008) whereas it is associated to the best patient outcome. Because two of the three identified clusters in the original article concern ER positive tumors, we also performed our method on this subset of patients and retrieve the 1q/16p subtype without mismatch. In this classification, one cluster was made of three tumors (S0041, S0041, S1519) also identified as similar in the original article. As a last result Table 3 indicates that the estimated signal to noise ratio is low and the impressive strength of the random effect ($\widehat{\lambda}_\mathrm{U} \sim 10^{-4}$) also indicates that the interindividual variability is ultra-high in these data. As a consequence, finding clusters with biological significance will require rather hundreds/thousands of patients compared with 55 in the original study.

6. Conclusion

In this work we provide a methodology for model-based clustering of functional data in the presence of interindividual variability. Our method is based on a wavelet decomposition of the signal and on a mixture model that integrates random effects. We illustrate the power of such an approach in two different fields of high-throughput biology using our package `curvclust`, and we show the potentialities of functional models on array CGH data. Overall, random effects allow us to properly model the variance structure of the data, and to exhibit the high proportion of variance due to interindividual variability. This part is usually omitted in high-throughput modelling. First perspective will concern the generalization of our approach to the supervised setting. Finding biomarkers has received enormous attention in the past years, with moderate success due to the lack of reproducibility. Our study in the nonsupervised framework shows that the interindividual variability is important in these data, which may be one explanation of the difficulty to find reliable markers. Integrating random effects in the supervised setting may produce more moderate results, but at least they would be more representative of the biological variability. Finally methodological perspectives of this work will mainly concern dimension reduction. The task is difficult in the non-supervised setting and the illustration on MS data shows that dimension reduction should be performed for fixed *and* for random effects which remains challenging. This would provide a better representation of the signal by thresholding coefficients with poor information, and would increase the speed of the estimation algorithm that is sensitive to the number of selected coefficients, which is of central interest of high dimensional data.

7. Supplementary Materials

Web Appendices, tables and figures referenced in Sections 4.1, 4.2 and 4.3 are available with this article at the Biometrics website on Wiley Online Library.

Acknowledgements

Part of this work was supported by the Interuniversity Attraction Pole (IAP) research network in Statistics P5/24 and by the MSTIC project of the Joseph-Fourier University.

References

Abramovich, F., Sapatinas, T., and Silverman, B. (1998). Wavelet thresholding via a Bayesian approach. *Journal of the Royal Statistical Society Series B Statistical Methodology* **60,** 725–749.

Amato, U. and Sapatinas, T. (2005). Wavelet shrinkage approaches to baseline signal estimation from repeated noisy measurements. *Advances and Applications in Statistics* **51,** 21–50.

Antoniadis, A., Bigot, J., Lambert-Lacroix, S., and Letue, F. (2007). Non parametric pre-processing methods and inference tools for analyzing time-of-flight mass spectrometry data. *Current Analytical Chemistry* **3,** 127–147.

Antoniadis, A., Bigot, J., and von Sachs, R. (2008). A multiscale approach for statistical characterization of functional images. *Journal of Computational and Graphical Statistics* **18,** 216–237.

Antoniadis, A. and Sapatinas, T. (2007). Estimation and inference in functional mixed-effects models. *Computational Statistics & Data Analysis* **51,** 4793–4813.

Bensmail, H., Aruna, B., Semmes, O. J., and Haoudi, A. (2005). Functional clustering algorithm for high-dimensional proteomics data. *Journal of Biomedicine and Biotechnology* **2005,** 80–86.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE PAMI* **22,** 719–725.

Celeux, G. and Diebolt, J. (1986). The SEM algorithm: A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational Statistics Quarterly* **2,** 73–82.

Chin, S. F., Teschendorff, A. E., Marioni, J. C., Wang, Y., Barbosa-Morais, N. L., Thorne, N. P., Costa, J. L., Pinder, S. E., van de Wiel, M. A., Green, A. R., Ellis, I. O., Porter, P. L., Tavaré, S., Brenton, J. D., Ylstra, B., and Caldas, C. (2007). High-resolution aCGH and expression profiling identifies a novel genomic subtype of ER negative breast cancer. *Genome Biology* **8,** R215.

Donoho, D. and Johnstone, I. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika* **81,** 425–455.

Donoho, D. and Johnstone, I. (1998). Minimax estimation via wavelet shrinkage. *Annals of Statistics* **26,** 879–921.

Eckel-Passow, J. E., Oberg, A. L., Therneau, T. M., and Bergen, H. R. (2009). An insight into high-resolution mass-spectrometry data. *Biostatistics* **10,** 481–500.

Fridlyand, J., Snijders, A. M., Ylstra, B., Li, H., Olshen, A., Segraves, R., Dairkee, S., Tokuyasu, T., Ljung, B. M., Jain, A. N., McLennan, J., Ziegler, J., Chin, K., Devries, S., Feiler, H., Gray, J. W., Waldman, F., Pinkel, D., and Albertson, D. G. (2006). Breast tumor copy number aberration phenotypes and genomic instability.*BMC Cancer* **6,** 96.

Hilario, M., Kalousis, A., Pellegrini, C., and Muller, M. (2006). Processing and classification of protein mass spectra. *Mass Spectrometry Reviews* **25,** 409–449.

James, G. and Sugar, C. (2003). Clustering for sparsely sampled functional data. *Journal of the American Statistical Association* **98,** 397–408.

Kiefer, J. (1953). Sequential minimax search for a maximum. *Proceedings of the American Mathematical Society* **4,** 502–506.

Morris, J. S., Baggerly, K. A., Gutstein, H. B., and Coombes, K. R. (2010). Statistical contributions to proteomic research. *Methods in Molecular Biology* **641,** 143–166.

Morris, J. S., Brown, P. J., Herrick, R. C., Baggerly, K. A., and Coombes, K. R. (2008). Bayesian analysis of mass spectrometry proteomic data using wavelet-based functional mixed models. *Biometrics* **64,** 479–489.

Morris, J. S. and Carroll, R. J. (2006). Wavelet-based functional mixed models. *Journal of the Royal Statistical Society Series B Statistical Methodology* **68,** 179–199.

Petricoin, E. F., Ardekani, A. M., Hitt, B. A., Levine, P. J., Fusaro, V. A., Steinberg, S. M., Mills, G. B., Simone, C., Fishman, D. A., Kohn, E. C., and Liotta, L. A. (2002). Use of proteomic patterns in serum to identify ovarian cancer. *Lancet* **359,** 572–577.

Ray, S. and Mallick, B. (2006). Functional clustering by Bayesian Wavelet methods. *Journal of the Royal Statistical Society Series B Statistical Methodology* **68,** 305–332.

Robinson, G. K. (1991). That BLUP is a good thing: The estimation of random effects. *Statistical Science* **6,** 15–32.

van de Wiel, M. A., Picard, F., van Wieringen, W. N., and Ylstra, B. (2011). Preprocessing and downstream analysis of microarray DNA copy number profiles. *Briefings in Bioinformatics* **12,** 10–21.

Van Wieringen, W. N., Van De Wiel, M. A., and Ylstra, B. (2008). Weighted clustering of called array CGH data. *Biostatistics* **9,** 484–500.

Yang, C., He, Z., and Yu, W. (2009). Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis. *BMC Bioinformatics* **10,** 1–13.

Zhang, J. and Walter, G. (1994). A wavelet-based KL-like expansion for wide sense stationary random processes. *Signal Processing, IEEE Transactions on* **42,** 1737–1745.

Received July 2011. *Revised July* 2011.
Accepted August 2012.

3.2 Construction d'une suite logicielle intégrant normalisation et analyse multi-patients de données génomiques

Suite aux précédents travaux de classification non supervisée de profils de nombres de copies d'ADN, j'ai initié une collaboration sur l'analyse multi-patients de profils génomiques avec l'équipe de C. Preudhomme (Inserm U837 et laboratoire d'hématologie du CHU de Lille). En 2012, ce laboratoire d'hématologie disposait de données de génotypage (puces SNP6.0) de 300 patients leucémiques, qui étaient analysées individuellement par patient. Les puces SNP6.0 contiennent 1,8 million de positions génomiques, la moitié interrogeant des CNP (copy number positions), l'autre moitié des SNP (Single Nuleotide Polymorphism). Un SNP est la variation d'une seule paire de bases du génome, entre individus d'une même espèce. Les puces SNP permettent ainsi d'analyser à la fois le nombre de copies d'ADN, comme les puces CGH, et les pertes d'hétérozygotie (passage de deux allèles différents à deux allèles identiques). L'idée de la collaboration était de proposer une suite logicielle intégrant la normalisation de puces SNP6.0 et l'analyse multi-patients. Ce projet a fait l'objet d'une action de développement technologique (ADT) Inria et a été poursuivi grâce à un financement SIRIC (site de recherche intégrée sur le cancer) Oncolille.
Du côté de la normalisation, les logiciels utilisés à l'époque par le laboratoire d'hématologie comme GeneSpring, OpenHelix ou Partek transformaient les données de façon à attribuer un statut "AA", "AB", "BB" aux SNP de chaque patient. Cependant, ces procédures ne nous paraissaient pas adaptées aux échantillons tumoraux, pour lesquels on observe très souvent d'autres statuts, par exemple "AAA", "AAB", "ABB", "BBB". Le génotypage était donc naturellement entaché d'erreurs ou entraînait une perte d'informations qui pouvait causer des décisions finales erronées. Dans le package R MPAGenomics présenté ci-après, nous avons décidé de garder les mesures continues du signal, tout en leur appliquant une très bonne normalisation (au sens débruitage) via des packages R. Par ailleurs, les logiciels utilisés ne permettaient pas d'analyse multi-patients. Pourtant, cela nous semblait intéressant de distinguer ce qui était caractéristique de la maladie et ce qui relevait du cas particulier du patient. En particulier, les médecins étaient intéressés par la recherche de marqueurs candidats prédisant le type de rechute des patients en utilisant des méthodes de classification supervisée avec sélection de variables discriminantes. De telles méthodes existaient dans la littérature et nous pensions alors qu'il suffisait d'interfacer les packages R correspondants avec les sorties de puces pour qu'elles soient plus utilisées dans ce domaine. Les premières analyses de données réelles ont révélé que peu de marqueurs présentant des anomalies étaient communs entre les patients, ce qui affaiblissait l'intérêt des analyses de classification supervisée avec sélection de variables. Il paraissait finalement plus intéressant de compter le nombre d'anomalies par patient et de classer ensuite ces patients en fonction de leur nombre d'anomalies. Les patients qui présentaient le plus d'anomalies étaient ceux qui avaient le plus gros risque de rechute. Cela supposait alors d'améliorer les méthodes de segmentation pour détecter correctement les anomalies. En particulier, il fallait affiner le choix des paramètres par défaut des packages existants, pour qu'ils soient appropriés aux données de génotypage. L'article suivant [Grimonprez et al., 2014] présente le package R MPAGenomics issu de cette ADT Inria. Des détails sur le choix du nombre de segments pour les méthodes de segmentation seront présentés après cet article.

Grimonprez *et al. BMC Bioinformatics* 2014, **15**:394
http://www.biomedcentral.com/1471-2105/15/394

BMC Bioinformatics

SOFTWARE Open Access

MPAgenomics: an R package for multi-patient analysis of genomic markers

Quentin Grimonprez[1*], Alain Celisse[1,2], Samuel Blanck[1], Meyling Cheok[3], Martin Figeac[4] and Guillemette Marot[1,5]

Abstract

Background: Last generations of Single Nucleotide Polymorphism (SNP) arrays allow to study copy-number variations in addition to genotyping measures.

Results: `MPAgenomics`, standing for multi-patient analysis (MPA) of genomic markers, is an R-package devoted to: (*i*) efficient segmentation and (*ii*) selection of genomic markers from multi-patient copy number and SNP data profiles. It provides wrappers from commonly used packages to streamline their repeated (sometimes difficult) manipulation, offering an easy-to-use pipeline for beginners in R.
The segmentation of successive multiple profiles (finding losses and gains) is performed with an automatic choice of parameters involved in the wrapped packages. Considering multiple profiles in the same time, `MPAgenomics` wraps efficient penalized regression methods to select relevant markers associated with a given outcome.

Conclusions: `MPAgenomics` provides an easy tool to analyze data from SNP arrays in R. The R-package `MPAgenomics` is available on CRAN.

Keywords: SNP arrays, Segmentation of genomic data, Marker selection, Multi-patient analysis, R package

Background

Genome-wide Single Nucleotide Polymorphism (SNP) arrays have been widely used over the past few years [1]. First generations were measuring only genetic variations of Single Nucleotide Polymorphisms, which are single base pair mutations at specific loci. Last generations (e.g. SNP5.0, SNP6.0) also include non-polymorphic probes in order to study copy-number variations along the genome in addition to genotyping measures. These arrays are especially used to study the impact of diseases, e.g. cancer, on the human genome.

Analyzing data from genome-wide SNP arrays within R requires several packages, e.g. `aroma` for normalization of Affymetrix® SNP arrays [2,3], `changepoint` or `cghseg` for segmentation of copy number profiles [4], `cghcall` for labelling segments [5], and `glmnet` for penalized regressions [6]. Each package performs a specific task along the whole analysis but none of them is related to the others. Output formats of given packages are often not compatible with input formats required by the other, making their use tricky for beginners in R. One main contribution of the `MPAgenomics` R package is to aggregate these commonly used packages, providing wrappers to inter-relate them automatically.

At each step of the analysis a large amount of packages are available to perform normalization, segmentation or marker selection. A careful choice of only a few methods is required to provide an easy-to-use and efficient tool.

In this software article, we describe two different pipelines implemented in the R package `MPAgenomics`. Both of them perform the whole analysis from raw data to normalization, and then either successive segmented profiles, or a list of genomic markers selected from all available profiles.

Implementation

`MPAgenomics` is implemented in R [7]. The package is divided in four main parts: data normalization, segmentation, calling and marker selection. Each part depends on different packages. `MPAgenomics` provides wrappers for some functions of these packages and facilitates the interaction between outputs and inputs of different functions.

*Correspondence: quentin.grimonprez@inria.fr
[1] MODAL team, Inria Lille-Nord Europe, Villeneuve-d'Ascq, France
Full list of author information is available at the end of the article

It remedies some problems with the wrapped packages such as confusing parameter names.

Data normalization

The normalization process in `MPAgenomics` contains *technical biases correction* and *copy number and allele B fraction estimation*. Following [8], *allele B fraction* refers to the proportion of the total signal coming from allele B. Normalization methods are available for Affymetrix® arrays (10K, 100K, 500K, GenomeWideSNP 5 & 6, and CytoScanHD). The estimation of the total copy number and allele B fraction is made by *CRMAv2* [9] originally implemented in the `aroma` packages. For studies with matched normal-tumor samples, a better estimation is suggested and implemented for the allele B fraction of the tumoral sample with the *TumorBoost* method [8].

The use of `aroma` packages is difficult for neophytes due to the strict folder architecture it requires and the documentation of the project which is mainly dedicated to experts able to criticize each method proposed and understand details of each procedure.

`MPAgenomics` provides documentation with a detailed example explaining how to quickly analyze data. The tutorial can be accessed in R by running the following commands:

```
library(MPAgenomics)
vignette("MPAgenomics")
```

More details on each step or wrapper are given to help advanced users to run each function separately.

Several features in the original `aroma` packages create new folders and files within the architecture. Matching files from different processes associated with a given sample can be tricky for neophytes. `MPAgenomics` implements a wrapper to build the folder architecture, check filenames automatically, process *CRMAv2* and *TumorBoost* normalization steps. Miscellaneous functions are also provided to ease some actions like signal extraction. Furthermore, different graphs such as the copy number profile can be saved in the working directory for further visualization.

The following steps (segmentation, calling and/or selection of genomic markers) are available in two settings. One is aroma-based and exploits the folder architecture and the files generated along the process. The second does not depend on aroma and allows advanced users to use their own normalized data.

Segmentation

Although the use of manual annotations provides the best segmentation results [10], it appears essential for multi-patient analysis to avoid relying on them since they are time-consuming.

Therefore, following simulation results of [10], `MPAgenomics` wraps the `CGHSEG` [11,12] and `PELT` (Pruned Exact Linear Time) [13] segmentation methods which appeared to be those with the best overall performance.

`PELT` and `CGHSEG` methods fit a Gaussian maximum likelihood model but they differ in the way they choose the number of segments. `CGHSEG` requires the maximal number of segments as input. In `MPAgenomics`, the optimal number of segments is chosen according to a penalty $C \times K \times \left(2.5 + \log\left(\frac{P}{K}\right)\right)$ with a profile of length P, K the number of segments and $C > 0$ a parameter to choose [14]. This choice is performed using slope heuristics [15]. The `PELT` method returns a segmentation with a number of segments automatically chosen by the algorithm according to a penalty $K\rho\log(P)$ with $\rho > 0$ a parameter to choose. The choice of the penalty parameter has been raised in [4]. `MPAgenomics` suggests an automatic sample-specific choice of ρ chromosome by chromosome (see package vignette for details on the method). In `MPAgenomics`, the two methods, `CGHSEG` with the slope heuristic and `PELT` with our calibration method, are proposed. By default, `CGHSEG` is used because it is quicker than `PELT` due to the multiple execution required by the ρ calibration method we propose.

The implemented segmentation methods are independently available for both copy number and allele B fraction profiles. In the case of allele B fraction segmentation, only heterozygous SNPs are kept. First, a naive genotype call [8] is performed on each normal sample in order to separate heterozygous SNPs from homozygous SNPs. Naive genotyping method assumes SNPs are bi-allelic and therefore is not recommended for tumor samples. Thus allele B fraction segmentation in `MPAgenomics` requires matched normal-tumor pairs. Then, following [16], the resulting signal is centered on 0.5 and symmetrized, which makes it similar to the usual copy number.

Calling

From each segmented profile, the *CGHcall* method [17] is run to label every copy-number segment in terms of *loss*, *normal*, and *gain*.

CGHcall depends on a parameter, named *cellularity*, corresponding to the contamination of a sample with healthy cells. In `MPAgenomics`, this parameter can be modified by users, by default its value is 1 meaning that tumor samples are pure.

In the aroma-dependent function, segmentation and calling are performed with the same wrapper. The calling is run for each profile separately. Results are saved in text format in the working folder architecture.

Selection of genomic markers

The goal is to select genomic markers (e.g. SNPs or CNV) associated with a given response from all patient profiles simultaneously. There is no need to perform segmentation

and calling before the multi-patient analysis, marker selection is made over all copy-number profiles. However, segmentation can be performed before marker selection if wanted, in order to reduce the noise and the dimensionality of the problem.

Assuming I individuals and P potential markers, then for each individual i, y_i denotes the response and $x_{i,p}$ the corresponding normalized value of copy number or allele B fraction signal at genomic position p.

Due to the huge number of markers ($P >> I$), `MPAgenomics` uses by default the *lasso* [18] regularization method to select very few ones. This method offers two advantages: (*i*) it selects only few variables, easing the interpretability of results, (*ii*) there exist some algorithms such as the lars [19] to solve quickly the *lasso* problem and support high-dimensional data.

The lasso regularization method consists in minimizing $g_\lambda : \beta \in \mathbb{R}^P \mapsto g_\lambda(\beta)$, where

$$g_\lambda(\beta) = \sum_{i=1}^{I} (y_i - (X\beta)_i)^2 + \lambda \sum_{p=1}^{P} |\beta_p| \quad ,$$

with $(X\beta)_i = \sum_p x_{i,p}\beta_p$ and $\lambda > 0$ controlling the number of non-zero coordinates of β. After minimization, non-zero coefficients β_p correspond to influential positions to predict the response.

`MPAgenomics` genomic marker selection drastically improves currently available packages in terms of computation time. With the linear regression model, it efficiently provides the exact solution by using the new R package `HDPenReg`, which is an optimized implementation of the *lars* algorithm [19] specially dedicated to a huge number of markers.

Since the theoretical grounding of Lasso when $P >> I$ relies on a theoretical condition (see [20]) that cannot be easily checked in practice, the spike and slab algorithm [21,22] – a three steps algorithm performing filtering, estimation and variable selection – is also provided in `MPAgenomics` as an alternative.

Logistic regression is also available for binary responses. In this case, `MPAgenomics` wraps the `glmnet` package [6] in the whole process. Unlike `HDPenReg` it does not provide the exact solution but is computationally very efficient. With `glmnet` and `HDPenReg`, the regularization parameter λ is chosen by k-fold cross-validation [23]. The selected variables are the most relevant ones regarding the response.

Discussion

`MPAgenomics` is mainly dedicated to beginners in SNP array analyses. It solves problems commonly encountered by neophytes such as interaction between different packages or specialized documentation dedicated to experts in the field. In addition, `MPAgenomics` suggests careful and automatic choices of crucial parameters at each part of the analysis.

To achieve simplicity of usage, `MPAgenomics` does not propose all options implemented in the wrapped packages, especially for normalization. However, outputs are generated in such a way that interaction between wrapped packages and `MPAgenomics` is facilitated. For example, the strict directory structure of `aroma` packages is built by `MPAgenomics`. Therefore, advanced users may directly use specific options of `aroma` to enhance their analysis without renormalizing data from scratch.

As specified in the data normalization section, segmentation, calling and marker selection steps can be performed without the use of aroma. This allows users to provide their own normalized data into matrices. This is useful for non-Affymetrix® SNP arrays, CGH arrays or high-throughput sequencing data. For the latter, count data might need a variance-stabilizing transformation into Gaussian data before using current segmentation, calling and marker selection. For example, the Anscombe transform [24] can be used in addition to appropriate normalization specific to the used technology (target sequencing, whole-genome sequencing).

Currently, copy number and allele B fraction are segmented independently from each other. Research is ongoing to propose joint segmentation methods allowing to detect uniparental disomies, fragments which present a normal copy number but a loss of heterozygosity in the corresponding allele B fraction.

Conclusions

`MPAgenomics` provides user-friendly wrappers for normalization and multi-patient analysis of high-throughput genomic data. It offers a guideline for beginners in copy-number variation analysis focusing on proven methods for their effectiveness. `MPAgenomics` also provides automatic choices of crucial parameters for segmentation and selection of markers.

Even though normalization is provided for Affymetrix® arrays, other steps (segmentation, calling, and marker selection) can be applied to normalized data from other DNA arrays and next-generation sequencing data.

Availability and requirements

Project name: MPAgenomics
Project home page: http://cran.at.r-project.org/package=MPAgenomics
Operating system(s): Platform independent
Programming language: R
Other requirements: none
License: GNU GPL (>=2)
Any restrictions to use by non-academics: None

Competing interests

The authors declare that they have no competing interests.

Authors' contributions
QG implemented the first versions of `MPAgenomics` and `HDPenReg`. He helped in their maintenance and drafted the manuscript and the vignette of the package. AC contributed for choices of crucial parameters in segmentation, and helped draft the manuscript. SB maintained `MPAgenomics` and its vignette. MC and MF participated in discussions on data analysis and results. GM conceived of the project and managed it, selected key packages for wrapping. She occasionally participated to the implementation and helped draft the manuscript and the vignette. All authors read and approved the final manuscript.

Acknowledgements
We thank Serge Iovleff for his help implementing `HDPenReg`. We also thank Claude Preudhomme and Olivier Nibourel for providing data presented in the vignette of the package, and for their helpful clinical competences to interpret the results. The development of this package was funded by the Inria Technological Development Action (ADT) named *MPAGenomics*.

Author details
[1]MODAL team, Inria Lille-Nord Europe, Villeneuve-d'Ascq, France. [2]Laboratoire Paul Painlevé, Université Lille 1, Villeneuve-d'Ascq, France. [3]Inserm, U837, Team 3, Cancer Research Institute of Lille, Lille, France. [4]Plate-forme de génomique fonctionnelle et structurale, IFR-114, Université Lille 2, Lille, France. [5]EA 2694, Université Lille 2,Lille, France.

Received: 23 July 2014 Accepted: 19 November 2014
Published online: 14 December 2014

References

1. LaFramboise T: **Single nucleotide polymorphism arrays: a decade of biological, computational and technological advances.** *Nucleic Acids Res* 2009, **37**(13):4181–4193.
2. Bengtsson H, Simpson K, Bullard J, Hansen K: **aroma.affymetrix: A generic framework in R for analyzing small to very large Affymetrix data sets in bounded memory.** Technical Report 745, Department of Statistics, University of California, Berkeley; 2008.
3. Bengtsson H, Bullard J, Hansen K, Neuvial P, Purdomand E, Robinson M, Simpson K: **Aroma project.** 2010. [http://www.aroma-project.org/]
4. Killick R, Eckley I: **Changepoint: An R Package for Changepoint Analysis.** 2013. R package version 1.1, [http://www.lancs.ac.uk/~killick/Pub/KillickEckley2011.pdf]
5. van de Wiel M, Vosse S: **CGHcall: Calling Aberrations for Array CGH Tumor Profiles.** 2012. R package version 2.20.0 [http://www.bioconductor.org/packages/release/bioc/vignettes/CGHcall/inst/doc/CGHcall.pdf]
6. Friedman JH, Hastie T, Tibshirani R: **Regularization paths for generalized linear models via coordinate descent.** *J Stat Softw* 2010, **33**(1):1–22.
7. R Core Team: *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing; 2014. [http://www.R-project.org/]
8. Bengtsson H, Neuvial P, Speed TP: **Tumorboost: Normalization of allele-specific tumor copy numbers from a single pair of tumor-normal genotyping microarrays.** *BMC Bioinformatics* 2010, **11:**245.
9. Bengtsson H, Wirapati P, Speed TP: **A single-array preprocessing method for estimating full-resolution raw copys from all Affymetrix genotyping arrays including GenomeWideSNP 5 & 6.** *Bioinformatics* 2009, **25**(17):2149–2156.
10. Hocking T, Schleiermacher G, Janoueix-Lerosey I, Boeva V, Cappo J, Delattre O, Bach F, Vert J-P: **Learning smoothing models of copy number profiles using breakpoint annotations.** *BMC Bioinformatics* 2013, **14**(1):164.
11. Picard F, Robin S, Lavielle M, Vaisse C, Daudin J-J: **A statistical approach for array CGH data analysis.** *BMC Bioinformatics* 2005, **6**(1):27.
12. Rigaill G: **Pruned dynamic programming for optimal multiple change-point detection.** arXiv e-print, 2010, arXiv/1004.0887.
13. Killick R, Fearnhead P, Eckley IA: **Optimal detection of changepoints with a linear computational cost.** *J Am Stat Assoc* 2012, **107**(500):1590–1598.
14. Lebarbier E: **Detecting multiple change-points in the mean of gaussian process by model selection.** *Signal Process* 2005, **85**(4):717–736.
15. Birgé L, Massart P: **Minimal penalties for gaussian model selection.** *Probability Theory Related Fields* 2007, **138**(1-2):33–73.
16. Staaf J, Lindgren D, Vallon-Christersson J, Isaksson A, Göransson H, Juliusson G, Rosenquist R, Höglund M, Borg Å, Ringnér M: **Segmentation-based detection of allelic imbalance and loss-of-heterozygosity in cancer cells using whole genome SNP arrays.** *Genome Biol* 2008, **9**(9):R136.
17. van de Wiel MA, Kim KI, Vosse SJ, van Wieringen WN, Wilting SM, Ylstra B: **CGHcall: calling aberrations for array CGH tumor profiles.** *Bioinformatics* 2007, **23**(7):892–894.
18. Tibshirani R: **Regression shrinkage and selection via the lasso.** *J R Stat Soci Series B* 1994, **58:**267–288.
19. Efron B, Hastie T, Johnstone I, Tibshirani R: **Least angle regression.** *Ann Stat* 2004, **32:**407–499.
20. Ravikumar P, Wainwright M. J, Raskutti G, Yu B: **High-dimensional covariance estimation by minimizing l_1-penalized log-determinant divergence.** *Electron J Stat* 2011, **5:**935–980.
21. Ishwaran H, Rao JS: **Spike and slab variable selection: frequentist and bayesian strategies.** *Ann Stat* 2005, **33**(2):730–773.
22. Ishwaran H, Rao JS: **Generalized ridge regression: geometry and computational solutions when p is larger than n.** Technical Report, Department of Public Health Sciences Division of Biostatistics, University of Miami, Miller School of Medecine; 2010.
23. Arlot S, Celisse A: **A survey of cross-validation procedures for model selection.** *Stat Surv* 2010, **4:**40–79.
24. Anscombe FJ: **The transformation of poisson, binomial, and negative-binomial data.** *Biometrika* 1948, **35**(3/4):246–254.

doi:10.1186/s12859-014-0394-y
Cite this article as: Grimonprez *et al.*: MPAgenomics: an R package for multi-patient analysis of genomic markers. *BMC Bioinformatics* 2014 **15**:394.

3.2.1 Choix du nombre de segments pour les méthodes de segmentation

Choix du nombre de segments pour CGHseg Le package CGHseg [Picard et al., 2005, Rigaill, 2015] demande à l'utilisateur de choisir un nombre maximal $K_{\max}$ de segments en entrée et renvoie toutes les segmentations possibles avec un nombre de segments compris entre 2 et ce nombre maximal. Cela suppose donc que l'utilisateur choisisse ensuite la meilleure segmentation. Dans MPAGenomics, nous avons implémenté l'heuristique de pente [Birgé and Massart, 2007] pour choisir automatiquement le nombre de segments.

Soit k le nombre de segments, le critère à minimiser proposé par [Lebarbier, 2005] est

$$g(k, C) = \frac{1}{P}\left(R(k) + C \times k \times \left(2.5 + \log(\frac{P}{k})\right)\right)$$

où $R(k)$ est la perte quadratique entre le signal observé et le signal segmenté, et C une constante à optimiser.

La première étape est de trouver la constante optimale $\hat{C}$ à utiliser. Etant donnée une liste de valeurs potentielles de $\mathcal{C}$, nous estimons, pour chaque constante $C \in \mathcal{C}$, le nombre de segments en minimisant $g(k, C)$ et traçons le nombre de segments obtenu en fonction de $\mathcal{C}$. La constante optimale $\hat{C}$ est alors définie comme $2 \times C_0$, avec C_0 la constante associée au plus grand saut illustré dans la figure 3.2.

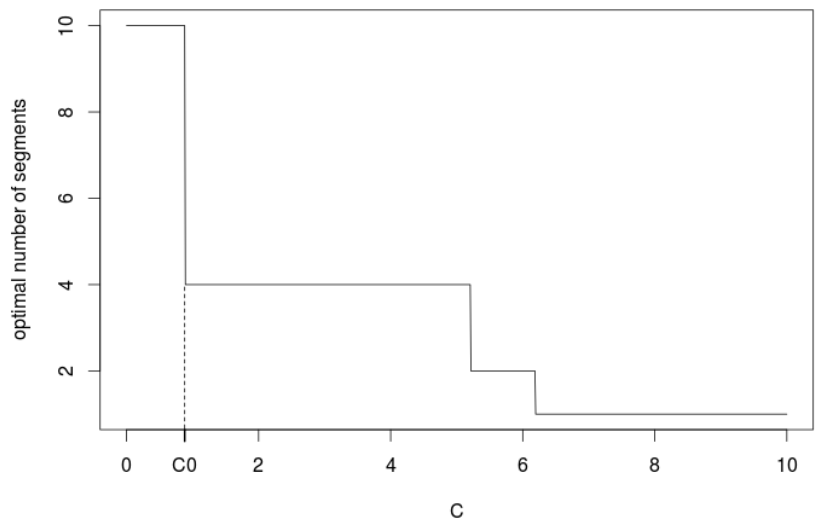


Figure 3.2: Nombre optimal de segments trouvé en minimisant $g(C, k)$ en fonction de C. C_0 est la constante associée au plus grand saut dans le nombre de segments.

Une fois la constante optimale $\hat{C}$ trouvée, le nombre optimal de segments $\hat{k}$ est celui minimisant $g(k, \hat{C})$.

Choix du nombre de segments pour PELT La pénalité de PELT [Killick et al., 2012] est de la forme $\lambda \times K \times \log(P)$ avec K le nombre de segments et P la longueur

du profil. Seul le paramètre λ peut être choisi par l'utilisateur. Comme la valeur par défaut dans PELT ($\lambda = 1$) ne fournissait pas des segments satisfaisants au regard des segments trouvés manuellement (observations sur 70 profils étudiés dans [Renneville et al., 2014]), nous avons proposé une autre méthode pour calibrer le paramètre λ. Pour choisir λ, MPAgenomics lance PELT sur une gamme de valeurs de λ fournie par défaut ou rentrée par l'utilisateur. Puis le nombre de segments est tracé en fonction du paramètre λ (Figure 3.3).

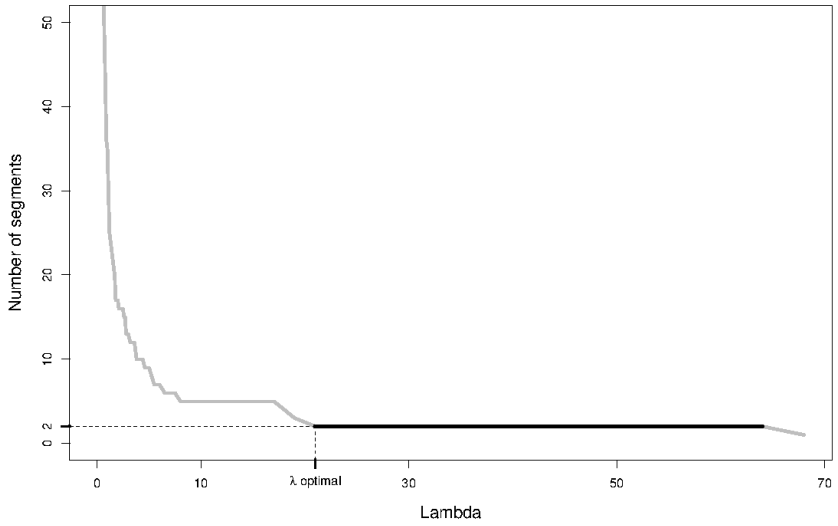


Figure 3.3: Variation du nombre de segments en fonction de λ.

Notre intuition est de dire qu'il faut choisir λ de telle façon à être dans le plateau le plus large. En effet, cela indique que la pénalité doit augmenter considérablement pour retirer des points de rupture. Dans ce plateau, nous gardons la valeur de λ la plus à gauche.

Paramètre spécifique à chaque profil ou paramètre commun? Répéter l'éxécution de PELT pour calibrer le paramètre λ augmente inévitablement le temps de calcul. Nous avons étudié la possibilité d'utiliser un paramètre commun pour PELT. Nous avons d'abord classé les 70 profils du jeu de données publié dans [Renneville et al., 2014] en trois groupes présentant des rapports signal sur bruit homogènes par un modèle de mélange gaussien. La figure 3.4 montre les résultats obtenus pour le chromosome 1 de chaque patient. Nous avons représenté les valeurs λ choisies par notre méthode de calibration pour chaque profil. Les couleurs (gris clair, gris, et noir) indiquent le rapport signal sur bruit de chaque groupe (respectivement faible, moyen, et fort).

Alors que le groupe avec le plus petit rapport signal sur bruit contient des plateaux avec des faibles valeurs de λ, les autres groupes montrent des plateaux correspondant à la fois à des gammes de faibles valeurs et des gammes de fortes valeurs. Ceci illustre le fait qu'un choix de paramètre commun aurait conduit à des segmentations erronées. Nous avons aussi observé ce type de comportement sur d'autres chromosomes et pour d'autres critères comme par exemple celui de la variance.

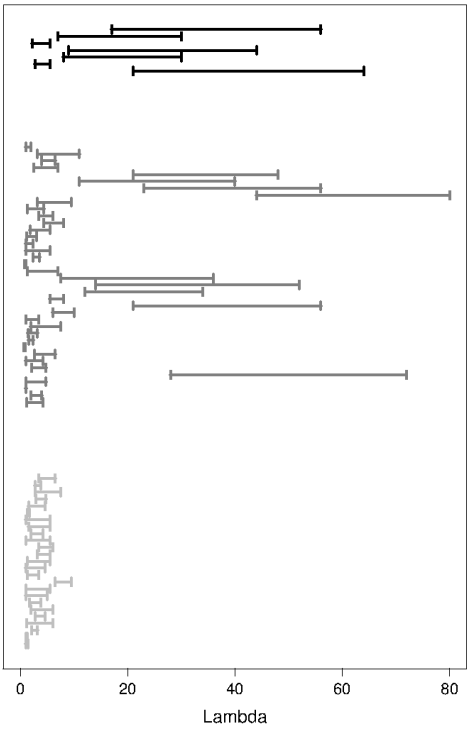


Figure 3.4: Plateaux les plus larges de λ (axe des abscisses) pour 70 profils de nombres de copies d'ADN sur le chromosome 1 (axe des ordonnées). Les couleurs indiquent les classes de rapport signal sur bruit (gris clair < gris < noir).

3.2.2 Perspectives sur ce travail

Même si le package R développé par Quentin Grimonprez était plus simple d'utilisation que les packages initiaux, notamment parce qu'il proposait des choix par défaut à l'utilisateur qui rendaient des résultats satisfaisants, nous avons constaté que le simple fait qu'il soit en R était un frein à son utilisation par les biologistes du laboratoire d'hématologie du CHU de Lille. C'est pourquoi Samuel Blanck l'a interfacé avec Galaxy durant la deuxième année de l'ADT Inria.

Dans le package R, nous avions choisi comme méthode de segmentation par défaut celle implémentée dans le package CGHseg développé par Guillem Rigaill (AgroParisTech, Univ. Evry puis Inrae) car son exécution était plus rapide que celle de PELT, à cause de notre procédure de calibration associée pour changer le paramètre $\lambda = 1$ par défaut. Le package CGHseg s'est rapidement retrouvé orphelin sur le site officiel du logiciel R (CRAN), faute de maintenance. Ayant pourtant toute confiance dans ce package CGHseg, nous sommes actuellement en train de développer une version dockerisée de l'instance Galaxy-MPAGenomics, qui sera mise à disposition sur le site de la plateforme bilille. Dans cette version, nous garderons les deux méthodes implémentées. Pour le CRAN, nous allons proposer une version allégée reposant uniquement sur PELT, afin de diminuer le travail de maintenance et pouvoir remettre MPAGenomics sur le CRAN.

3.3 Détection de ruptures à partir de méthodes à noyaux

Dans le package MPAgenomics, nous proposons d'analyser les pertes d'hétérozygotie en gardant les mesures continues du signal. Nous récupérons les fractions d'allèle B pour chaque position SNP, puis en centrant le signal autour de 0,5 et en le symétrisant, nous obtenons un profil ressemblant à celui du nombre de copies d'ADN et pouvons alors utiliser les mêmes méthodes de segmentation [Grimonprez et al., 2014]. Etudier à la fois les profils de nombre de copies d'ADN et celles des fractions d'allèle B permet de détecter des disomies uniparentales (UPD), c'est à dire la présence chez un patient de deux chromosomes d'une même paire provenant d'un seul de ses parents. Ce genre d'anomalies étant courant dans les leucémies aigües myéloblastiques, les hématologues étaient très demandeurs d'avoir une analyse conjointe des profils de nombre de copies d'ADN et de fraction d'allèle B.

Par ailleurs, Alain Celisse, impliqué dans ce projet d'ADT, avait participé à la conception d'une nouvelle approche de détection de ruptures basée sur les méthodes à noyaux [Arlot et al., 2012]. Cette approche permettait de détecter des changements dans la distribution et pas seulement dans la moyenne ou dans la variance. L'utilisation des noyaux rendait possible l'utilisation conjointe des deux types de profils, en définissant un nouveau noyau à partir des noyaux de chaque type (cf. équation 18 de l'article présenté dans cette section). Cependant, son implémentation de l'époque, trop coûteuse en temps et en espace, rendait inenvisageable son application sur les signaux des puces SNP6.0 du laboratoire d'hématologie. C'est alors qu'Alain Celisse a contacté Guillem Rigaill, à la fois expert en programmation dynamique et en analyse de profils de copies d'ADN pour implémenter plus efficacement l'approche. L'article suivant décrit ces nouveaux algorithmes efficaces développés dans le cadre de cette collaboration. Ces algorithmes sont implémentés dans le package R KernSeg, disponible sur Rforge.

Ma contribution a essentiellement été l'encadrement de Morgane Pierre-Jean, qui a mis en place les simulations, et la reprise de ses simulations avec Guillem pour avoir des résultats probants à publier. Bien que l'approche soit théoriquement très performante, il nous a fallu plusieurs allers retours et plusieurs mois avant de comprendre qu'il fallait exclure des segments de petite taille dans KernSeg pour espérer battre la méthode ECP. Cette contrainte est détaillée dans le paragraphe 4.5.2 de l'article suivant.

Computational Statistics and Data Analysis 128 (2018) 200–220

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

New efficient algorithms for multiple change-point detection with reproducing kernels

A. Celisse [c,e,*], G. Marot [a,c], M. Pierre-Jean [a,b], G.J. Rigaill [b,d]

[a] Univ. Lille Droit et Santé, EA 2694 - CERIM, F-59000 Lille, France
[b] UMR 8071 CNRS - Université d'Evry - INRA, Laboratoire Statistique et Génome Evry, France
[c] Inria Lille Nord Europe, Équipe-projet Inria MODAL, France
[d] Institute of Plant Sciences Paris-Saclay, UMR 9213/UMR1403, CNRS, INRA, Université Paris-Sud, Université d'Evry, Université Paris-Diderot, Sorbonne Paris-Cité, France
[e] Univ. Lille Sciences et Technologies, CNRS, UMR 8524 - Laboratoire Paul Painlevé, F-59000 Lille, France

ARTICLE INFO

Article history:
Received 30 August 2016
Received in revised form 4 July 2018
Accepted 6 July 2018
Available online 17 July 2018

Keywords:
Kernel method
Gram matrix
Nonparametric change-point detection
Model selection
Algorithms
Dynamic programming
DNA copy number
Allele B fraction

ABSTRACT

Several statistical approaches based on reproducing kernels have been proposed to detect abrupt changes arising in the full distribution of the observations and not only in the mean or variance. Some of these approaches enjoy good statistical properties (oracle inequality, consistency). Nonetheless, they have a high computational cost both in terms of time and memory. This makes their application difficult even for small and medium sample sizes ($n < 10^4$). This computational issue is addressed by first describing a new efficient procedure for kernel multiple change-point detection with an improved worst-case complexity that is quadratic in time and linear in space. It is based on an exact optimization algorithm and deals with medium size signals (up to $n \approx 10^5$). Second, a faster procedure (based on an approximate optimization algorithm) is described. It relies on a low-rank approximation to the Gram matrix and is linear in time and space. The resulting procedure can be applied to large-scale signals ($n \geq 10^6$). These two procedures (based on the exact or approximate optimization algorithms) have been implemented in R and C for various kernels. The computational and statistical performances of these new algorithms have been assessed through empirical experiments. The runtime of the new algorithms is observed to be faster than that of other considered procedures. Finally, simulations confirmed the higher statistical accuracy of kernel-based approaches to detect changes that are not only in the mean. These simulations also illustrate the flexibility of kernel-based approaches to analyze complex biological profiles made of DNA copy number and allele B frequencies.

1. Introduction

In this paper we consider the multiple change-point detection problem (Brodsky and Darkhovsky, 2013) where the goal is to recover abrupt changes arising in the distribution of a sequence of n independent random variables $X_1, \ldots, X_n$ observed at respective time $t_1 < t_2 < \cdots < t_n$.

State-of-the-art. Many parametric models (Normal, Poisson,...) have been proposed (Hautaniemi et al., 2003; Rigaill et al., 2012; Cleynen and Lebarbier, 2014). These models allow detecting different types of changes: in the mean, in the

* Corresponding author at: Univ. Lille Sciences et Technologies, CNRS, UMR 8524 - Laboratoire Paul Painlevé, F-59000 Lille, France.
E-mail address: alain.celisse@math.univ-lille1.fr (A. Celisse).

https://doi.org/10.1016/j.csda.2018.07.002
0167-9473/

variance and in both the mean and variance (see also Hautaniemi et al., 2003, Jong et al., 2003, Picard et al., 2005). Efficient algorithms and heuristics have been proposed for these models. Some of them scale in $\mathcal{O}(n \log(n))$ or even in $\mathcal{O}(n)$. In practice, these parametric approaches have proven to be successful for various application fields (see for example Hocking et al., 2013, Cleynen et al., 2014a). However one of their main drawbacks is their lack of flexibility. For instance, any change of distributional assumption requires the development of a new dedicated inference scheme.

By contrast, the recently proposed kernel change-point detection approach (Harchaoui and Cappé, 2007; Arlot et al., 2012) is more generic. It has the potential to detect any change arising in the distribution, which is not easily captured by standard parametric models. More precisely in this approach, the observations are first mapped into a Reproducing Kernel Hilbert Space (RKHS) through a kernel function (Aronszajn, 1950). The difficult problem of detecting changes in the distribution is then recast as simply detecting changes in the mean element of observations in the RKHS.

One practical limitation of this kernel-based approach is its considerable computational cost owing to the use of a $n \times n$ Gram matrix combined with a dynamic programming algorithm (Auger and Lawrence, 1989). More precisely (Harchaoui and Cappé, 2007) described a dynamic programming algorithm to recover the best segmentation from 1 to $D_{\max}$ segments. They claim that their algorithm has a $\mathcal{O}(D_{\max} n^2)$ time complexity. However, the latter is not described in full details and its straightforward implementation is not efficient. First, it requires the storage of a $n \times n$ cost matrix (personal communication with the first author of Harchaoui and Cappé (2007) who was kind enough to send us his code). Thus the algorithm has a $\mathcal{O}(n^2)$ space complexity, which is a severe limitation with nowadays sample sizes. For instance analyzing a signal of length $n = 10^5$ requires storing a $10^5 \times 10^5$ matrix of doubles, which takes 80 GB. Second, computing the cost matrix is not straightforward. In fact simply using formula (8) of Harchaoui and Cappé (2007) to compute each term of this cost matrix leads to an $\mathcal{O}(n^4)$ time complexity.

Contributions. The present paper contains several contributions to the computational aspects and the statistical performance of the kernel change-point procedure introduced by Arlot et al. (2012).

The first one is to describe a new algorithm to simultaneously perform the dynamic programming step of Harchaoui and Cappé (2007) and also compute the required elements of the cost matrix on the fly. On the one hand, this algorithm has a complexity of order $\mathcal{O}(D_{\max} n^2)$ in time and $\mathcal{O}(D_{\max} n)$ in space (including both the dynamic programming and the cost matrix computation). We also emphasize that this improved space complexity comes without an increased time complexity. This is a great algorithmic improvement upon the change-point detection approach described by Arlot et al. (2012) since it allows the efficient analysis of signals with up to $n = 10^5$ data-points in a matter of a few minutes on a standard laptop.

On the other hand, our approach is generic in the sense that it works for any positive semidefinite kernels. Importantly one cannot expect to exactly recover the best segmentations from 1 to $D_{\max}$ segments in less than $\mathcal{O}(D_{\max} n^2)$ without additional specific assumptions on the kernel. Indeed, computing the cost of a given segmentation has already a time complexity of order $\mathcal{O}(n^2)$.

It is also noticeable that our algorithm can be applied to other existing strategies such as the so-called ECP (Matteson and James, 2014). To be specific, we show that the *divisive clustering algorithm* it is based on and that provides an approximate solution with a complexity of order $O(n^2)$ in time and space can be replaced by our algorithm that provides the exact solution with the same time complexity but a reduced memory complexity.

Our second contribution is a new algorithm dealing with larger signals ($n > 10^5$) based on a low-rank approximation to the Gram matrix. This computational improvement is possible at the price of an approximation. It returns approximate best segmentations from 1 to $D_{\max}$ segments with a complexity of order $\mathcal{O}(D_{\max} p^2 n)$ in time and $\mathcal{O}((D_{\max} + p)n)$ in space, where p is the rank of the approximation.

The last contribution of the paper is the empirical assessment of the statistical performance of the KCP procedure introduced by Arlot et al. (2012). This empirical analysis is carried out in the biological context of detecting abrupt changes from a two-dimensional signal made of DNA copy numbers and allele B fractions (Lai, 2012). The assessment is done by comparing our approach to state-of-the-art alternatives on resampled real DNA copy number data (Pierre-Jean et al., 2014; Matteson and James, 2014). This illustrates the versatility of the kernel-based approach. To be specific this approach allows the detection of changes in the distribution of such complex signals without explicitly modeling the type of change we are looking for. The described procedure has been implemented in an R package called *KernSeg* (Marot et al., 2018)

The remainder of the paper is organized as follows. In Section 2, we describe our kernel-based framework and detail the connection between detecting abrupt changes in the distribution and model selection as described in Arlot et al. (2012). A slight generalization of the KCP procedure (Arlot et al., 2012) is also derived in Section 2.5 by introducing a new parameter ℓ encoding an additional constraint on the minimal length of any candidate segment. This turns out to be particularly useful in low signal-to-noise ratio settings. The versatility of this kernel-based framework is emphasized in Section 2.6 where it is shown how the ECP approach (Matteson and James, 2014) can be rephrased in terms of kernels. Our main algorithmic improvements are detailed and justified in Section 3. We empirically illustrate the improved runtime of our algorithm and compare it to the ones of ECP and RBS in Section 3.1.3. In Section 3.2 we detail our faster (but approximate) algorithm used to analyze larger profiles ($n > 10^5$). It is based on the combination of a low-rank approximation to the Gram matrix and the binary segmentation heuristic (Yang, 2012). An empirical comparison of the runtimes of the exact and approximate algorithms is provided in Section 3.2.3. Finally, Section 4 illustrates the statistical performance of our kernel-based change-point procedure in comparison with state-of-the-art alternatives in the context of biological signals such as DNA copy numbers and allele B fractions (Lai, 2012).

2. Kernel framework

In this section we recall the framework of Harchaoui and Cappé (2007) where detecting changes in the distribution of a complex signal is rephrased as detecting changes of the mean element of a sequence of points in a Hilbert space. Then we detail the so-called KCP (Kernel Change-point Procedure) (Arlot et al., 2012), which has been proved to be optimal in terms of an oracle inequality.

2.1. Notation

Let $X_1, X_2, \ldots, X_n \in \mathcal{X}$ be a time-series of n independent random variables, where $\mathcal{X}$ denotes any set assumed to be *separable* (Dieuleveut and Bach, 2016) throughout the paper. Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ denote a symmetric positive semi-definite kernel (Aronszajn, 1950), and $\mathcal{H}$ be the associated reproducing kernel Hilbert space (RKHS). We refer to (Berlinet and Thomas-Agnan, 2004) for an extensive presentation about kernels and RKHS. Let us also introduce the canonical feature map $\Phi : \mathcal{X} \to \mathcal{H}$ defined by $\Phi(x) = k(x, \cdot) \in \mathcal{H}$, for every $x \in \mathcal{X}$. This canonical feature map allows to define the inner product on $\mathcal{H}$ from the kernel k, by

$$\forall x, y \in \mathcal{X}, \qquad \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}} = k(x, y). \tag{1}$$

The advantage of kernels. One main advantage of kernels is to enable dealing with complex data of any type provided a kernel can be defined. In particular no vector space structure is required on $\mathcal{X}$. For instance $\mathcal{X}$ can be a set of DNA sequences, a set of graphs or a set of distributions to name but a few examples (see Gartner, 2008 for various instances of $\mathcal{X}$ and related kernels). Therefore, as long as a kernel k can be defined on $\mathcal{X}$, any element $x \in \mathcal{X}$ is mapped, through the canonical feature map Φ, to an element of the Hilbert space $\mathcal{H}$. This provides a unified way to deal with different types of (simple or complex) data. Then for every index $1 \leq t \leq n$, let us note

$$Y_t = \Phi(X_t) \in \mathcal{H}. \tag{2}$$

From now on, we will only consider the following sequence $Y_1, \ldots, Y_n \in \mathcal{H}$ of independent Hilbert-valued random vectors.

The kernel trick. As a space of functions from $\mathcal{X}$ to $\mathbb{R}$, the RKHS $\mathcal{H}$ can be infinite dimensional. From a computational perspective one could be worried that manipulating such objects is computationally prohibitive. However this is not the case and our algorithm relies on the so-called *kernel trick*, which consists in translating any inner product in $\mathcal{H}$ in terms of the kernel k by use of Eq. (1). For every $1 \leq i, j \leq n$, it results

$$\langle Y_i, Y_j \rangle_{\mathcal{H}} = k(X_i, X_j) = \mathbf{K}_{i,j},$$

where $\mathbf{K}_{i,j}$ denotes the (i, j)-th coefficient of the $n \times n$ Gram matrix $\mathbf{K} = \big[\, k(X_i, X_j) \,\big]_{1 \leq i,j \leq n}$.

2.2. Detecting changes in the distribution using kernels

Let us consider the model introduced by Arlot et al. (2012), which connects every Y_t to its "mean" $\mu_t^\star \in \mathcal{H}$ by

$$\forall\, 1 \leq t \leq n, \qquad Y_t = \Phi(X_t) = \mu_t^\star + \epsilon_t \in \mathcal{H}, \tag{3}$$

where $\mu_t^\star$ denotes the *mean element* associated with the distribution $\mathbb{P}_{X_t}$ of X_t, and $\epsilon_t = Y_t - \mu_t^\star$. Let us also recall (Ledoux and Talagrand, 1991) that if $\mathcal{X}$ is separable and $\mathbb{E}\,[\, k(X_t, X_t)\,] < +\infty$, then $\mu_t^\star$ exists and is defined as the unique element in $\mathcal{H}$ such that

$$\forall f \in \mathcal{H}, \quad \langle \mu_t^\star, f \rangle_{\mathcal{H}} = \mathbb{E}\langle \Phi(X_t), f \rangle_{\mathcal{H}}. \tag{4}$$

For characteristic kernels (Sriperumbudur et al., 2010), a change in the distribution of X_t implies a change in the mean element $\mu_t^\star$, that is

$$\forall\, 1 \leq i \neq j \leq n, \qquad \mathbb{P}_{X_i} \neq \mathbb{P}_{X_j} \Rightarrow \mu_i^\star \neq \mu_j^\star, \tag{5}$$

the converse implication being true by definition of $\mu_t^\star$ in Eq. (4). The idea behind kernel change-point detection (Arlot et al., 2012) is to translate the problem of detecting changes in the distribution into detecting changes in the mean of Hilbert-valued vectors.

Remark 1. When considering $\mathcal{X} \subset \mathbb{R}^q$ for some integer $q > 0$, several classical kernels are characteristic. For instance,

- The Gaussian kernel: $k(x, y) = e^{-\|x-y\|^2/\delta}$, with $x, y \in \mathbb{R}^q$ and $\delta > 0$,
- The Laplace kernel: $k(x, y) = e^{-\|x-y\|/\delta}$, with $x, y \in \mathbb{R}^q$ and $\delta > 0$,
- The exponential kernel: $k(x, y) = e^{-\langle x, y \rangle_q/\delta}$, with $x, y \in \mathbb{R}^q$ and $\delta > 0$,

where $\|\cdot\|$ and $\langle\cdot,\cdot\rangle_q$ respectively denote the usual Euclidean norm and inner product in $\mathbb{R}^q$. The energy-based kernel discussed in Section 2.6 is also a characteristic kernel (see Lemma 1 in Matteson and James (2014)). However, with more general sets $\mathcal{X}$, building a characteristic kernel is challenging as illustrated by Sriperumbudur et al. (2010) and Christmann and Steinwart (2010).

Let us also notice that the procedure developed by Arlot et al. (2012) can be seen as a "kernelized version" of the procedure proposed by Lebarbier (2005), which was originally designed to detect changes in the mean of real-valued variables.

2.3. Statistical framework

From Eq. (5) it results that any sequence of abrupt changes in the distribution over time corresponds to a sequence of D^* true change-points $1 = \tau_1^* < \tau_2^* < \cdots < \tau_{D^*}^* \leq n$ (with $\tau_{D^*+1}^* = n+1$ by convention) such that

$$\mu_1^* = \cdots = \mu_{\tau_2^*-1}^* \neq \mu_{\tau_2^*}^* = \cdots = \mu_{\tau_{D^*}^*-1}^* \neq \mu_{\tau_{D^*}^*}^* = \cdots = \mu_n^*.$$

In other words we get that $\mu^* = (\mu_1^*, \ldots, \mu_n^*)' \in \mathcal{H}^n$ is piecewise constant.

From a set of D candidate change-points $1 = \tau_1 < \cdots < \tau_D \leq n$, let τ be defined by

$$\tau = (\tau_1, \tau_2, \ldots, \tau_{D-1}, \tau_D)\,,$$

with the convention $\tau_1 = 1$ and $\tau_{D+1} = n+1$. With a slight abuse of notation, we also call τ the segmentation of $\{1, \ldots, n\}$ associated with the change-points $1 = \tau_1 < \cdots < \tau_D \leq n$. The estimator $\hat{\mu}^\tau = \left(\hat{\mu}_1^\tau, \ldots, \hat{\mu}_n^\tau\right)' \in \mathcal{H}^n$ of $\mu^* = \left(\mu_1^*, \ldots, \mu_n^*\right)'$ proposed by Arlot et al. (2012) is defined by

$$\forall\, 1 \leq i \leq D, \quad \forall\, t \in \{\tau_i, \ldots, \tau_{i+1} - 1\}\,, \quad \hat{\mu}_t^\tau = \frac{1}{\tau_{i+1} - \tau_i} \sum_{t'=\tau_i}^{\tau_{i+1}-1} Y_{t'}.$$

The performance of $\hat{\mu}^\tau$ is measured by the quadratic risk:

$$\mathcal{R}(\hat{\mu}^\tau) = \mathbb{E}\left[\left\|\mu^* - \hat{\mu}^\tau\right\|_{\mathcal{H},n}^2\right] = \mathbb{E}\left[\sum_{i=1}^n \left\|\mu_i^* - \hat{\mu}_i^\tau\right\|_{\mathcal{H}}^2\right],$$

where $\|\cdot\|_{\mathcal{H}}$ denotes the norm in the Hilbert space $\mathcal{H}$.

2.4. Model selection

If the signal-to-noise ratio is small, Arlot et al. (2012) emphasized that all true change-points cannot be recovered without including false change-points. This leads them to define the best segmentation τ^* (for a finite sample size) as

$$\tau^* = \operatorname*{arg\,min}_{\tau \in \mathcal{T}_n} \left\|\mu^* - \hat{\mu}^\tau\right\|_{\mathcal{H},n},$$

where $\mathcal{T}_n$ denotes the collection of all possible segmentations τ of $\{1, \ldots, n\}$ with at most $D_{\max}$ segments. When the signal-to-noise ratio is large enough, τ^* coincides with the true segmentation.

As a surrogate to the previous criterion which is not computable in practice because μ^* is unknown, Arlot et al. (2012) optimize the following penalized criterion

$$\hat{\tau} = \operatorname*{arg\,min}_{\tau \in \mathcal{T}_n} \left\{\left\|Y - \hat{\mu}^\tau\right\|_{\mathcal{H},n}^2 + \mathrm{pen}(\tau)\right\}, \qquad \text{with} \quad \mathrm{pen}(\tau) = c_1 D_\tau + c_2 \log\binom{n-1}{D_\tau - 1}, \tag{6}$$

where $Y = (Y_1, \ldots, Y_n)^\top \in \mathbb{R}^n$, $c_1, c_2 > 0$ are constants to be fixed, and D_τ denotes the number of segments of the segmentation τ. Since this penalty only depends on τ through D_τ, optimizing (6) can be formulated as a two-step procedure. The first step consists in solving:

$$\forall\, 1 \leq D \leq D_{\max}, \quad \hat{\tau}_D = \operatorname*{arg\,min}_{\tau \in \mathcal{T}_n^D} \left\|Y - \hat{\mu}^\tau\right\|_{\mathcal{H},n}^2, \tag{7}$$

where $\mathcal{T}_n^D$ denotes the set of segmentations with D segments. This optimization problem, which is usually solved by dynamic programming (Auger and Lawrence, 1989; Rigaill et al., 2012), is computationally hard since the cardinality of $\mathcal{T}_n^D$ is $\binom{n-1}{D-1}$. The second step is to straightforwardly optimize:

$$\hat{D} = \operatorname*{arg\,min}_{1 \leq D \leq D_{\max}} \left\{\left\|Y - \hat{\mu}_{\hat{\tau}_D}\right\|_{\mathcal{H},n}^2 + \mathrm{pen}(\hat{\tau}_D)\right\} \qquad \text{and} \quad \hat{\tau} = \hat{\tau}_{\hat{D}}. \tag{8}$$

The right-most term in the penalty (6) accounts for the number of candidate segmentations with D segments (see the comments of Theorem 2 in Arlot et al. (2012)). Intuitively this term balances the tendency of the estimator (7) to overfit because of the large number of candidate segmentations.

Remark 2. It is important to notice that the above two-step procedure depends on the hyper-parameter $D_{\max}$, which is the maximum number of segments of the candidate segmentations.

In fact choosing an appropriate $D_{\max}$ is related to the calibration of constants c_1 and c_2 in the penalty term of (6). Since the optimal values of c_1 and c_2 depend (at least) on the variance of the signal at hand, they have to be calibrated in a data-driven way. Here they have been calibrated by using the so-called *slope heuristic* technique described in Arlot et al. (2012) (see also the numerical experiments in Section 4 for more details). In particular $D_{\max}$ has to be chosen large enough to make the slope heuristic work well. Given some prior knowledge of an adequate range of values for D taking D_{max} to be 10 to 20 times larger than that seems to work well in practice. Typically for copy number data (see Section 4.1.1) one rarely expects more than 10 change-points per chromosome and taking $D_{max} \approx 100$ or 200 often makes sense.

From a theoretical point of view, this model selection procedure has been proved to be optimal in terms of an oracle inequality by Arlot et al. (2012). This is the usual non-asymptotic optimality result for model selection procedures (Birgé and Massart, 2007). This procedure has also been proved to provide consistent estimates of the change-points (Garreau and Arlot, 2016). However, from a computational point of view, the first step (*i.e.* solving Eq. (7)) remains challenging. Indeed existing dynamic programming algorithms are time and space consuming when used in the kernel framework as will be clarified in Section 3.1.1. The main purpose of the present paper is to provide a new computationally efficient algorithm to solve Eq. (7). Our new algorithm has a reduced space and time complexity and allows the analysis of signals larger than $n = 10^4$.

2.5. Low signal-to-noise and minimal length of a segment

In settings where the signal-to-noise ratio is weak (see for instance Fig. 4 where the tumor percentage is low) change-point detection procedures are more likely to put changes in noisy regions. This results in overfitting and meaningless small segments (Arlot and Celisse, 2011). A common solution is to include a constraint on the minimum length ℓ of segments. For instance by default ECP enforces that the estimated segmentation has segments with at least $\ell = 30$ points (James and Matteson, 2013).

One important side effect of this constraint on ℓ is that the total number of candidate segmentations with D segments quickly decreases with ℓ. Therefore the penalty in (6) has to be modified.

The following lemma gives the cardinality of this set of segmentations.

Lemma 1. *Let $\mathcal{T}_n^\ell(D)$ denote the set of segmentations of $(1, \ldots, n)$ in exactly $D \geq 1$ segments such that the length of each segment is at least $\ell \geq 1$. Then the cardinality of $\mathcal{T}_n^\ell(D)$ satisfies*

$$\text{Card}\left(\mathcal{T}_n^\ell(D)\right) = \binom{n - D(\ell-1) - 1}{D-1}.$$

Let us notice that if $\ell = 1$, one recovers the usual cardinality that is used in the penalty (see Eq. (6)). As an illustration of the influence of the constraint on ℓ, let us consider the set-up where $n = 100$, $D = 10$, and $\ell = 10$. Then the size of the unconstrained set of segmentations with 10 segments $\mathcal{T}_{100}(10) = \mathcal{T}_{100}^1(10)$ is $\text{Card}\left(\mathcal{T}_{100}(10)\right) \approx 1.7 \cdot 10^{12}$, whereas the constrained set $\mathcal{T}_{100}^{10}(10)$ is smaller since its cardinality is equal to 1.

Proof of Lemma 1. The proof consists in showing that there is a one-to-one mapping between the set $\mathcal{T}_n^\ell(D)$ of segmentations of $(1, \ldots, n)$ with D segments of length at least $\ell \geq 1$, and the set $\mathcal{S}_n^\ell(D)$ of segmentations of $(1, \ldots, n - D(\ell-1))$ with D (non-empty) segments.

Let us consider one segmentation $\tau \in \mathcal{T}_n^\ell(D)$. Since each segment of τ is of length at least ℓ, let us remove $\ell - 1$ points from the left edge of each of the D segments. Then the resulting segmentation belongs to $\mathcal{S}_n^\ell(D)$.

Conversely, take one segmentation $\tau \in \mathcal{S}_n^\ell(D)$. Then each segment of τ contains at least one point. Adding $\ell - 1$ points to each segment (from the left edge) clearly provides a segmentation with D segments of length at least ℓ. This allows to conclude. □

This leads to the following generalized change-points detection procedure involving a constraint on the minimum length $\ell \geq 1$ of each segment.

Step 1: Solve

$$\forall\, 1 \leq D \leq D_{\max}, \quad \hat{\tau}_D^\ell = \underset{\tau \in \mathcal{T}_n^\ell(D)}{\arg\min} \left\| Y - \hat{\mu}^\tau \right\|_{\mathcal{H},n}^2, \tag{9}$$

where $\mathcal{T}_n^\ell(D)$ denotes the set of segmentations with D segments of length at least $\ell \geq 1$.

Step 2: Find

$$\hat{D}^\ell = \underset{D}{\arg\min} \left\{ \left\| Y - \hat{\mu}^{\hat{\tau}_D^\ell} \right\|_{\mathcal{H},n}^2 + \text{pen}_\ell(\hat{\tau}_D^\ell) \right\} \quad \text{and} \quad \hat{\tau}^\ell = \hat{\tau}_{\hat{D}^\ell}^\ell, \tag{10}$$

where $\text{pen}_\ell(\tau) = c_1 D_\tau + c_2 \log\left(\binom{n - D_\tau(\ell-1) - 1}{D_\tau - 1}\right)$.

Let us emphasize that this generalized procedure, including this additional parameter $\ell \geq 1$, is very similar the previous one. The optimization step of Eq. (9) is performed by dynamic programming up to a minor change of implementation. The optimization of the second step (10) remains unchanged except it involves a slightly different penalty shape. The tuning of the constant c_1 and c_2 is still made by the slope heuristic (see Section 4).

2.6. A link between kernels and energy-based distances

Note that the kernel-based framework developed in Sections 2.1–2.3 is very general. Various existing procedures can be rephrased in this framework by use of a particular kernel. For example the procedure of Lebarbier (2005), which is devoted to the detection of changes in the mean of a one-dimensional real-valued signal, reduces to ours by use of the linear kernel. More interestingly the procedure called ECP developed by Matteson and James (2014) and that relies on an energy-based distance to detect changes in multivariate distributions, can also be integrated into our framework using a particular kernel as explained in what follows.

For every $\alpha \in (0, 2)$, let us define $\rho_\alpha(x, y) = \|x - y\|^\alpha$, where $x, y \in \mathbb{R}^q$ and $\|\cdot\|$ denotes the Euclidean norm on $\mathbb{R}^q$. Then ρ_α is a semimetric of negative type (Berg et al., 1984), and for any independent random variables $X, X', Y, Y' \in \mathbb{R}^q$ with respective probability distributions satisfying $P_X = P_{X'}$ and $P_Y = P_{Y'}$, Matteson and James (2014) introduce the energy-based distance:

$$\mathcal{E}(X, Y; \alpha) = 2E\left[\rho_\alpha(X, Y)\right] - E\left[\rho_\alpha(X, X')\right] - E\left[\rho_\alpha(Y, Y')\right], \tag{11}$$

with the assumption that $\max\left(E\left[\rho_\alpha(X, X')\right], E\left[\rho_\alpha(X, Y)\right], E\left[\rho_\alpha(Y, Y')\right]\right) < +\infty$. Then following Sejdinovic et al. (2013) and for every $x_0 \in \mathbb{R}^q$, we define

$$k_\alpha^{x_0}(x, y) = \frac{1}{2}\left[\rho_\alpha(x, x_0) + \rho_\alpha(y, x_0) - \rho_\alpha(x, y)\right], \tag{12}$$

which is a positive semi-definite kernel leading to an RKHS $\mathcal{H}_\alpha^0$. Plugging this in Eq. (11), one can easily check that

$$\begin{aligned}
\mathcal{E}(X, Y; \alpha) &= 2E\left[\rho_\alpha(X, x_0) + \rho_\alpha(Y, x_0) - 2k_\alpha^{x_0}(X, Y)\right] \\
&\quad - E\left[\rho_\alpha(X, x_0) + \rho_\alpha(X', x_0) - 2k_\alpha^{x_0}(X, X')\right] \\
&\quad - E\left[\rho_\alpha(Y, x_0) + \rho_\alpha(Y', x_0) - 2k_\alpha^{x_0}(Y, Y')\right] \\
&= 2E\left[k_\alpha^{x_0}(X, X') + k_\alpha^{x_0}(Y, Y') - 2k_\alpha^{x_0}(X, Y)\right] \\
&= 2\left\|\mu_{P_X}^\alpha - \mu_{P_Y}^\alpha\right\|_{\mathcal{H}_\alpha^0}^2,
\end{aligned}$$

where $\mu_{P_X}^\alpha, \mu_{P_Y}^\alpha \in \mathcal{H}_\alpha^0$ respectively denote the mean elements of the distributions P_X and P_Y, and $\|\cdot\|_{\mathcal{H}_\alpha^0}$ is the norm in $\mathcal{H}_\alpha^0$.

An important consequence of this derivation is that the exact and approximate algorithms described in Section 3 immediately apply to procedures relying on the optimization of the energy-based distance $\mathcal{E}(X, Y; \alpha)$. This is all the more remarkable as our *exact optimization algorithm* has a lower memory complexity than the *approximate optimization algorithm* of ECP (with a similar time complexity). Therefore, for the same computational time, our exact optimization algorithm could replace the approximate algorithm used in ECP. One can also emphasize that our approximating algorithm which is even faster could also be used (see its description in Section 3.2). This particular energy-based kernel, which is a characteristic kernel, has been used in our simulation experiments as well (Section 4.2.1).

3. New algorithms

In this section we first show how to avoid the preliminary calculation of the cost matrix required by Harchaoui and Cappé (2007) to apply dynamic programming. The key idea is to compute the elements of the cost matrix on the fly when they are required by the dynamic programming algorithm. Roughly, this can be efficiently done by reordering the loops involved in Algorithm 1 proposed in Harchaoui and Cappé (2007). This leads to the new exact Algorithm 3. It has a reduced space complexity of order $\mathcal{O}(n)$ compared to $\mathcal{O}(n^2)$ for the one used in Harchaoui and Cappé (2007). Note that including the constraint (introduced in Section 2.5) on the segment sizes mostly change the index of the **for** loops in Algorithm 3. We choose to describe the algorithm in the unconstrained version (7) to ease the understanding.

Second, we provide a faster but approximate optimization algorithm (Section 3.2), which enjoys a smaller complexity of order $\mathcal{O}(D_{\max} n)$ in time. It combines a low-rank approximation to the Gram matrix and the use of the binary segmentation heuristic (Section 3.2.2). This approximating algorithm allows the analysis of very large signals ($n \geq 10^6$).

3.1. New efficient algorithm to recover the best segmentation from the Gram matrix

As exposed in Section 2.4, the main computational cost of the change-point detection procedure results from Eq. (7), that is recovering the best segmentation with $1 \leq D \leq D_{\max}$ segments and solving

$$\begin{aligned}
\mathbf{L}_{D,n+1} &= \min_{\tau \in \mathcal{T}_D} \left\|Y - \hat{\mu}^\tau\right\|_{\mathcal{H},n}^2 && \text{(best fit to the data)} \\
\hat{\tau}_D &= \arg\min_{\tau \in \mathcal{T}_D} \left\|Y - \hat{\mu}^\tau\right\|_{\mathcal{H},n}^2 && \text{(best segmentation)}
\end{aligned} \tag{13}$$

for every $1 \leq D \leq D_{\max}$, where $\mathcal{T}_D$ denotes the collection of segmentations of $\{1, \ldots, n\}$ with D segments. This challenging step involves the use of dynamic programming (Bellman, 1961; Auger and Lawrence, 1989), which provides the exact solution to the optimization problem (13). Let us first provide some details on the usual way dynamic programming is implemented.

3.1.1. Limitations of the standard dynamic programming algorithm for kernels

Let τ denote a segmentation in D segments (with the convention that $\tau_1 = 1$ and $\tau_{D+1} = n + 1$). For any $1 \leq d \leq D$, the segment $\{\tau_d, \ldots, \tau_{d+1} - 1\}$ of the segmentation τ has a cost that is equal to

$$C_{\tau_d, \tau_{d+1}} = \sum_{i=\tau_d}^{\tau_{d+1}-1} k(X_i, X_i) - \frac{1}{\tau_{d+1} - \tau_d} \sum_{i=\tau_d}^{\tau_{d+1}-1} \sum_{j=\tau_d}^{\tau_{d+1}-1} k(X_i, X_j). \tag{14}$$

Then the cost of the segmentation τ is given by

$$\|Y - \hat{\mu}^{\tau}\|^2_{\mathcal{H},n} = \sum_{d=1}^{D} C_{\tau_d, \tau_{d+1}},$$

which is clearly *segment additive* (Harchaoui and Cappé, 2007; Arlot et al., 2012).

Dynamic programming solves (13) for all $1 \leq D \leq D_{\max}$ by applying the following update rules

$$\forall\, 2 \leq D \leq D_{\max}, \quad \mathbf{L}_{D,n+1} = \min_{\tau \leq n} \{\, \mathbf{L}_{D-1,\tau} + C_{\tau,n+1} \,\}, \tag{15}$$

which exploits the property that the optimal segmentation in D segments over $\{1, \ldots, n\}$ can be computed from optimal ones with $D - 1$ segments over $\{1, \ldots, \tau\}$ ($\tau \leq n$). Making the key assumption that *the cost matrix* $\left\{C_{i,j}\right\}_{1 \leq i,j \leq n+1}$ *has been stored*, we can compute $\mathbf{L}_{D,n+1}$ with Algorithm 1.

Algorithm 1 Basic use of Dynamic Programming

1: **for** $D = 2$ to $D_{\max}$ **do**
2: **for** $\tau' = D$ to n **do**
3: $\mathbf{L}_{D,\tau'+1} = \min_{\tau \leq \tau'} \{\, \mathbf{L}_{D-1,\tau} + C_{\tau,\tau'+1} \,\}$
4: **end for**
5: **end for**

This algorithm is used by Harchaoui and Cappé (2007) and suffers two main limitations. First it assumes that the $C_{\tau,\tau'}$ have been already computed, and does not take into account the computational cost of its calculation. Second, it stores all $C_{\tau,\tau'}$ in a $\mathcal{O}(n^2)$ matrix, which is memory expensive.

A quick inspection of the algorithm reveals that the main step at Line 3 requires $\mathcal{O}(\tau')$ operations (assuming the $C_{i,j}$s have been already computed). Therefore, with the two **for** loops we get a complexity of $\mathcal{O}(D_{\max} n^2)$ in time. Note that without any particular assumption on the kernel $k(\cdot, \cdot)$, computing $\left\|Y - \hat{\mu}^{\tau}\right\|^2_{\mathcal{H},n}$ for a given segmentation τ is already of order $\mathcal{O}(n^2)$ in time since it involves summing over a quadratic number of terms of the Gram matrix (see Eq. (14)). Therefore, there is no hope to solve (13) exactly in less than quadratic time without additional assumptions on the kernel.

From Eq. (14) let us also remark that computing each $C_{i,j}$ ($1 \leq i < j \leq n$) naively requires itself a quadratic number of operations. Computing the whole cost matrix would require a complexity $\mathcal{O}(n^4)$ in time. Taking this into account, the dynamic programming step (Line 3 of Algorithm 1) is not the limiting factor and the overall time complexity of Algorithm 1 is $\mathcal{O}(n^4)$.

Finally, let us emphasize that this high computational burden is not specific of detecting change-points with kernels. It is rather representative of most learning procedures based on reproducing kernels and the associated Gram matrix (Bach, 2013).

3.1.2. Improved use of dynamic programming for kernel methods

Reducing space complexity. From Algorithm 1, let us first remark that each $C_{\tau,\tau'}$ is used several times along the algorithm. A simple idea to avoid that is to swap the two **for** loops in Algorithm 1. This leads to the following modified Algorithm 2, where each column $C_{\cdot,\tau'+1}$ of the cost matrix is only used once unlike in Algorithm 1.

Algorithm 2 Improved space complexity

1: **for** $\tau' = 2$ to n **do**
2: **for** $D = 2$ to $\min(\tau', D_{\max})$ **do**
3: $\mathbf{L}_{D,\tau'+1} = \min_{\tau \leq \tau'} \{\, \mathbf{L}_{D-1,\tau} + C_{\tau,\tau'+1} \,\}$
4: **end for**
5: **end for**

Importantly swapping the two **for** loop does not change the output of the algorithm and does not induce any additional calculations. Furthermore, at step τ' of the first **for** loop we do not need the whole $n \times n$ cost matrix to be stored, but only the column $C_{\cdot,\tau'+1}$ of the cost matrix. This column is of size at most $\mathcal{O}(n)$.

Algorithm 2 finally requires storing coefficients $\left\{\mathbf{L}_{d,\tau}\right\}_{1\leq d\leq D,\ 2\leq\tau\leq n}$ that are computed along the algorithm as well as successive column vectors $\left\{C_{\cdot,\tau}\right\}_{2\leq\tau\leq n}$ (of size at most n) of the cost matrix. This leads to an overall complexity of $\mathcal{O}(D_{\max}n)$ in space. The only remaining problem is to compute these successive column vectors efficiently. Let us recall that a naive implementation is prohibitive: each coefficient of the column vector can be computed in $\mathcal{O}(n^2)$, which would lead to $\mathcal{O}(n^3)$ to get the entire column.

Iterative computation of the columns of the cost matrix. The last ingredient of our final exact algorithm is the efficient computation of each column vector $\left\{C_{\cdot,\tau}\right\}_{2\leq\tau\leq n}$. Let us explain how to iteratively compute each vector in linear time.

First it can be easily observed that Eq. (14) can be rephrased as follows

$$C_{\tau,\tau'} = \sum_{i=\tau}^{\tau'-1}\left(k\left(X_i, X_i\right) - \frac{A_{i,\tau'}}{\tau'-\tau}\right) = D_{\tau,\tau'} - \frac{1}{\tau'-\tau}\sum_{i=\tau}^{\tau'-1} A_{i,\tau'},$$

where $D_{\tau,\tau'} = \sum_{i=\tau}^{\tau'-1} k\left(X_i, X_i\right)$, and $A_{i,\tau'}$ is given by

$$A_{i,\tau'} = -k(X_i, X_i) + 2\sum_{j=i}^{\tau'-1} k(X_i, X_j), \quad \text{if } i < \tau',$$

and by further using $A_{j,j} = -k(X_j, X_j)$ for any $1 \leq j \leq n$. Second, both $D_{\tau,\tau'}$ and $\left\{A_{i,\tau'}\right\}_{i\leq\tau'}$ can be iteratively computed from τ' to $\tau'+1$ by use of the two following equations:

$$D_{\tau,\tau'+1} = D_{\tau,\tau'} + k(X_{\tau'}, X_{\tau'}), \qquad \text{and} \qquad A_{i,\tau'+1} = A_{i,\tau'} + 2k(X_{\tau'}, X_{\tau'}),\ \forall i \leq \tau'.$$

Therefore, as long as computing $k(x_i, x_j)$ requires $\mathcal{O}(1)$ operations, updating from τ' to $\tau'+1$ requires $\mathcal{O}\left(\tau'\right)$ operations.

Remark 3. Note that for many classical kernels, computing $k(x_i, x_j)$ is indeed $\mathcal{O}(1)$ in time. If $x_i \in \mathbb{R}^q$ with q a positive integer being negligible with respect to other influential quantities such as $D_{\max}$ and n, several kernels such as the Gaussian, Laplace, or χ^2 ones lead to a $\mathcal{O}(q) = \mathcal{O}(1)$ time complexity for evaluating $k(x_i, x_j)$. By contrast when the size of q is no longer negligible, the resulting time complexity is multiplied by a factor q, which corroborates the intuition that the computational complexity increases with the "complexity" of the objects in $\mathcal{X}$.

This update rule leads us to the following Algorithm 3, where each column $C_{\cdot,\tau'+1}$ in the first **for** loop is computed only once:

Algorithm 3 Improved space and time complexity (*Kernseg*)

1: **for** $\tau' = 2$ to n **do**
2: Compute the $(\tau'+1)$-th column $C_{\cdot,\tau'+1}$ from $C_{\cdot,\tau'}$
3: **for** $D = 2$ to $\min(\tau', D_{\max})$ **do**
4: $\mathbf{L}_{D,\tau'+1} = \min_{\tau\leq\tau'}\{\mathbf{L}_{D-1,\tau} + C_{\tau,\tau'+1}\}$
5: **end for**
6: **end for**

From a computational point of view, each step of the first **for** loop in Algorithm 3 requires $\mathcal{O}(\tau')$ operations to compute $C_{\cdot,\tau'+1}$ and at most $\mathcal{O}(D_{\max}\tau')$ additional operations to perform the dynamic programming step at Line 4. Then the overall complexity is $\mathcal{O}(D_{\max}n^2)$ in time and $\mathcal{O}(D_{\max}n)$ in space. This should be compared to the $\mathcal{O}(D_{\max}n^4)$ time complexity of the naive calculation of the cost matrix and to the $\mathcal{O}(n^2)$ space complexity of the standard Algorithm 1 from Harchaoui and Cappé (2007).

3.1.3. Runtimes comparison to other implementations

The purpose of the present section is to perform the comparison between Algorithm 3 and other competitors to illustrate their performances as the sample size increases with $D_{\max} = 100$. For all these simulation experiments we simulated data following a Gaussian distribution with mean 0 and variance 1. All simulations where run on a laptop with 7.7Gb of RAM and 4 Core CPU with 2.1 GHz each.

The first comparison has been carried out between Algorithm 3 and the naive computation of the cost matrix (Algorithm 1, KCP). These two algorithms have been implemented in C and packaged in R. Results for these algorithms are reported in Fig. 1 in the top-left panel. Unsurprisingly, our $\mathcal{O}(n^2)$ algorithm *Kernseg* used with a Gaussian kernel (KS.Gau) is faster than a $\mathcal{O}(n^4)$ computation of the cost matrix (called KCP) even for very small sample sizes ($n \leq 320$).

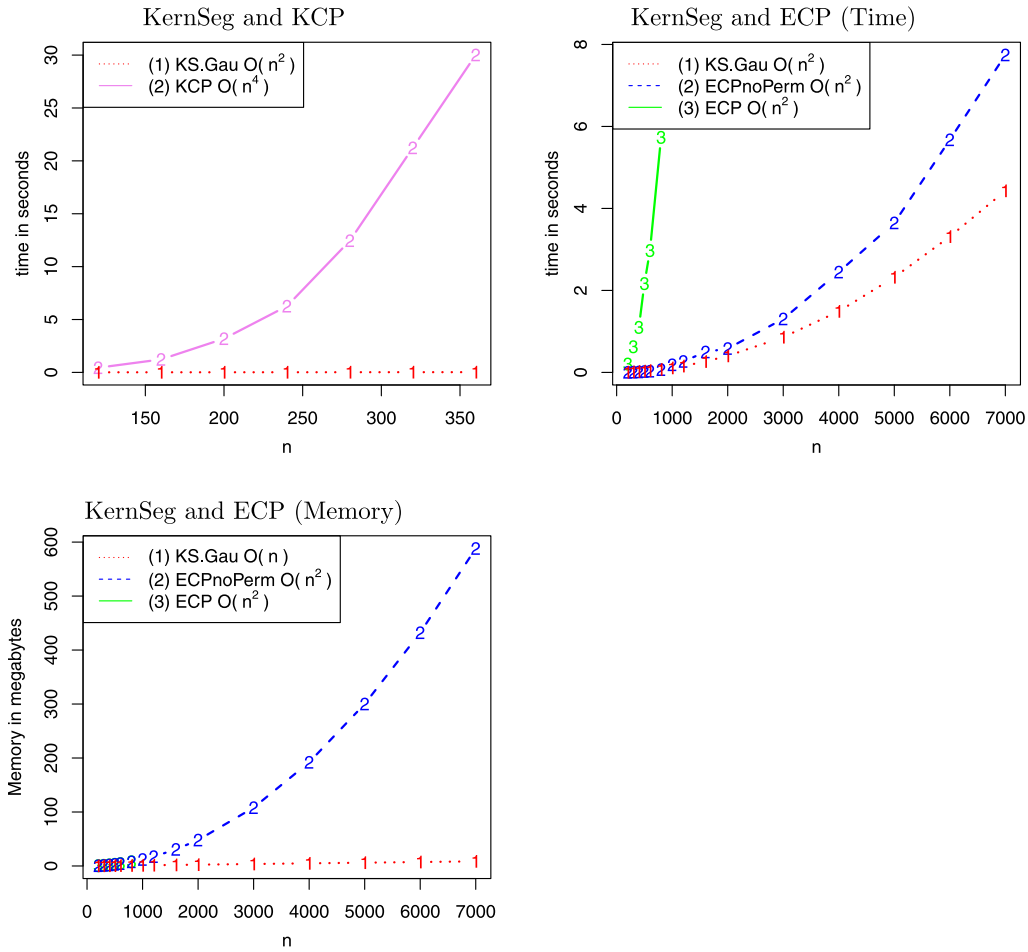


Fig. 1. (Top-Left) Average runtime in seconds of Algorithm 3 with a Gaussian Kernel (KS.Gau) as a function of the length of the signal (n) for $D_{\max} = 100$ (1-red) and a $\mathcal{O}(n^4)$ computation of the cost matrix (2-violet). (Top-Right) Average runtime in seconds of Algorithm 3 with a Gaussian Kernel (KS.Gau) as a function of the length of the signal (1-red) and of ECP without permutation (2-blue) and ECP with the default number of permutations (3-green). (Bottom-Left) Memory in mega-bytes of Algorithm 3 as a function of the length of the signal (1-red) and of ECP without permutation (2-blue) and ECP with the default number of permutations (3-green). The memory performances of ECP with or without permutation are almost exactly the same. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Second, we also compared the runtime of *Kernseg* (Algorithm 3) with that of ECP discussed in Section 2.6 implemented in the R package (James and Matteson, 2013) (see the top-right panel of Fig. 1). Since ECP is based on the binary segmentation heuristic applied to an energy-based distance, its worst-case complexity is at most $\mathcal{O}(D_{\max}n^2)$ in time, which is the same as that of *Kernseg*. Note also that the native implementation of ECP involves an additional procedure relying on permutations to choose the number of change-points. If B denotes the number of permutations, the induced complexity is then $\mathcal{O}(BD_{\max}n^2)$ in time. To be fair, we compared *Kernseg* with a Gaussian Kernel (KS.Gau) and ECP with and without the permutation layer. Finally it is also necessary to emphasize that unlike *Kernseg*, ECP does not provide the exact but only an approximate solution to the optimization problem (13). Results are summarized in Fig. 1 in the top-right panel. It illustrates that our exact algorithm (*Kernseg*) has a quadratic complexity similar to that of ECP with and without permutations. Our algorithm is the overall fastest one even for small sample size ($n < 1\,000$). Although this probably results from implementation differences, it is still noteworthy since *Kernseg* is exact unlike ECP.

Finally, Fig. 1 (Bottom-Left) illustrates the worse memory use of ECP (with and without any permutations) as compared to that of the exact KS.Gau (*Kernseg* used with the Gaussian kernel). ECP has an $\mathcal{O}(n^2)$ space complexity, while KS.Gau is $\mathcal{O}(n)$. For n larger than 10^4 the quadratic space complexity of ECP is a limitation since several Gb of RAM are required.

3.2. Approximating the Gram matrix to speed up the algorithm

In Section 3.1.2, we described an improved algorithm called *Kernseg*, which carefully combines dynamic programming with the computation of the cost matrix elements. This new algorithm (Algorithm 3) provides the exact solution to the optimization problem given by Eq. (13). However without any further assumption on the underlying reproducing kernel, this algorithm only achieves the complexity $\mathcal{O}(n^2)$ in time, which is a clear limitation with large scale signals ($n \geq 10^5$). Note also that this limitation results from the use of general positive semi-definite kernels (and related Gram matrices) and cannot

be improved by existing algorithms to the best of our knowledge. For instance, the binary segmentation heuristic (Fryzlewicz, 2014), which is known to be computationally efficient for parametric models, suffers the same $\mathcal{O}(n^2)$ time complexity when used in the reproducing kernel framework (see Section 3.2.2).

Let us remark however that for some particular kernels it is possible to reduce this time complexity. For example for the linear one $k(x, y) = \langle x, y\rangle_q$, $x, y \in \mathbb{R}^q$, one can use the following trick

$$\sum_{1\le i\neq j\le n} k(X_i, X_j) = \sum_{1\le i\le n} \left\langle X_i, \sum_{j=1}^{n} X_j - X_i \right\rangle_q = \left\| \sum_{i=1}^{n} X_i \right\|^2 - \sum_{i=1}^{n} \|X_i\|^2 , \tag{16}$$

where $\|\cdot\|$ denotes the Euclidean norm in $\mathbb{R}^q$.

The purpose of the present section is to describe a versatile strategy (*i.e* applicable to any kernel) relying on a low-rank approximation to the Gram matrix (Williams and Seeger, 2001; Smola and Schölkopf, 2000; Fine et al., 2001). This approximation allows to considerably reduce the computation time by exploiting (16). Note however that the resulting procedure achieves this lower time complexity at the price of only providing an approximation to the exact solution to (13) (unlike the algorithm described in Section 3.1.2).

3.2.1. Low-rank approximation to the Gram matrix

The main idea is to follow the same strategy as the one described by Drineas and Mahoney (2005) to derive a low-rank approximation to the Gram matrix $\mathbf{K} = \left\{\mathbf{K}_{i,j}\right\}_{1\le i,j\le n}$, where $\mathbf{K}_{i,j} = k(X_i, X_j)$.

Assuming $\mathbf{K}$ has rank $\mathrm{rk}(\mathbf{K}) \ll n$, we could be tempted to compute the best rank approximation to $\mathbf{K}$ by computing the $\mathrm{rk}(\mathbf{K})$ largest eigenvalues (and corresponding eigenvectors) of $\mathbf{K}$. However such computations induce a $\mathcal{O}(n^3)$ time complexity which is prohibitive.

Instead, Drineas and Mahoney (2005) suggest applying this idea on a square sub-matrix of $\mathbf{K}$ with size $p \ll n$. For any subsets $I, J \subset \{1, \dots, n\}$, let $\mathbf{K}_{I,J}$ denote the sub-Gram matrix with respectively row and column indices in I and J. Let $J_p \subset \{1, \dots, n\}$ denote such a subset with cardinality p, and consider the sub-Gram matrix $\mathbf{K}_{J_p,J_p}$ which is of rank $r \le p$. Further assuming $r = p$, the best rank p approximation to $\mathbf{K}_{J_p,J_p}$ is $\mathbf{K}_{J_p,J_p}$ itself. This leads to the final approximation to the Gram Matrix $\mathbf{K}$ (Drineas and Mahoney, 2005; Bach, 2013) by

$$\widetilde{\mathbf{K}} = \mathbf{K}_{I_n,J_p}\, \mathbf{K}^{+}_{J_p,J_p}\, \mathbf{K}_{J_p,I_n},$$

where $I_n = \{1, \dots, n\}$, and $\mathbf{K}^{+}_{J_p,J_p}$ denotes the pseudo-inverse of $\mathbf{K}_{J_p,J_p}$. Further considering the SVD decomposition of $\mathbf{K}_{J_p,J_p} = \mathbf{U}'\Lambda\mathbf{U}$, for an orthonormal matrix $\mathbf{U}$, we can rewrite

$$\widetilde{\mathbf{K}} = \mathbf{Z}'\mathbf{Z}, \qquad \text{with} \quad \mathbf{Z} = \Lambda^{-1/2}\mathbf{U}\,\mathbf{K}_{J_p,I_n} \in \mathcal{M}_{p,n}(\mathbb{R}),$$

where $\mathcal{M}_{p,n}(\mathbb{R})$ is the set of all p by n matrices. Note that the resulting time complexity is $\mathcal{O}(p^2 n)$, which is smaller than the former $\mathcal{O}(n^3)$ as long as $p = o(\sqrt{n})$. This way, columns $\{Z_i\}_{1\le i\le n}$ of $\mathbf{Z}$ act as new p-dimensional observations, and each $\widetilde{\mathbf{K}}_{i,j}$ can be seen as the inner product between two vectors of $\mathbb{R}^p$, that is

$$\widetilde{\mathbf{K}}_{i,j} = Z_i' Z_j. \tag{17}$$

The main interest of this approximation is that, using Eq. (16), computing the cost of a segment of length t has a complexity $\mathcal{O}(t)$ in time unlike the usual $\mathcal{O}(t^2)$ that holds with general kernels.

Interestingly such an approximation to the Gram matrix can be also built from a set of deterministic points in $\mathcal{X}$. This remark has been exploited to compute our low-rank approximation for instance in the simulation experiments as explained in Section 4.5.5.

Note that choosing the set J_p of columns/rows leading to the approximation $\widetilde{\mathbf{K}}$ is of great interest in itself for at least two reasons. First from a computational point of view, the p columns have to be selected following a process that does not require to compute the n possible columns beforehand (which would induce an $O(n^2)$ time complexity otherwise). Second, the quality of $\widetilde{\mathbf{K}}$ to approximate $\mathbf{K}$ crucially depends on the rank of $\widetilde{\mathbf{K}}$ that has to be as close as possible to that of $\mathbf{K}$, which remains unknown for computational reasons. However such questions are out of scope of the present paper, and we refer interested readers to Williams and Seeger (2001), Drineas and Mahoney (2005) and Bach (2013) where this point has been extensively discussed.

3.2.2. Binary segmentation heuristic

Since the low-rank approximation to the Gram matrix detailed in Section 3.2.1 leads to finite dimensional vectors in $\mathbb{R}^p$ (17), the change-point detection problem described in Section 2.3 amounts to recover abrupt changes of the mean of a p-dimensional time-series. Therefore any existing algorithm usually used to solve this problem in the p-dimensional framework can be applied. An exhaustive review of such algorithms is out of the scope of the present paper. However we will mention only a few of them to highlight their drawbacks and motivate our choice. Let us also recall that our purpose is to provide an efficient algorithm allowing: (*i*) to (approximately) solve Eq. (13) for each $1 \le D \le D_{\max}$ and (*ii*) to deal with large sample sizes ($n \ge 10^6$).

The first algorithm is the usual version of constrained dynamic programming (Auger and Lawrence, 1989). Although it has been recently revisited with $p = 1$ by Rigaill (2015), Cleynen et al. (2014b) and Maidstone et al. (2017), it has a $\mathcal{O}(n^2)$ time complexity with $p > 1$, which excludes dealing with large sample sizes. Another version of regularized dynamic programming has been explored by Killick et al. (2012) who designed the PELT procedure. It provides the best segmentation over all segmentations with a penalty of λ per change-point with an $\mathcal{O}(n)$ complexity in time if the number of change-points is linear in n. Importantly, the complexity of the pruning inside PELT depends on the true number of change-points. For only a few change-points, the PELT complexity remains quadratic in time. With PELT, it is not straightforward to efficiently solve Eq. (13) for each $1 \leq D \leq D_{\max}$, which is precisely the goal we pursue. Note however that it would still be possible to recover some of those segmentations by exploring a range of λ values like in CROPS (Haynes et al., 2017).

A second possible algorithm is the so-called *binary segmentation* (Olshen et al., 2004; Yang, 2012; Fryzlewicz, 2014) that is a standard heuristic for approximately solving Eq. (13) for each $1 \leq D \leq D_{\max}$. This iterative algorithm computes the new segmentation $\widetilde{\tau}(D+1)$ with $D+1$ segments from $\widetilde{\tau}(D)$ by splitting one segment of $\widetilde{\tau}(D)$ into two new ones without modifying other segments. More precisely considering the set of change-points $\widetilde{\tau}(D) = \{\tau_1, \ldots, \tau_{D+1}\}$, binary segmentation provides

$$\widetilde{\tau}(D+1) = \underset{\tau \in \mathcal{T}_{D+1} | \tau \cap \widetilde{\tau}(D) = \widetilde{\tau}(D)}{\arg\min} \left\{ \|Y - \hat{\mu}^{\tau}\|^2_{\mathcal{H},n} \right\}.$$

Since only one segment of the previous segmentation is divided into two new segments at each step, the binary segmentation algorithm provides a simple (but only approximate) solution to Eq. (13) for each $1 \leq D \leq D_{\max}$.

We provide some pseudo-code for binary segmentation in Algorithm 5. It uses a sub-routine described by Algorithm 4 to compute the best split of any segment $[\tau, \tau'[$ of the data. To be specific, this BestSplit routine outputs four things: (1) the reduction in cost of splitting the segment $[\tau, \tau'[$, (2) the best change $\hat{t}$ in the segment (3) the resulting left segment and (4) the resulting right segment.

In the binary segmentation algorithm candidate splits are stored and handled using a binary heap data structure (Cormen, 2009) using the reduction in cost as a key. This data structure allows to efficiently insert new splits and extract the best split in $\mathcal{O}(\log(D_{max}))$ at every time step. Without such a structure inserting splits and extracting the best split would typically be in $\mathcal{O}(D_{max})$ and for large D_{max} the binary segmentation heuristic is at best $\mathcal{O}(n^2)$. Note that the RBS procedure (Pierre-Jean et al., 2014), which is involved in our simulation experiments (Section 4.2.3), also uses this heuristic.

Algorithm 4 BestSplit of segment $[\tau, \tau'[$

$\hat{m} = \min_{\tau < t < \tau'} \{C_{\tau,t} + C_{t,\tau'}\}$ and $\hat{t} = \arg\min_{\tau < t < \tau'} \{C_{\tau,t} + C_{t,\tau'}\}$

Output four things (1) $C_{\tau,\tau'} - \hat{m}$, (2) $\hat{t}$, (3) $[\tau, \hat{t}[$ and (4) $[\hat{t}, \tau'[$

Algorithm 5 Binary Segmentation

```
Segs = {[1, n + 1[}
Changes = ∅
CandidateSplit = ∅ [a binary heap]
for D_max iteration do
    for aseg ∈ Segs do
        Insert BestSplit(aseg) in CandidateSplit
    end for
    Extract the best split of CandidateSplit and recover: t̂, [τ, t̂[ and = [t̂, τ'[
    Add t̂ in Changes
    Set Segs to { [τ, t̂[, [t̂, τ'[ }
end for
```

Assuming the best split of any segment is linear in its length the overall time complexity of binary segmentation for recovering approximate solutions to (13) for all $1 \leq D \leq D_{\max}$ is around $\mathcal{O}(\log(D_{\max})n)$ in practice. The worst case time complexity is $\mathcal{O}(D_{\max}n)$. A typical setting where it is achieved is with the linear kernel when $i \mapsto X_i = \exp(i)$ for instance. At the ith iteration of the binary segmentation algorithm, the best split of a segment of length $n - i + 1$ corresponds to one segment of length 1 and another one of length $n - i$.

An important remark is that binary segmentation only achieves this reduced $\mathcal{O}(\log(D_{\max})n)$ time complexity provided that recovering the best split of any segment is linear in its length. This is precisely what has been allowed by the low-rank matrix approximation summarized by Eq. (17). Indeed with the low-rank approximation, computing the best split of any segment is linear in n and p. The resulting time complexity of binary segmentation is thus $\mathcal{O}(p\log(D_{\max})n)$, which reduces to $\mathcal{O}(\log(D_{\max})n)$ as long as p is small compared to n. By contrast without the approximation, recovering the best split is typically quadratic in the length of the segment and binary segmentation would suffer an overall time complexity of order $\mathcal{O}(\log(D_{\max})n^2)$ or $\mathcal{O}(D_{\max}n^2)$.

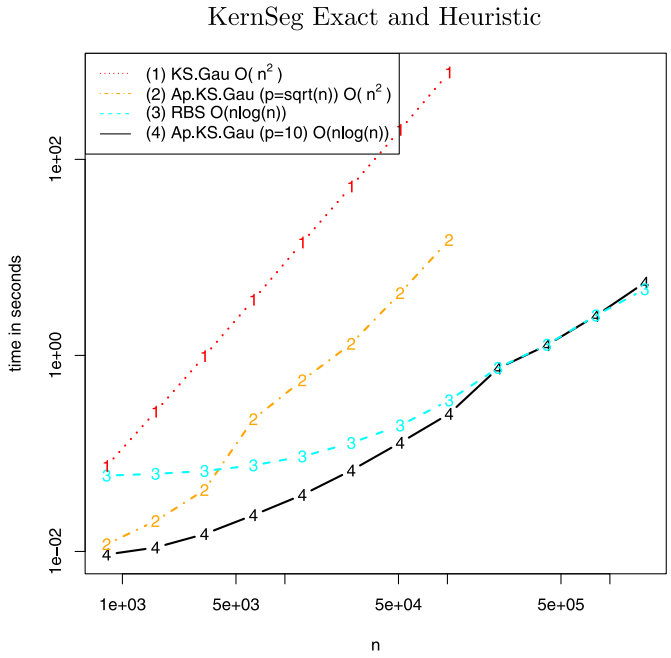


Fig. 2. Runtime as a function of n (length of the signal) for $D_{max} = 100$. Average runtime of exact Algorithm 3 with a Gaussian kernel (KS.Gau, 1-red) , our approximate algorithm with a Gaussian kernel and $p = \sqrt{n}$ (Ap.KS.Gau 2-orange), RBS (3-cyan) and our approximate algorithm with a Gaussian kernel and $p = 10$ (Ap.KS.Gau, 4-black). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.2.3. Implementation and runtimes of the approximate solution

The approximate algorithm we recommend is the combination of the low-rank approximation step detailed in Section 3.2.1 and of the binary segmentation discussed in Section 3.2.2. We provide the pseudo-code of this approximate algorithm, namely Algorithm 6. The resulting time complexity is then $\mathcal{O}(p^2 n + p \log(D_{\max}) n)$, which allows dealing with large sample sizes ($n \geq 10^6$).

Algorithm 6 ApKS: Low rank approximation followed by binary segmentation

1: Compute the partial Gram-matrix $\mathbf{K}_{J_p J_p}$
2: Use SVD to recover the p by n matrix $\mathbf{Z}$
3: Run binary segmentation on $\mathbf{Z}$

From this time complexity it arises that an influential parameter is the number p of columns of the matrix used to build the low-rank approximation. In particular this low-rank approximation remains computationally attractive as long as $p = o(\sqrt{n})$. Fig. 2 illustrates the actual time complexity of this fast algorithm (implemented in C) with respect to n for various values of p: (i) a constant value of p and (ii) $p = \sqrt{n}$. To ease the comparison, we also plotted the runtime of the exact algorithm (Algorithm 3) detailed in Section 3.1.2 and RBS that uses binary segmentation (see Section 4.2.3).

Our fast approximating algorithm (*ApKS*) recovers a quadratic complexity if $p = \sqrt{n}$. However its overhead is much smaller than that of the exact algorithm, which makes it more applicable than the latter with large signals in practice. Note also that Fig. 2 illustrates that *ApKS* returns the solution in a matter of seconds with a sample size of $n = 10^5$, which is much faster than *Kernseg* (based on dynamic programming) that requires a few minutes. The RBS implementation involves preliminary calculations which make it slower than *ApKS* with $n \leq 2 \cdot 10^3$. However for larger values ($n \geq 10^4$) RBS is as fast as *ApKS* with $p = 10$.

4. Segmentation assessment

From a statistical point of view *Kernseg* provides the same performance as that of Arlot et al. (2012). However it greatly improves on the latter in terms of computational complexity as proved in Section 3.1. Their simulation experiments (Arlot et al., 2012) mainly focus on detecting change-points in the distribution of $\mathbb{R}$-valued data as well as of more structured objects such as histograms. Here we rather investigate the performance of the kernel change-point procedure on specific two-dimensional biological data: the DNA copy number and the BAF profiles (see Section 4.1.1). More precisely our experiments highlight two main assets of applying reproducing kernels to these biological data: (i) reasonable kernels avoid the need for modeling the type of change-points we are interested in and improve upon state-of-the-art approaches in this biological

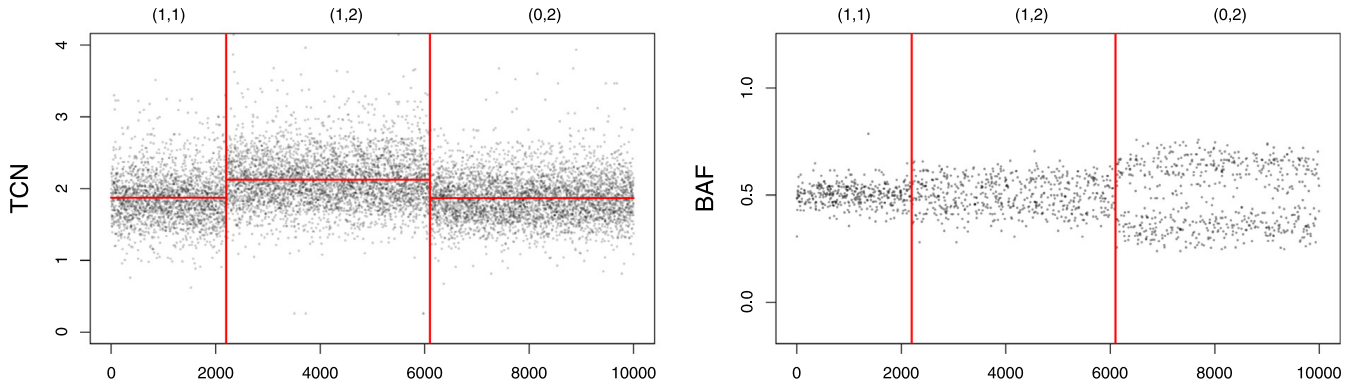


Fig. 3. SNP array data. Total copy numbers (TCN), allelic ratios (BAF) along 10,000 genomic loci. Red vertical lines represent change-points, and red horizontal lines represent estimated mean signal levels between two change-points. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

context, and (*ii*) the high flexibility of kernels facilitates data fusion, that is allows to combine different data-types and get more power to detect true change-points.

In the following we first briefly introduce the type of data we are looking at, and describe our simulation experiments obtained by resampling from a set of real annotated DNA profiles. Second, we provide details about the change-point procedures involved in our comparison. We also define the criteria used to assess the performance of the estimated segmentations. Finally, we report and discuss the results of these experiments.

4.1. Data description

4.1.1. DNA copy number data

DNA copy number alterations are a hallmark of cancer cells (Hanahan and Weinberg, 2011). The accurate detection and interpretation of such changes are two important steps toward improved diagnosis and treatment. Normal cells have two copies of DNA, inherited from each biological parent of the individual. In tumor cells, parts of a chromosome of various sizes (from kilobases to a chromosome arm) can be deleted, or copied several times. As a result, DNA copy numbers in tumor cells are piecewise constant along the genome. Copy numbers can be measured using microarray or sequencing experiments. Fig. 3 displays an example of copy number profiles that can be obtained from SNP-array data (Neuvial et al., 2011).

The left panel (denoted by TCN) represents estimates of the total copy number (TCN). The right panel (denoted by BAF) represents estimates of allele B fractions (BAF) using only homozygous position. We refer to Neuvial et al. (2011) for an explanation of how these estimates are obtained. In the normal region [0-2200], TCN is centered around two copies and BAF has three modes at 0, 1/2 and 1.

On top of Fig. 3, numbers (a, b) represent each parental copy number in the corresponding segment. For instance $(0, 2)$ means that the total number of copies in the segment is 2. But a copy from one of the two parents is missing while the other copy has been duplicated. Importantly any change in only one of the parental copy numbers is reflected in both TCN and BAF. Therefore it makes sense to jointly analyze both dimensions to ease the identification of change-points.

Importantly in the following, allelic ratios (BAF) are always symmetrized (or folded) – that is we consider $|BAF - 0.5|$ – to facilitate the segmentation task. This is common practice in the field (Staaf et al., 2008).

4.1.2. Generated data

Realistic DNA profiles with known truth (similar to that of Fig. 3) have been generated using the `acnr` package (Pierre-Jean et al., 2014). The constituted benchmark consists of profiles with 5,000 positions of heterozygous SNPs and exactly $K = 10$ change-points. As in Pierre-Jean et al. (2014) we only consider four biological states for the segments. The `acnr` package allows to vary the difficulty level by adding normal cell contamination, thus degrading tumor percentage. Three levels of difficulty have been considered by varying tumor percentage: 100% (easy case), 70%, and 50% (difficult case). Fig. 4 displays three examples of simulated profiles (one for each tumor purity level).

For each level, $N = 50$ profiles (with the same segment states and change-points) are generated making a total of 150 simulated profiles both for BAF and TCN.

4.2. Competing procedures

4.2.1. Kernseg and ApKS

The implemented exact algorithm *Kernseg* (corresponding to Algorithm 3) and its fast approximation *ApKS* (based on binary segmentation) are applicable with any kernel (Gaussian, exponential, polynomial, …). All the experimental results exposed in what follows have been obtained from the R-package *KernSeg* (Marot et al., 2018).

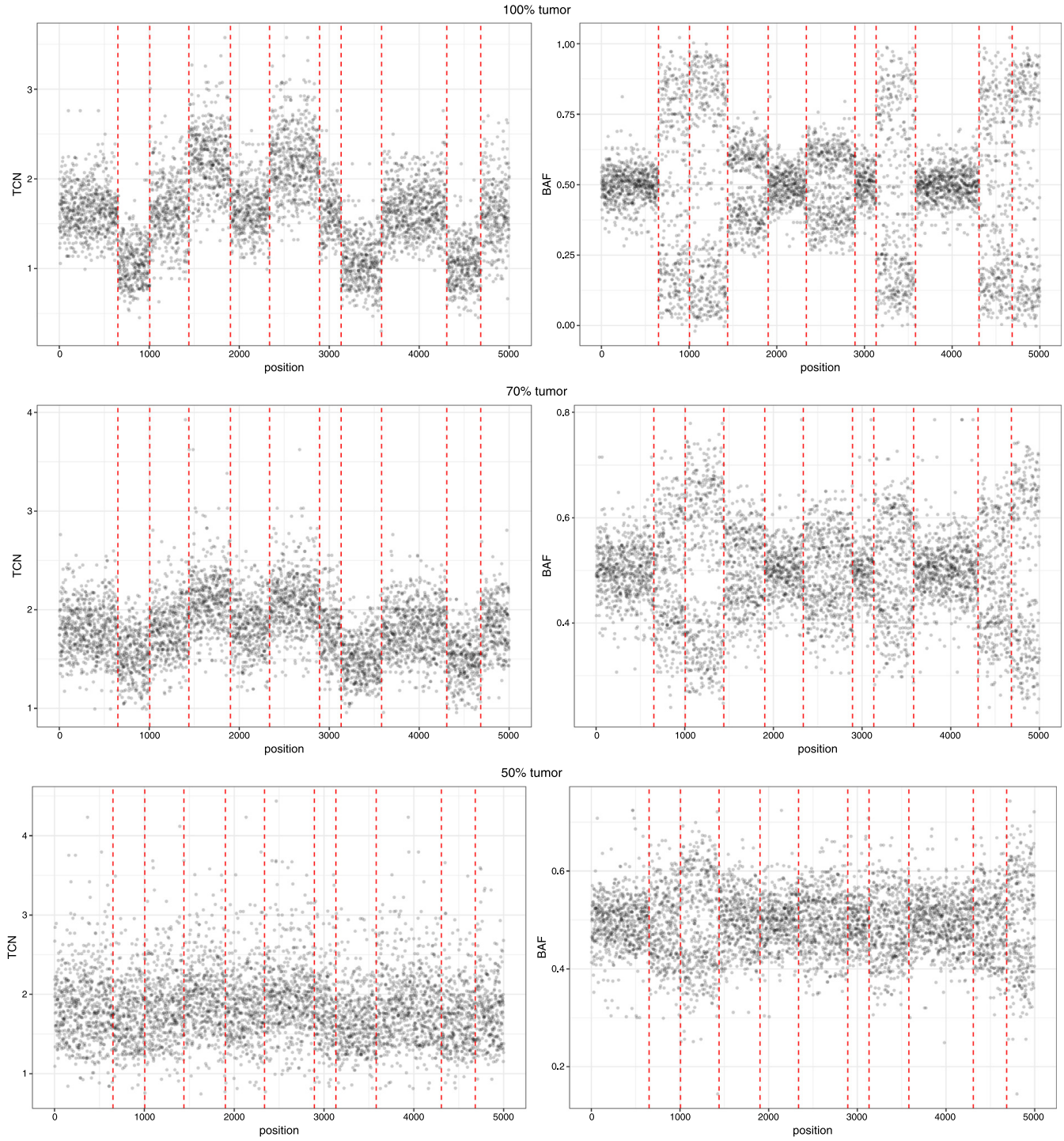


Fig. 4. Benchmark 1: Profiles simulated with the `acnr` package. Each line corresponds to a tumor percentage (100%, 70% and 50%). The first column corresponds to copy number (TCN) and the second to the allele B fraction (BAF).

In our simulation experiments we consider three kernels.

- The first one is the so-called linear kernel defined by $k(x, y) = \langle x, y \rangle_1$, where $x, y \in \mathbb{R}$. It is used as a baseline since *Kernseg* with this kernel reduces to the procedure of Lebarbier (2005). The corresponding procedure is denoted by KS.Lin.
- The second one is the Gaussian kernel defined for every $x, y \in \mathbb{R}$ by

$$k_\delta(x, y) = \exp\left[\frac{-|x - y|^2}{\delta}\right], \qquad \forall \delta > 0.$$

Since it belongs to the class of characteristic kernels (Sriperumbudur et al., 2010), it is a natural choice to detect any abrupt changes arising in the full distribution (Arlot et al., 2012). We call this procedure KS.Gau.

- The third one is the kernel associated with the energy-based distance introduced in Eq. (12) with $\alpha = 1$ and $x_0 = 0$. This particular choice is the prescribed value in the ECP package (James and Matteson, 2013). We call this procedure KS.ECP.

These three kernels allow to (*i*) illustrate the interest of characteristic kernels compared to non characteristic ones, and (*ii*) assess the performances of change-point detection with kernels (*Kernseg*) compared to other approaches (ECP, RBS). However other characteristic kernels such as the Laplace or exponential ones (see Section 2.2) could have been considered as well.

For all kernels we considered $D_{max} = 100$. Note also that for all approaches and for both TCN and BAF profiles we first scaled the data using a difference based estimator of the variance. To be specific we get an estimator of the variances by dividing by $\sqrt{2}$ the median absolute deviation of disjoint successive differences. This is common practice in the change-point literature (see for example Fryzlewicz, 2014). Such estimators are less sensitive to any shift in the mean than the classical ones. For the Gaussian kernel we then used $\delta = 1$.

As mentioned earlier, one main asset of kernels is that they allow to easily perform data fusion, which consists of combining several data profiles to increase the power of detecting small changes arising at the same location in several of them. Here the joint segmentation of the two-dimensional signal (TCN, BAF) is carried out by defining a new kernel as the sum of two coordinate-wise kernels (Aronszajn, 1950), that is

$$k(x_1, x_2) = k(c_1, c_2) + k(b_1, b_2) \tag{18}$$

with $x_1 = (c_1, b_1)$ and $x_2 = (c_2, b_2)$ where the first coordinates of x_1 and x_2 refer to TCN and the second ones refer to BAF.

Remark 4. Let us point out that many alternative ways exist to build such a "joint kernel", using the standard machinery of reproducing kernels exposed in Aronszajn (1950) and Gartner (2008).

For instance replacing the sum in Eq. (18) by a product of kernels is possible. With the Gaussian kernel, this amounts to consider one Gaussian kernel applied to a mixture of squared norms where each coordinate receives a different weight depending on its influence. Another promising direction is to exploit some available side information about the importance of each coordinate in detecting change-points. This can be done by considering a convex sum of kernels where the weights reflect this *a priori* knowledge.

Finally let us mention that designing the optimal kernel for a learning task is a widely open problem in the literature even if some attempts exist (see Section 7.2 in Arlot et al. (2012) for a thorough discussion, and Gretton et al. (2012) for a first partial answer with two-sample tests).

4.2.2. ECP

The ECP procedure (Matteson and James, 2014) (earlier discussed in Section 2.6) has been also introduced in our comparison since it allows us to detect changes in the distribution of multivariate observations.

We used the implementation provided by the authors in the R package (James and Matteson, 2013) with the default parameters $\alpha = 1$ and $\ell = 30$ (minimum length of any segment). Let us remark that, unlike our kernel-based procedures relying on efficiently minimizing a prescribed penalized criterion, ECP chooses the number of segments by iteratively testing each new candidate change-point by means of a permutation test, which makes it highly time-consuming on large profiles (around 15 min per profiles for $n = 5000$ compared to 5 s for KS.Gau).

4.2.3. Recursive Binary Segmentation (RBS)

In the recent paper by Pierre-Jean et al. (2014), it has been shown that for a known number of change-points the Recursive Binary Segmentation (RBS) (Gey and Lebarbier, 2008) is a state-of-the-art change-point procedure for analyzing (TCN, BAF) profiles. RBS is a two-step procedure. In a first step it uses the binary segmentation heuristic (described in Section 3.2.2) on the (TCN) or (TCN,BAF) profile. In a second step it uses dynamic programming on the set of changes identified by the binary segmentation heuristic. We refer interested readers to Pierre-Jean et al. (2014) for a discussion as to why RBS can outperform a pure dynamic programming strategy despite the fact that it provides only an approximation to the solution of the targeted optimization problem.

Since the present biological context is the same as that of Pierre-Jean et al. (2014), we therefore decided to carry out the comparison between our kernel-based procedures and RBS.

From a computational perspective RBS relies on the binary segmentation algorithm described in Algorithm 5. The final segmentation output by RBS is then an approximate solution to the optimization problem (in the same way as *ApKS*), while being efficiently computed as illustrated by Fig. 2.

4.3. Performances assessment

The quality of the resulting segmentations is quantified in two ways. First we infer the ability of the procedure to provide a reliable estimate of the regression function by computing the quadratic risk of the estimator based on the TCN profile (Section 4.3.1). Second, we also assess the quality of the estimated segmentations by measuring the discrepancy between the true and estimated change-points using the Frobenius distance (Section 4.3.2).

4.3.1. Risk of a segmentation

From a practical point of view, there is no hope to recover true change-points in regions where the signal-to-noise ratio is too low without including false positives, which we would like to avoid. In such non-asymptotic settings, the quality of the estimated segmentation τ can be measured by the risk $R(\hat{f}^\tau)$ which measures the gap between the regression function $f = (f_1, \ldots, f_n) \in \mathbb{R}^n$ and its piecewise-constant estimator based on τ, that is $\hat{f}^\tau = \left(\hat{f}_1^\tau, \ldots, \hat{f}_n^\tau\right) \in \mathbb{R}^n$. This risk is defined by

$$R(\hat{f}^\tau) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\left(f_i - \hat{f}_i^\tau\right)^2\right].$$

In the following simulation results, the risks of all segmentations are always computed with respect to the regression function of the corresponding TCN profile.

4.3.2. Frobenius distance

We also quantify the gap between a segmentation τ and the true segmentation τ^* by using the Frobenius distance (Lajugie et al., 2014) between matrices as follows. First, for any segmentation $\tau = (\tau_1, \tau_2, \ldots, \tau_D)$, let us introduce a matrix $M^\tau = \left\{M_{i,j}^\tau\right\}_{1\le i,j\le n}$ such that

$$M_{i,j}^\tau = \sum_{k=1}^{D} \frac{\mathbb{1}_{(\tau_k \le i,j < \tau_{k+1})}}{\tau_{k+1} - \tau_k}, \qquad \text{(with } \tau_1 = 1 \text{ and } \tau_{D+1} = n+1 \text{ by convention)}$$

where $\mathbb{1}_{(\tau_k \le i,j < \tau_{k+1})} = 1$, if $i, j \in [\tau_k, \tau_{k+1}[\cap \mathbb{N}$, and 0 otherwise. Note that $M_{i,j}^\tau \neq 0$ if and only if i, j are in the same segment of τ, which leads to a block-diagonal matrix with D blocks (whose squared Frobenius norm is equal to D). The idea behind the value in each block of this matrix is to define a one-to-one mapping between the set of segmentations in D segments and matrices whose squared Frobenius norm is D.

Let us now consider the matrix $M^{\tau^\star}$ defined from the true segmentation τ^* in the same way. Then, the Frobenius distance between segmentations τ and τ^* is given, through the distance between matrices M^τ and $M^{\tau^\star}$, by

$$d_F\left(\tau, \tau^\star\right) = \left\|M^\tau - M^{\tau^\star}\right\|_F = \sqrt{\sum_{i,j=1}^{n}\left(M_{i,j}^\tau - M_{i,j}^{\tau^\star}\right)^2}.$$

4.4. Testing procedure for significance assessment

A paired Student test has been used to assess the significance of the difference between any pair of approaches on a given dataset and for a given performance measure. Each test has been performed from 100 repetitions.

4.5. Results

In our experiments, we successively considered two types of signals: (*i*) the total copy number profiles (TCN) and (*ii*) the joint profiles in $\mathbb{R}^2$ made of (TCN,BAF).

4.5.1. Comparison with KS.Lin and ECP for a high tumor percentage (easy case)

First we compare all approaches in the simple case where the tumor percentage is equal to 100%. The performances, using only the TCN or the (TCN,BAF) profiles, are reported in Fig. 5 and measured in terms of accuracy (top) and risk (bottom).

In all these experiments RBS clearly performs badly. We believe this is mostly due to the poor estimation of the number of segments made by RBS. Indeed the performances of RBS are closer to the ones of other approaches when considering the true number of segments (results not shown here).

We then compare KS.Lin to KS.Gau, KS.ECP, and ECP. With TCN data, KS.Lin has a small advantage over KS.Gau (with an average accuracy difference of 0.03 and a p-value of 0.012) and ECP (with an average accuracy difference of 0.1 and p-values of 0.007). This is also true when considering the risk. KS.ECP has a slightly better empirical average accuracy than KS.Lin but this difference is not significant. Let us also mention that none of the differences are found significant with the (TCN, BAF) profiles. It is our opinion that in this simple scenario all true change-points arise mostly in the mean of the distribution. Thus it is remarkable that the performances of approaches also looking for changes in the whole distribution (like ECP, KS.Gau, KS.ECP) are (almost) on par with those specifically looking for change-points in the mean (like KS.Lin and RBS for the true number of change-points $D^\star$).

We also compared ECP to KS.Gau and KS.ECP. We found no differences except between ECP and KS.ECP for TCN profiles. In that case KS.ECP has significantly better accuracy and risk than ECP. But this difference remains small as can be seen on Fig. 5.

Note that for all approaches, performances on (TCN,BAF) profiles are slightly better than those with TCN profiles (p-values smaller than 10^{-4}).

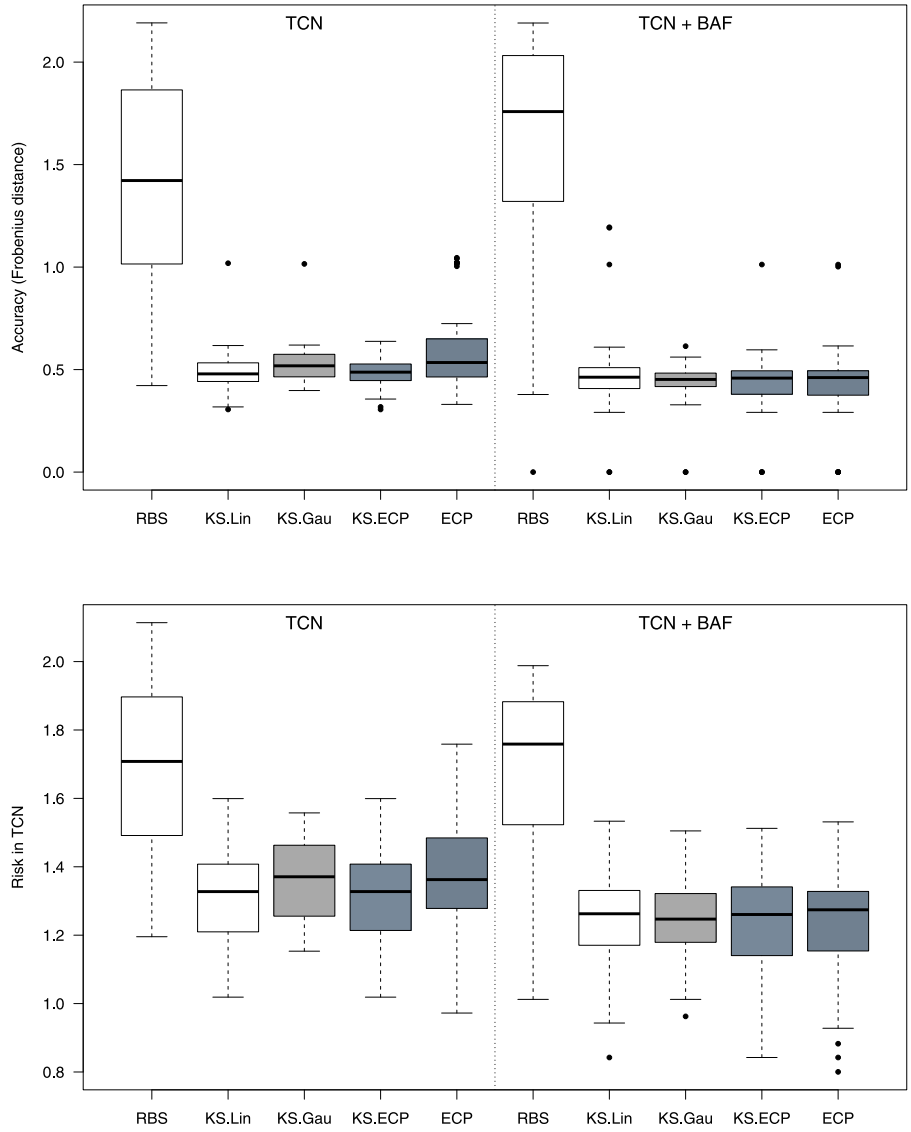


Fig. 5. Accuracy (top) and Risk (bottom) on TCN and (TCN,BAF) profiles with a tumor percentage of 100%. Boxplots of RBS, ECP, KS.Lin, KS.Gau and KS.ECP for their selected number of change-points ($\hat{D}$) are shown.

4.5.2. Constraint on the segment sizes for a low tumor percentage (difficult case)

We then turn to the more difficult case where the tumor percentage is equal to 50%. In this scenario excluding segments with less than 30 points (as is done by default in ECP) is beneficial. Fig. 6 illustrates this strong improvement when adding this constraint to KS.Lin, KS.Gau and KS.ECP and when considering the true number of change-points $D^\star = 10$ (p-values of respectively (8.10^{-9}, 9.10^{-3} and 10^{-4}). More generally it is our experience that such a constraint can greatly improve performances when the signal-to-noise ratio is low. For this reason, in the remainder of our experiments and for a tumor percentage of 50%, we will report results including the constraint on the segment sizes ($\ell = 30$). Let us also mention that for higher tumor percentages adding the constraint does not change the segmentation in $D^\star$ segments recovered by KS.Lin, KS.Gau and KS.ECP.

4.5.3. Comparison with KS.Lin and ECP for a low tumor percentage (difficult case)

We compared KS.Lin to KS.Gau, KS.ECP, and ECP for a tumor percentage of 50%. The accuracy of all these approaches is reported in Fig. 7. The minimum length of any segment is fixed at $\ell = 30$ for all approaches (except RBS as it is not possible) and the number of segments is estimated.

In all these experiments RBS performs badly. We believe this is because it poorly selects the number of segments and also because it does not include a constraint on segment sizes.

We compared KS.Lin to KS.Gau, KS.ECP, and ECP. For both TCN and (TCN,BAF) profiles KS.Lin performs worse than KS.Gau, KS.ECP, and ECP in terms of accuracy and risk (all p-values are smaller than 2.10^{-4}). In this more difficult scenario, changes

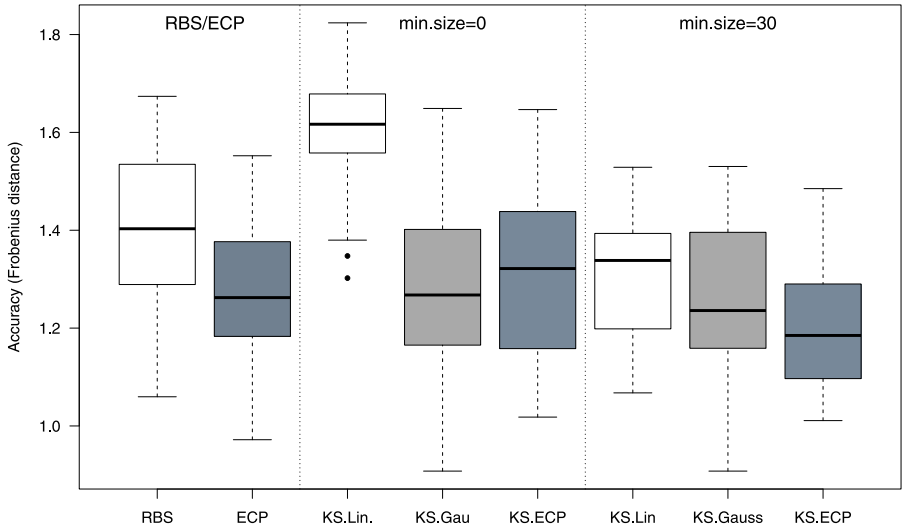


Fig. 6. Performances of KS.Lin, KS.Gau and KS.ECP for the true number of changepoints $D^\star = 10$ with or without a constraint on the minimal size of segments (left: $\ell \geq 1$, right: $\ell \geq 30$). The results for RBS and ECP for $D^\star = 10$ are also reported. RBS does not include a constraint while ECP has a default minimal size of 30.

do not arise only in the mean of the distribution, which gives an advantage to approaches looking for changes in the whole distribution and not only in the mean as KS.Lin does.

We then compare ECP to KS.Gau and KS.ECP. First, KS.Gau seems to have a slightly better accuracy and risk than ECP for both TCN and (TCN,BAF) profiles. Two of these differences are found significant with a cut-off of 5% and none with a cut-off of 1%. This leads us to conclude that ECP and KS.Gau have similar performances in the present experiments. Second, KS.ECP has a slightly better accuracy and risk than ECP in TCN for both TCN and (TCN,BAF) profiles. All of these differences are found significant (for the accuracy in (TCN,BAF) a p-value of 0.0076, in TCN a p-value of 0.015, for the risk in (TCN,BAF) a p-value of 0.0081 and in TCN a p-value of 0.00016). Although significant these differences remain small (about three times smaller than the differences between KS.Lin and ECP).

Finally it should be noted that KS.ECP is faster than ECP for a profile of $n = 5000$ (5 s against 15 min). All of this leads to conclude that, in our simulation experiments, KS.Gau and KS.ECP are the best change-point detection procedures among the considered ones since they perform as well as ECP while being by far less memory and time consuming.

4.5.4. Estimation of the number of segments including the minimum length constraint

Let us now assess the behavior of the model selection procedure derived in Section 2.5 by taking into account the new constraint on the minimal length of the candidate segments.

From Fig. 8 it can be seen that for all kernels the performances of the *Kernseg* procedure at the estimated number of segments $\widehat{D}$ are worse than those at $D^\star$. This difference remains very small. This empirically validates the use of our modified penalty taking into account a constraint on the size of the segments. We recall that adding this constraint is important in low-signal-to-noise settings, which are common in practice.

4.5.5. Quality of the approximation

The purpose of the present section is to illustrate the behavior of *ApKS* (in terms of statistical precision) as an alternative to *Kernseg* (which is more time consuming). Since we do not provide any theoretical guarantee on the model selection performances of *ApKS*, we only show its results for several values of $p \in \{4, 10, 40, 80, 160\}$ at the true number of segments $D^\star$. For each value of p and each of the TCN and BAF profiles, we compute the approximation by: (i) evaluating the smallest and largest observed value (respectively denoted by m and M), (ii) using an equally spaced grid of p deterministic values between m and M and (iii) use those p values to perform the approximation of the Gram matrix.

From Fig. 9 it clearly appears that the number of points used to build the low-rank approximation to the Gram matrix is an influential parameter that has to be carefully fixed. However as long as p is chosen large enough, the approximation seems to provide very similar results. This suggests that one should find a trade-off between the statistical performances and the computation cost. Indeed from a statistical point of view increasing p is beneficial (or at least not detrimental). In contrast from a computational point of view increasing p is detrimental and increases the complexity in time ($O(p^2 n)$).

Let us finally emphasize that for large enough p the performances of *ApKS* are very close to those of KS.Gau. Given the low time complexity of *ApKS* compared to KS.Gau we argue that for large profiles ($n \gg 10^5$) *ApKS* could be an interesting alternative to KS.Gau.

Nevertheless several questions related to the use of *ApKS* remain open. For instance, the optimal way to build the low-rank approximation to the Gram matrix is a challenging question which can be embedded in the more general problem of choosing the optimal kernel. Designing a theoretically grounded penalized criterion to perform model selection with *ApKS* is also a crucial problem which remains to be addressed.

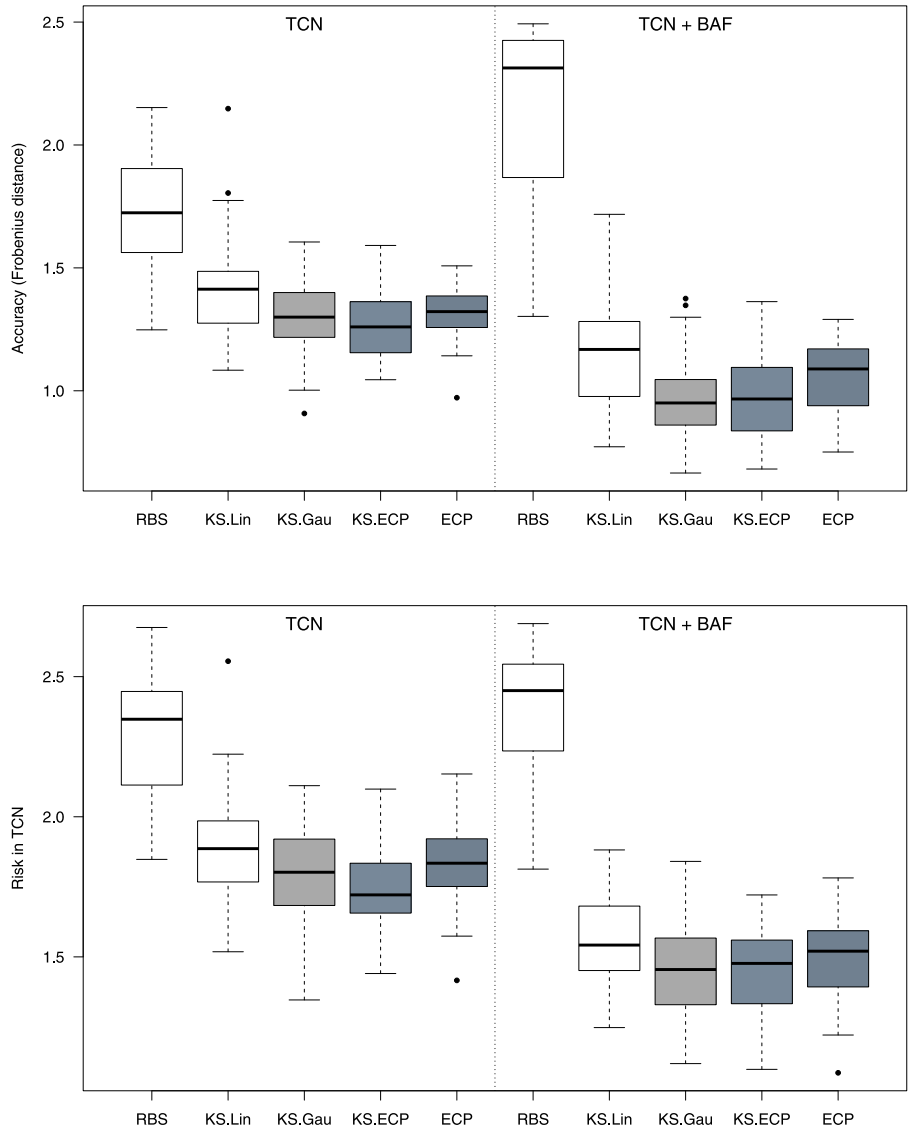


Fig. 7. Accuracy (top) and Risk (bottom) on TCN and (TCN,BAF) profiles with a tumor percentage of 50%. Boxplots of RBS, ECP, KS.Lin, KS.Gau and KS.ECP for their selected number of changepoints ($\hat{D}$) are shown.

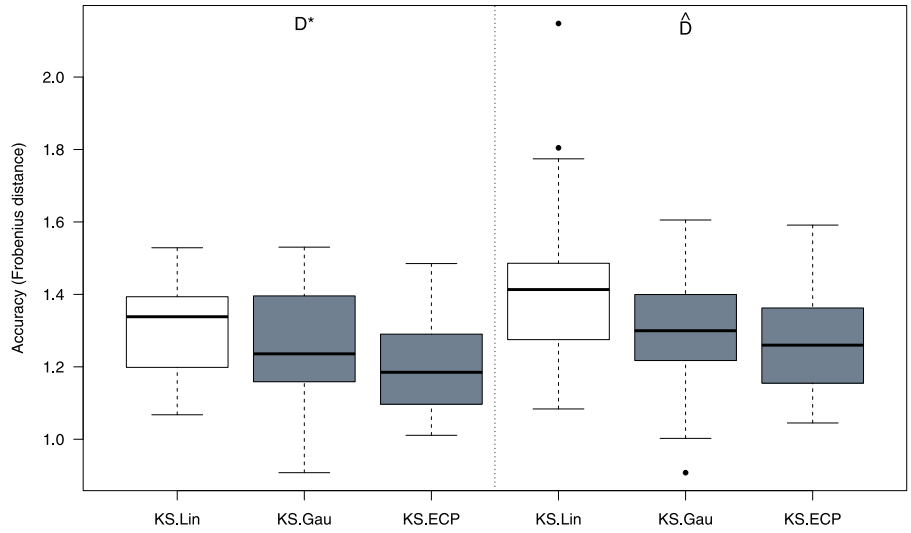


Fig. 8. Accuracy of KS.Lin, KS.Gau and KS.ECP on (TCN,BAF) for a tumor percentage of 50% for the true number of segments $D*$ (left) and the estimated number of segments $\hat{D}$ (right).

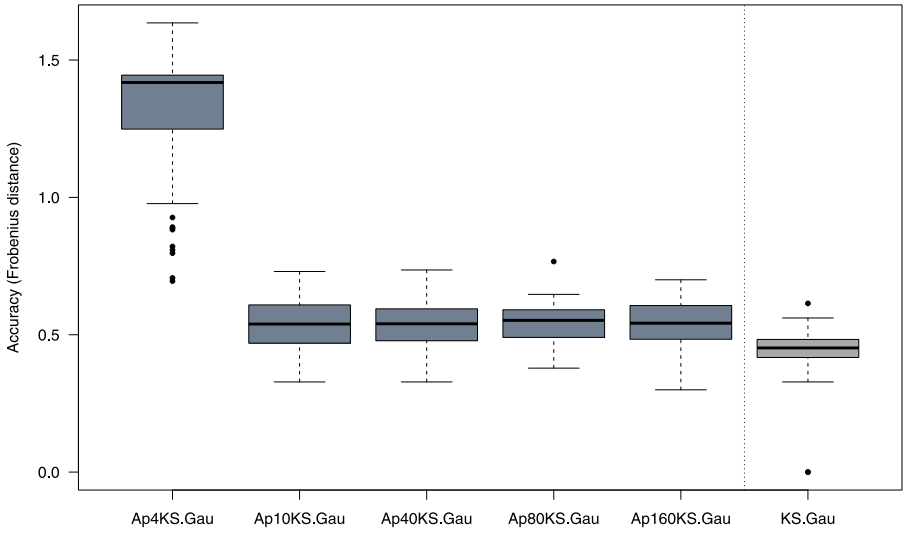


Fig. 9. Accuracy of *ApKS* with the Gaussian Kernel and for various p and of KS.Gau on (TCN,BAF) for a tumor percentage of 50% for $D*$.

5. Conclusion

Existing nonparametric change-point detection procedures such as that of Arlot et al. (2012) exhibit promising statistical performances. Yet their high computational costs (time and memory) are severe limitations that often make it difficult to use for practitioners. Therefore an important task is to develop computationally efficient algorithms (leading to exact or approximate solutions) reducing the time and memory costs of these statistically effective procedures.

In this paper we focus on the multiple change-points detection framework with reproducing kernels. We have detailed a versatile (*i.e.* applicable to any kernel) exact algorithm (*Kernseg*) which is quadratic in time and linear in space. We also provided a versatile approximating algorithm (*ApKS*) which is linear both in time and space and allows to deal with very large signals ($n \geq 10^6$) on a standard laptop. The computational efficiency in time and space of these two new algorithms has been illustrated on empirical simulation experiments showing that the new algorithms is more efficient than its direct competitor ECP. The statistical accuracy of our kernel-based procedures has been empirically assessed in the setting of DNA copy numbers and allele B fraction profiles. In particular, results illustrate that characteristic kernels (enabling the detection of changes in any moment of the distribution) can lead to better performances than procedures dedicated to detecting changes arising only in the mean.

Acknowledgments

This work has been funded by the French Agence Nationale de la Recherche (ANR) under reference ANR-11-BS01-0010, by the CNRS under the PEPS BeFast, by the CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015–2020, and by Chaire d'excellence 2011–2015 Inria/Lille 2.

We thank Pierre Neuvial and Toby Hocking for fruitful discussions and proof reading the paper.

References

Arlot, S., Celisse, A., 2011. Segmentation of the mean of heteroscedastic data via cross-validation. Stat. Comput. 21 (4), 613–632.
Arlot, S., Celisse, A., Harchaoui, Z., 2012. A kernel multiple change-point algorithm via model selection. ArXiv preprint arXiv:1202.3878.
Aronszajn, N., 1950. Theory of reproducing kernels. Trans. Amer. Math. Soc. 68 (3), 337–404.
Auger, I.E., Lawrence, C.E., 1989. Algorithms for the optimal identification of segment neighborhoods. Bull. Math. Biol. 51 (1), 39–54.
Bach, F., 2013. Sharp analysis of low-rank kernel matrix approximations. In: Proc. COLT, 2013. pp. 185–209.
Bellman, R., 1961. On the approximation of curves by line segments using dynamic programming. Commun. ACM 4 (6), 284.
Berg, C., Christensen, J.P.R., Ressel, P., 1984. Harmonic Analysis on Semigroups. Springer, New-York.
Berlinet, A., Thomas-Agnan, C., 2004. Reproducing Kernel Hilbert Spaces in Probability and Statistics. Kluwer Academic Publishers, Boston, MA, p. xxii+355. With a preface by Persi Diaconis. http://dx.doi.org/10.1007/978-1-4419-9096-9.
Birgé, L., Massart, P., 2007. Minimal penalties for Gaussian model selection. Probab. Theory Related Fields 138 (1–2), 33–73.
Brodsky, E., Darkhovsky, B.S., 2013. Nonparametric Methods in Change Point Problems. Springer Science & Business Media.
Christmann, A., Steinwart, I., 2010. Universal kernels on non-standard input spaces. In: Advances in Neural Information Processing Systems. pp. 406–414.
Cleynen, A., Dudoit, S., Robin, S., 2014a. Comparing segmentation methods for genome annotation based on RNA-seq data. J. Agric. Biol. Environ. Stat. 19 (1), 101–118.
Cleynen, A., Koskas, M., Lebarbier, E., Rigaill, G., Robin, S., 2014b. Segmentor3IsBack: an R package for the fast and exact segmentation of Seq-data. Algorithms Mol. Biol. 9, 6.
Cleynen, A., Lebarbier, E., 2014. Segmentation of the Poisson and negative binomial rate models: a penalized estimator. ESAIM Probab. Stat. 18, 750–769.

Cormen, T.H., 2009. Introduction to Algorithms. MIT press.
Dieuleveut, A., Bach, F., 2016. Nonparametric stochastic approximation with large step-sizes. Ann. Statist. 44 (4), 1363–1399.
Drineas, P., Mahoney, M.W., 2005. On the Nyström method for approximating a gram matrix for improved kernel-based learning. J. Mach. Learn. Res. 6, 2153–2175.
Fine, S., Scheinberg, K., Cristianini, N., Shawe-Taylor, J., Williamson, B., 2001. Efficient SVM training using low-rank kernel representations. J. Mach. Learn. Res. 2, 243–264.
Fryzlewicz, P., 2014. Wild binary segmentation for multiple change-point detection. Ann. Statist. 42 (6), 2243–2281.
Garreau, D., Arlot, S., 2016. Consistent change-point detection with kernels. ArXiv preprint arXiv:1612.04740.
Gartner, T., 2008. Kernels for Structured Data. World Scientific.
Gey, S., Lebarbier, E., 2008. Using CART to detect multiple change points in the mean for large sample. Tech. Rep. HAL. https://hal.inria.fr/file/index/docid/327146/filename/article_CART.pdf.
Gretton, A., Sejdinovic, D., Strathmann, H., Balakrishnan, S., Pontil, M., Fukumizu, K., Sriperumbudur, B.K., 2012. Optimal kernel choice for large-scale two-sample tests. In: Advances in Neural Information Processing Systems. pp. 1205–1213.
Hanahan, D., Weinberg, R.A., 2011. Hallmarks of cancer: the next generation. Cell 144 (5), 646–674.
Harchaoui, Z., Cappé, O., 2007. Retrospective mutiple change-point estimation with kernels. In: Statistical Signal Processing, 2007. SSP'07. IEEE/SP 14th Workshop on. IEEE, pp. 768–772.
Hautaniemi, A.R., Kauraniemi, S., Yli-Harja, P., Astola, O., Wolf, J., Kallioniemi, M., 2003. A CGH-plotter: MATLAB toolbox for CGH-data analysis. Bioinformatics 13 (1714–1715).
Haynes, K., Eckley, I.A., Fearnhead, P., 2017. Computationally efficient changepoint detection for a range of penalties. J. Comput. Graph. Statist. 26 (1), 134–143.
Hocking, T., Schleiermacher, G., Janoueix-Lerosey, I., Boeva, V., Cappo, J., Delattre, O., Bach, F., Vert, J.P., 2013. Learning smoothing models of copy number profiles using breakpoint annotations. BMC Bioinformatics 14 (1), 164.
James, N.A., Matteson, D.S., 2013. ecp: An R package for nonparametric multiple change point analysis of multivariate data. ArXiv preprint arXiv:1309.3295. http://cran.r-project.org/web/packages/ecp/index.html.
Jong, K., Marchiori, E., van der Vaart, A., Ylstra, B., Weiss, M., Meijer, G., 2003. Chromosomal breakpoint detection in human cancer. In: Applications of Evolutionary Computing. In: EvoWorkshops 2003: Proceedings, vol. 2611, Springer-Verlag Heidelberg, pp. 54–65.
Killick, R., Fearnhead, P., Eckley, I., 2012. Optimal detection of changepoints with a linear computational cost. J. Amer. Statist. Assoc. 107 (500), 1590–1598.
Lai, Y., 2012. Change-point analysis of paired allele-specific copy number variation data. J. Comput. Biol. 19 (6), 679–693.
Lajugie, R., Bach, F., Arlot, S., 2014. Large-margin metric learning for constrained partitioning problems. In: International Conference on Machine Learning. pp. 297–305.
Lebarbier, E., 2005. Detecting multiple change-points in the mean of Gaussian process by model selection. Signal Process. 85 (4), 717–736. http://dx.doi.org/10.1016/j.sigpro.2004.11.012.
Ledoux, M., Talagrand, M., 1991. Isoperimetry and processes. In: Probability in Banach Spaces. In: Ergebnisse der Mathematik und ihrer Grenzgebiete (3) [Results in Mathematics and Related Areas (3)], vol. 23, Springer-Verlag, Berlin, p. xii+480.
Maidstone, R., Hocking, T., Rigaill, G., Fearnhead, P., 2017. On optimal multiple changepoint algorithms for large data. Stat. Comput. 27 (2), 519–533.
Marot, G., Celisse, A., Rigaill, G., 2018. R-package *KernSeg*. https://r-forge.r-project.org/R/?group_id=2300.
Matteson, D.S., James, N.A., 2014. A nonparametric approach for multiple change point analysis of multivariate data. J. Amer. Statist. Assoc. 109 (505), 334–345.
Neuvial, P., Bengtsson, H., Speed, T.P., 2011. Statistical analysis of Single Nucleotide Polymorphism microarrays in cancer studies. In: Handbook of Statistical Bioinformatics, first ed.. In: Springer Handbooks of Computational Statistics, Springer, pp. 225–255.
Olshen, A.B., Venkatraman, E.S., Lucito, R., Wigler, M., 2004. Circular binary segmentation for the analysis of array-based DNA copy number data. Biostatistics 5 (4), 557–572.
Picard, F., Robin, S., Lavielle, M., Vaisse, C., Daudin, J.J., 2005. A statistical approach for array-CGH data analysis. BMC Bioinform. 6, 27. http://www.ncbi.nlm.nih.gov/pubmed/15705208.
Pierre-Jean, M., Rigaill, G., Neuvial, P., 2014. Performance evaluation of DNA copy number segmentation methods. Brief. Bioinform. http://bib.oxfordjournals.org/content/early/2014/09/08/bib.bbu026.abstract.
Rigaill, G., 2015. A pruned dynamic programming algorithm to recover the best segmentations with 1 to K_max change-points.. J. Soc. Fr. Stat. 156 (4), 180–205.
Rigaill, G., Lebarbier, É., Robin, S., 2012. Exact posterior distributions and model selection criteria for multiple change-point detection problems. Stat. Comput. 22 (4), 917–929.
Sejdinovic, D., Sriperumbudur, B., Gretton, A., Fukumizu, K., 2013. Equivalence of distance-based and RKHS-based statistics in hypothesis testing. Ann. Statist. 41 (5), 2263–2291.
Smola, A.J., Schölkopf, B., 2000. Sparse Greedy Matrix Approximation for Machine Learning. In: Proceedings of the Seventeenth International Conference on Machine Learning. In: ICML '00, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 911–918. http://dl.acm.org/citation.cfm?id=645529.657980.
Sriperumbudur, B.K., Fukumizu, K., Lanckriet, G.R., 2010. On the relation between universality, characteristic kernels and RKHS embedding of measures. In: Proc. of 13th International Conference on Artificial Intelligence and Statistics. pp. 773–780.
Staaf, J., Lindgren, D., Vallon-Christersson, J., Isaksson, A., et al., 2008. Segmentation-based detection of allelic imbalance and loss-of-heterozygosity in cancer cells using whole genome SNP arrays. Genome Biol. 9 (9), R136.
Williams, C., Seeger, M., 2001. Using the Nyström Method to Speed Up Kernel Machines. In: Advances in Neural Information Processing Systems, vol. 13. MIT Press, pp. 682–688.
Yang, T., 2012. Simple binary segmentation frameworks for identifying variation in DNA copy number. BMC Bioinform. 13 (1), 277.

3.4 Sélection de groupes de variables corrélées en grande dimension

Comme mentionné dans la section 3.2 de ce mémoire, l'utilisation d'approches de classification supervisée avec sélection de variables n'a finalement pas été retenue pour l'analyse des données réelles du laboratoire d'hématologie. Avant de conclure que peu d'anomalies étaient communes entre les patients grâce au travail d'amélioration des méthodes de segmentation effectué en parallèle, nous nous sommes d'abord posé la question de savoir si le Lasso [Tibshirani, 1996] ne souffrait pas du nombre de marqueurs étudiés (1,8 million dans notre jeu de données réelles). En particulier, il était connu dans la littérature que des problèmes de corrélations étaient responsables de l'instabilité apparente des variables sélectionnées [Meinshausen and Bühlmann, 2010]. La grande dimension apporte nécessairement un problème de redondance au sein des données. Plusieurs variables portent la même information, ce qui se traduit par une corrélation élevée entre elles. Le Lasso en sélectionne une, pas toujours la même si on change légèrement l'échantillon de départ, ce qui rend difficile l'interprétation biologique. Pour faciliter l'interprétation, une solution est de sélectionner des groupes de variables plutôt que des variables seules. Cependant, les approches existantes à l'époque comme le group-Lasso [Yuan and Lin, 2006] nécessitaient de fournir des groupes pré-existants. Dans un contexte d'analyse de positions génomiques, cela paraissait difficile de s'appuyer sur des catégories fonctionnelles pré-existantes. Une solution naïve était d'utiliser une approche de classification non supervisée de variables pour définir des groupes mais le choix d'une mauvaise partition aurait alors entraîné de mauvaises performances de la méthode de sélection. C'est dans ce cadre que nous avons initié la thèse de Quentin Grimonprez. L'article ci-après présente le package R MLGL résultant de son travail de thèse. Nous avons proposé une approche innovante combinant classification ascendante hiérarchique et sélection de variables. L'originalité de notre approche est de pouvoir sélectionner des groupes à différents niveaux de la hiérarchie. Cela induit un coût algorithmique, par ailleurs maîtrisé par une implémentation efficace, comme l'explique l'article suivant.

MLGL: An R package implementing correlated variable selection by hierarchical clustering and group-Lasso

Quentin Grimonprez [1]*, Samuel Blanck [3], Alain Celisse [1,2] and Guillemette Marot [1,3]
[1] MΘDAL team, Inria Lille-Nord Europe, France
[2] Laboratoire Paul Painlevé, Université de Lille, France
[3] EA 2694, Université de Lille, France

August 14, 2018

Abstract

The MLGL R-package, standing for Multi-Layer Group-Lasso, implements a new procedure of variable selection in the context of redundancy between explanatory variables, which holds true with high dimensional data. A sparsity assumption is made that is, only a few variables are assumed to be relevant for predicting the response variable. In this context, the performance of classical Lasso-based approaches strongly deteriorates as the redundancy strengthens.

The proposed approach combines variables aggregation and selection in order to improve interpretability and performance. First, a hierarchical clustering procedure provides at each level a partition of the variables into groups. Then, the set of groups of variables from the different levels of the hierarchy is given as input to group-Lasso, with weights adapted to the structure of the hierarchy. At this step, group-Lasso outputs sets of candidate groups of variables for each value of regularization parameter.

The versatility offered by MLGL to choose groups at different levels of the hierarchy a priori induces a high computational complexity. MLGL however exploits the structure of the hierarchy and the weights used in group-Lasso to greatly reduce the final time cost. The final choice of the regularization parameter – and therefore the final choice of groups – is made by a multiple hierarchical testing procedure.

keywords: penalized regression, correlated variables, hierarchical clustering, group selection

1 Introduction

In the high-dimensional setting where the number of variables p is larger than the sample size n, variable selection becomes a challenging problem which is often addressed by regularization procedures such as Lasso [Tibshirani, 1994, Tibshirani et al., 2005, Yuan and Lin, 2006]. These procedures have become very popular since they are specifically designed to select a subset of the explanatory variables for predicting the response. Nevertheless, high dimension raises several problems such as the high correlation level between variables. For instance correlation can be responsible for the apparent instability of the selected variables which can change from one draw to another [Meinshausen and Bühlmann, 2010]. The present work tackles the problem of variable selection in the high-dimensional setting with a strong correlation between explanatory variables.

*to whom correspondence should be addressed: quentin.grimonprez@inria.fr

Let X denote a $n \times p$ matrix where each column vector $X_j \in \mathbb{R}^n$ $(1 \leq j \leq p)$ corresponds to the values of the jth variable measured on n individuals. The quantitative response vector $y \in \mathbb{R}^n$ is then related to X through the linear regression model

$$y = X\beta^* + \epsilon, \tag{1}$$

where $\epsilon \sim \mathcal{N}_n(0, \sigma^2 I_n)$ is a Gaussian vector (noise), and $\beta^* \in \mathbb{R}^p$ is the parameter vector encoding the influence of each of the p candidate variables on the response y. The intercept of the regression model is removed by assuming X_j is centered for all $j = 1, \ldots, p$.

Moreover, the parameter vector β^* is assumed to be sparse that is, the cardinality of its support $S^* = S(\beta^*) = \{1 \leq j \leq p \mid \beta_j^* \neq 0\}$ is such that

$$\mathrm{Card}(S^*) = k \ll p.$$

This is consistent with the goal of identifying a small subset of interpretable (groups of) variables which turn to be relevant in explaining the response.

The first naive approach for estimating β^* from Eq. (1) is to compute the minimizer of the least squares error

$$\beta^{\mathrm{LS}} \in \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2} \|y - X\beta\|_2^2 \right\}. \tag{2}$$

However in the present high-dimensional context where $p \gg n$, there are infinitely many solutions to this problem and most of them are certainly not sparse.

The Lasso procedure [Tibshirani, 1994] is generally used to perform variable selection in this high-dimensional setting. Unlike the above least squares minimization problem, a regularization term consisting of the ℓ_1-norm of the estimated vector (the penalty) is added to get a unique and sparse solution to the following optimization problem:

$$\beta_\lambda^{\mathrm{Lasso}} = \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \right\}, \tag{3}$$

where $\lambda > 0$ is called the regularization parameter and controls the amount of shrinkage. For instance, a large value of λ yields an estimator with only a few non-zero coefficients. In practice, the calibration of λ can be done by means of V-fold cross-validation [Arlot and Celisse, 2010] or various information criteria such as AIC, BIC, ...

Although (asymptotic) consistency results on the selected variables have been proven [Zhao and Yu, 2006], establishing such consistency results with highly correlated variables remains highly challenging or even impossible if the correlation is too strong [Wainwright, 2009]. Intuitively, Lasso selects one (or a few) variable(s) among each group of correlated variables as long as the correlation is strong enough, even if all these variables belong to the true support S^*. In such a case grouping correlated variables turns out to be necessary to select meaningful groups of influential variables. The group-Lasso [Yuan and Lin, 2006] was precisely developed for taking into account the *a priori* knowledge of groups of (correlated) variables. More precisely given a partition of the p candidate variables into g groups $\mathcal{G} = \{G_1, \ldots, G_g\}$, the group-Lasso estimator is defined by

$$\beta_\lambda^{\mathcal{G}} = \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2} \|y - X\beta\|_2^2 + \lambda \sum_{i=1}^{g} w_i \|\beta_{G_i}\|_2 \right\}, \tag{4}$$

where $\lambda > 0$ is the regularization parameter, and $w_i > 0$ denotes the weight associated with the group G_i (generally $w_i = \sqrt{\text{Card}(G_i)}$). Obviously, the statistical performance of the group-Lasso estimator strongly depends on the partition $\mathcal{G}$ that has to be known *a priori*. When no such knowledge is available regarding groups of correlated variables, a preliminary step aiming at providing a meaningful partition of the candidate variables is crucial.

Several strategies such as first grouping candidate variables and then selecting groups by Lasso or group-Lasso have been studied in the literature. Most of them rely on hierarchical clustering at the first stage where only one level of the hierarchy is chosen (resulting in a partition of the candidate variables). For example [Park et al., 2007] perform hierarchical clustering first. Then Lasso is successively applied to each level of the hierarchy where each candidate group is summarized by a representative variable. Both the hierarchy level and the subset of groups from the corresponding partition are selected by cross-validation. By contrast, Cluster Representative Lasso and Cluster Group-Lasso [Bühlmann et al., 2013] apply hierarchical clustering and choose first one particular level of the hierarchy. Then groups from this partition are selected either by using Lasso (applied to representative variables of each group) or by using the corresponding partition as an input of group-Lasso. Let us also mention alternative strategies such as Supervised Group-Lasso [Ma et al., 2007] and Cluster Elastic Net [Witten et al., 2014] to name but a few. One main contribution of the present work is to relax the dependence of the final selected (groups of) variables on a particular level of the hierarchy. The main asset is some robustness to possible mistakes resulting from the iterative clustering process. Our procedure combines hierarchical clustering and group selection by allowing group-Lasso for selecting groups from different hierarchy levels that is, from different partitions of the candidate variables.

The following of the paper is organized as follows. Section 2 introduces the whole procedure that is successively based on hierarchical clustering (AHC), group-Lasso (gLasso), and a post-treatment selection involving hierarchical multiple testing (HMT). Then, the usage of the R-package MLGL is described in Section 3. The statistical performance of the procedure is assessed in Section 4 by comparison to alternative ones. Finally, some conclusions and perspectives are discussed in Section 5.

2 Overview of the MLGL package

Generally group-Lasso is applied with only one prescribed partition of the variables into groups (corresponding in the present context to one particular level of the hierarchy). One main originality of the present package is to select groups of variables by applying group-Lasso to several partitions at the same time. A possible resulting issue is the presence of overlapping groups in the partitions given as inputs to group-Lasso.

The whole procedure implemented in the *MLGL package* (standing for Multi-Layer Group-Lasso) consists of four main steps:

1. Building a hierarchy (hierarchical clustering),
2. Computing the path of groups selected by group-Lasso with respect to $\lambda > 0$ (the regularization parameter),
3. Performing hierarchical multiple testing (HMT) to remove false positive groups for each λ,
4. Tuning λ to select the final groups of influential variables.

These different steps are detailed in what follows.

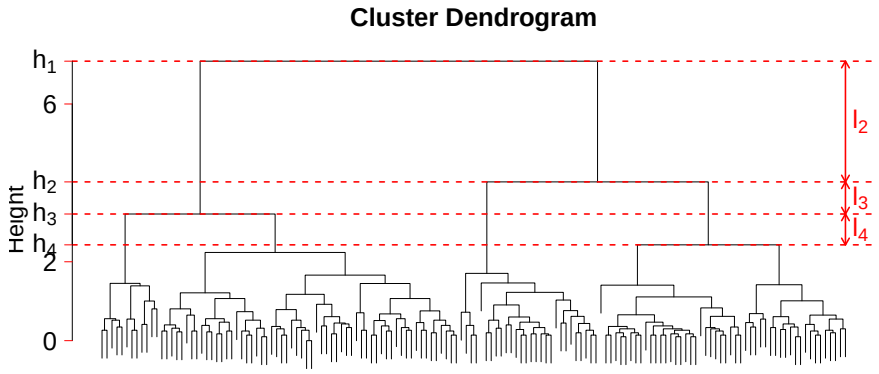


Figure 1: Dendrogram obtained using a hierarchical algorithm.

2.1 Building a hierarchy

Two main families of methods co-exist for performing (unsupervised) clustering: *hierarchical clustering* algorithms and the so-called *partitional* algorithms (see [Jain et al., 1999] for a review). The main difference lies in that partitional algorithms return only one partition of the candidate variables into a prescribed number of groups (k-means for instance), whereas hierarchical clustering algorithms yield a nested hierarchy of partitions of the candidate variables. This hierarchy can be represented by a dendrogram (Figure 1), so that each hierarchy level defines a partition of the candidate variables into groups. Moreover the hierarchy enjoys the property that each group at a given level can be split into sub-groups located at different sub-levels of this hierarchy as illustrated by Fig. 1.

The general process of hierarchical clustering is summarized in Pseudo-code 1. A similarity

Pseudo-code 1 Ascendent Hierarchical Clustering (AHC)

Input: Candidate variables, similarity measure

Compute the distance matrix between all variables.
Place each variable in its own group.
repeat
Aggregate the two nearest groups according to the similarity measure.
until all the variables belong to the same group.

Return: Dendrogram

measure has to be specified and determines the order in which (groups of) variables will be aggregated. Classical similarity measures are the Ward's criterion (which minimizes the total within-group variance) and the average linkage (which aggregates the two groups minimizing the average distance between each pair of points (one from each group)).

Considering the level $s \in \{1, \ldots, p\}$ of the hierarchy where the variables are partitioned into s groups, let h_s denote the value of the similarity measure between the two groups merged for obtaining the partition with s groups, and the jump size $l_s = h_{s-1} - h_s$ (see Figure 1). Choosing the number of groups can be performed following the *highest jump* rule, which consists in choosing the partition $\mathcal{G}_{\hat{s}}$ such that

$$\hat{s} = \underset{s}{\operatorname{argmax}}\{l_s\}. \tag{5}$$

Intuitively, a large value of l_s indicates that the groups merged from level s to $s-1$ were far apart according to the similarity measure. This explains why the partition with s groups is usually preferred in this setting.

In the MLGL package, there is no need to choose the number of groups output from the hierarchical clustering since all levels of the hierarchy are kept as an input of group-Lasso. The latter selects simultaneously the number of groups as well as the groups. Nevertheless, the jump sizes are exploited as weights within the group-Lasso procedure, which turns out to reduce the whole computational cost (see Section 2.2).

2.2 Computing the path of candidate groups

One main originality of the MLGL package is to simultaneously provide the groups from all levels of the hierarchy as an input to group-Lasso. The resulting procedure should be less sensitive to possible mistakes induced by the iterative clustering process.

Since no selection of a particular hierarchy level is made, numerous overlapping groups arise in the input of group-Lasso. With overlapping groups, [Jacob et al., 2009] designed a overlap group-Lasso penalty and expressed it in such a way they could apply classical algorithms to minimize the group-Lasso problem to solve the overlap group-Lasso problem. The trick is exposed in what follows.

From a collection $\mathcal{G} = \{G_1, \ldots, G_g\}$ of $g \in \mathbb{N}^*$ groups of indices such that $G_i \subset \{1, \ldots, p\}$, for all $i = 1, \ldots, g$, let us introduce X_{G_i} as the $n \times \text{card}(G_i)$ matrix obtained by concatenating the columns of X corresponding to variables with indices in G_i. Let also $X^{\mathcal{G}} = [X_{G_1}, X_{G_2}, \ldots, X_{G_g}]$ denote the $n \times l$ extended design matrix defined as the concatenation of the matrices $X_{G_1}, X_{G_2}, \ldots, X_{G_g}$, where $l = \sum_{i=1}^{g} \text{card}(G_i)$. Then the overlap group-Lasso estimator built from the design matrix X and the collection $\mathcal{G}$ can be expressed as a group-Lasso estimator with extended design matrix $X^{\mathcal{G}}$ as

$$\hat{\beta}_\lambda^{\mathcal{G}} = \underset{\beta \in \mathbb{R}^{pl}}{\text{argmin}} \left\{ \frac{1}{2} \|y - X^{\mathcal{G}} \beta\|_2^2 + \lambda \sum_{i=1}^{g} w_i \|\beta_{G_i}\|_2 \right\}, \tag{6}$$

where $\lambda > 0$ is the regularization parameter and w_i denotes a weight associated with G_i. This rephrasing allows for using all the partitions output by the hierarchical clustering as an input of group-Lasso.

Considering the dendrogram output by hierarchical clustering, let $\mathcal{G}_s$ be the partition of the p candidate variables into s groups, for $1 \leq s \leq p$, and $\mathcal{G}_* = \cup_{s=1}^{p} \mathcal{G}_s$ denote the union of all the partitions at the different levels of the hierarchy. Then the above Eq. (6) applied with $\mathcal{G} = \mathcal{G}_*$ leads to

$$\hat{\beta}_\lambda^{\mathcal{G}_*} = \underset{\beta \in \mathbb{R}^{p^2}}{\text{argmin}} \left\{ \frac{1}{2} \|y - X^{\mathcal{G}_*} \beta\|_2^2 + \lambda \sum_{s=1}^{p} \rho_s \sum_{i=1}^{g_s} w_i^s \|\beta_{G_i^s}\|_2 \right\}, \tag{7}$$

where G_i^s is the ith group of the partition $\mathcal{G}_s$ and $\mathcal{G}_s = \cup_{i=1}^{g_s} G_i^s$, $X^{\mathcal{G}_*} = [\underbrace{X, \ldots, X}_{p \text{ times}}]$ denotes the corresponding extended design matrix, and ρ_s is a weight encoding how likely $\mathcal{G}_s$ is a meaningful partition of the candidate variables.

It is worth noticing that Eq. (7) shows that the present approach is included in the general framework described in [Jenatton et al., 2011], where penalties are designed to define groups according to a prescribed structure in the support of β^*.

Choice of ρ_s For $s = 1, \ldots, p$, ρ_s is a weight reflecting the quality of the partition $\mathcal{G}_s$. This weight must weakly penalize a "good" partition and heavily penalize a "bad" one. The MLGL package uses a weight ρ_s inspired from the somewhat classical highest jump rule that is, a small weight is given to partitions with a large jump size l_s. More precisely,

$$\rho_s = \frac{1}{\sqrt{l_s}} \ . \tag{8}$$

It is important to keep in mind that this definition of ρ_s promotes the selection of groups belonging to the partition with the largest jump size. But the described procedure remains free to select groups from different partitions (from different hierarchy levels).

Storage improvement From the reformulation in Eq. (7), it clearly arises that several duplications of the $n \times p$ design matrix X are used. The extended design matrix $X^{\mathcal{G}_*}$ has size $n \times p^2$ when all the levels from the hierarchy are kept as an input. In usual high-dimensional settings, the p^2 columns induce a prohibitive computational cost both in space and time. Therefore, the MLGL package exploits the redundancy of the partitions along the hierarchy to drastically reduce the computational costs.

On the one hand, let us notice that two successive partitions from a hierarchy — say $\mathcal{G}_s$ and $\mathcal{G}_{s-1}$ the ones with respectively s and $s-1$ groups — share $s-2$ common groups: At each step of the hierarchical clustering process, only two groups are aggregated while the others remain unchanged. On the other hand, these groups (which remain the same from a level $\mathcal{G}_{s-1}$ to the next one $\mathcal{G}_s$) are penalized with a different weight depending on the partition they belong to. More precisely, each such group is weighted once with ρ_s and once with ρ_{s-1}. The following Lemma 1 establishes that if $\rho_{s-1} \neq \rho_s$, then only the group with the smallest weight has a chance to be selected. The proof is given in Appendix A.

Lemma 1. *With the notations of Eq.* (6)*, let $\mathcal{G}$ denote any collection of g subsets (groups) of $\{1, \ldots, p\}$ that are not necessarily disjoint and assume that there exist $G_1, G_2 \in \mathcal{G}$ such that $G_1 = G_2$, with $w_2 > w_1 > 0$.*

Then the solution $\hat{\beta}^{\mathcal{G}}_\lambda \in \mathbb{R}^l$ of Eq. (6) *satisfies that the subset of its coordinates corresponding to G_2 is equal to zero that is, $(\hat{\beta}^{\mathcal{G}}_\lambda)_{G_2} = 0$.*

From several copies of the same group with different weights, only the one with the smallest weight is worth considering according to Lemma 1. This justifies simplifying the optimization problem from Eq. (7) to drastically reduce the induced computational costs.

Let us define $\mathcal{G}^{\mathrm{u}}_*$ as the collection of all the *distinct* groups output from hierarchical clustering (without including copies) that is,

$$\mathcal{G}^{\mathrm{u}}_* = \bigcup_{i=1}^{2p-1} G^{\mathrm{u}}_i, \quad \text{such that} \qquad \forall 1 \leq i \neq j \leq 2p-1, \quad G^{\mathrm{u}}_i \neq G^{\mathrm{u}}_j.$$

This new collection $\mathcal{G}^{\mathrm{u}}_*$ exactly contains $2p-1$ distinct groups: p groups made of one variable from the pth level of the hierarchy (the leaves of the dendrogram), and one new group from each other level (there are $p-1$ of them). The resulting extended design matrix $X^{\mathcal{G}^{\mathrm{u}}_*}$ is clearly less space demanding than the former $X^{\mathcal{G}_*}$. Consistently with the above remarks, the optimization problem from Eq. 7 can be equivalently reformulated as

$$\hat{\beta}^{\mathcal{G}^{\mathrm{u}}_*}_\lambda = \operatorname*{argmin}_\beta \left\{ \frac{1}{2}\|y - X^{\mathcal{G}^{\mathrm{u}}_*}\beta\|_2^2 + \lambda \sum_{i=1}^{2p-1} \rho^{\mathrm{u}}_i w^{\mathrm{u}}_i \|\beta_{G^{\mathrm{u}}_i}\|_2 \right\}, \tag{9}$$

with $\lambda > 0$ the regularization parameter, w_i^{u} the weight associated with G_i^{u}, and ρ_i^{u} the smallest weight associated with one partition containing G_i^{u}, that is

$$\rho_i^{\mathrm{u}} = \min \{\rho_s \mid s \in 1, \ldots, p \text{ such that } G_i^{\mathrm{u}} \in \mathcal{G}_s\}.$$

Since this simplified problem is an instance of group-Lasso as earlier discussed at Eq. (6), the MLGL package solves Eq. (9) by means of classical optimization algorithms solving the group-Lasso problem [Yang and Zou, 2015]. In particular, such an algorithm gives access to the whole path $\lambda \mapsto \hat{\beta}_\lambda^{\mathcal{G}_*^{\mathrm{u}}}$ of the candidate groups selected by group-Lasso for each λ.

2.3 Hierarchical Multiple Testing

For each λ, the previous step returns a set of selected groups of variables from which, most of the time, an additional filtering step is required for two main reasons. First, it is well known that in the high-dimensional context where the number of (groups of) variables is larger than n and only a few candidate variables are likely to be influential (sparsity assumption), then Lasso and its extensions can only identify most of the true variables at the price of including false positives among the selected ones [Wainwright, 2009, Barber et al., 2015]. Second, the solution of Eq. (9) contains groups potentially located at different levels of the hierarchy. Furthermore some groups can even be sub-groups of some others as explained by Figure (2) (redundancy of groups). Then choosing which one from the group or its sub-group should be selected has to be done by an additional dedicated step.

For all these reasons, the MLGL package applies a hierarchical multiple testing procedure (HMT) which selects the final groups for each value of λ. The choice of the regularization parameter λ is discussed in Section 2.4. The next two paragraphs review the main goals the HMT procedure achieves for a given value of λ: (i) reducing the number of selected groups, and (ii) avoiding the redundancy of groups.

2.3.1 Reducing the number of groups

With Lasso, [Wasserman and Roeder, 2009] suggest to perform a least squares estimation of the coefficients of the selected variables, so that they test the nullity of each coefficient by means of multiple testing procedures. Adjusted p-values are computed for controlling the Family-Wise Error Rate (FWER) [Dunn, 1959] or the False Discovery Rate (FDR) [Benjamini and Hochberg, 1995].

With group-Lasso, it can happen that more variables than individuals are selected at a given λ value (in particular when λ is very close to 0). A least squares estimation cannot be directly performed in this situation. This issue can be overcome by first summarizing each selected group by one representative variable and then performing least squares estimation using these representative variables. Note that this is always possible since the number of selected groups cannot be larger than the number of individuals [Liu and Zhang, 2009].

In the MLGL package, the representative variable summarizing each group output by group-Lasso is first computed by means of the first principal component. Then, the least squares estimators of the coefficients of each representative variable are computed. Finally, all p-values resulting from the test of the nullity of the estimated coefficients are corrected following Bonferroni's procedure [Dunn, 1959], which allows for controlling the FWER. This three-step procedure is described by Pseudo-code 2.

Pseudo-code 2 Reducing the number of groups

Input: Groups selected by group-Lasso for a given λ: $G_1^\lambda, \ldots, G_m^\lambda$

1- *Compute* the first principal component $\dot{X}_i$ of X_{G_i}, for all $i = 1, \ldots, m$.

2- From $\dot{X} = [\dot{X}_1, \ldots, \dot{X}_m]$ and the model $y = \dot{X}\dot{\beta} + \epsilon$,
compute $\hat{\dot{\beta}}$ the least-squares estimator of $\dot{\beta}$.

3- *Test* the nullity of the coefficients, *apply* the multiple testing correction to the corresponding p-values [Dunn, 1959], and *reject* all null hypotheses with an adjusted p-value lower than the prescribed level.

Output: The set of rejected null hypotheses.

2.3.2 Avoiding the redundancy of groups

As exposed in Section 2.2, the MLGL package allows for selecting groups from different levels of the hierarchy, which especially arises with small values of λ. It can therefore happen that one selected group is included in another one. It is then desirable to select only this group or its subgroup, but not both of them. This can be achieved by applying a hierarchical testing procedure (HTP) for controlling the FWER [Meinshausen, 2008].

The intuitive idea is to select the smallest possible groups of variables with a significant effect on the response variable. In particular this would avoid including a large group of variables with only a few of them being truly influential ones.

From a hierarchical tree (see Figure 2a), the importance of groups is tested sequentially with partial F-tests, which have been extensively used in the context of nested models in multiple linear regression problems [Jamshidian et al., 2007]. The importance of a group G of variables is tested with the following hypotheses:

$$H_{0,G} \ : \ \beta_G = 0, \qquad \text{versus} \qquad H_{1,G} \ : \ \exists i \in G, \ \beta_i \neq 0,$$

where β_i is the coefficient corresponding to the variable index $i \in G$, and $\beta_G = 0$ encodes that the group G has no influence on the response y.

HTP starts by testing the group containing all the variables at the top of the hierarchical tree. Then, for any rejected null hypothesis $H_{0,G}$, the null hypotheses associated with the children of group G (subgroups of G) are subsequently tested. The process is repeated until no more null hypothesis is rejected. Each computed p-value is adjusted following Bonferroni's procedure for controlling the FWER [Dunn, 1959].

2.3.3 The MLGL processing of the candidate groups

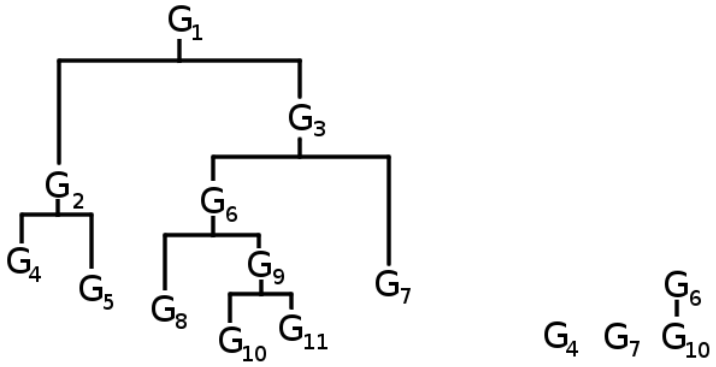
Let us consider the collection of candidate groups selected at the end of Section 2.2 for a given value of λ. At this stage, the MLGL package faces the two problems mentioned above that is, multiplicity and redundancy. This is the goal of the HMT procedure implemented in the MLGL package to overcome these problems.

More precisely the HMT procedure starts by splitting the selected groups into d disjoint hierarchical trees (denoted by $\mathcal{T}_i$, $i = 1, \ldots, d$) and one set $\mathcal{S}$ of candidate groups with no hierarchical structure (see Example 1).

Example 1 (Separate the selected groups in hierarchical trees)**.** *Let us consider a hierarchy built from 6 variables with groups as follows:* $G_1 = \{1, 2, 3, 4, 5, 6\}$, $G_2 = \{1, 2\}$, $G_3 = \{3, 4, 5, 6\}$, $G_4 = \{1\}$, $G_5 = \{2\}$, $G_6 = \{3, 4, 5\}$, $G_7 = \{6\}$, $G_8 = \{3\}$, $G_9 = \{4, 5\}$, $G_{10} = \{4\}$, $G_{11} = \{5\}$. *The resulting hierarchy is displayed in Figure 2a.*

For a specific value of λ, let us assume that the groups G_4, G_6, G_7, and G_{10} are selected (see Figure 2b).

Then the HMT procedure defines one set $\mathcal{S} = \{G_4, G_7\}$ and one hierarchical tree $\mathcal{T}_1 = \{G_6, G_{10}\}$, where $G_{10} \subset G_6$.



(a) Hierarchical tree from Example 1

(b) Selected groups from Example 1

Figure 2: Illustration from Example 1.

An important remark is that hierarchical trees must be *complete* that is, each group in the tree $\mathcal{T}_i$ is either a leaf (a group without any subgroups) or the union of its subgroups. This is a necessary requirement of our strategy since the importance of a candidate group G is tested through its leaves (subgroups of G without any children). If a group (which is not a leaf) is not the union of its children in the hierarchical tree, then the hierarchical testing procedure of [Meinshausen, 2008] cannot be properly applied. Therefore, some groups are added to the hierarchical tree for completing hierarchies which are not *complete* (see Example 2).

Example 2 (Complete a hierarchical tree)**.** *The groups $G_6 = \{3, 4, 5\}$ and $G_{10} = \{4\}$ from the hierarchical tree $\mathcal{T}_1$ in Example 1 do not form a complete hierarchy (G_6 is not equal to the union of its subgroups).*

The group $\bar{G}_{10} = \{3, 5\}$ is then defined as the complement of G_{10} within G_6, which leads to the new (full) hierarchical tree $\bar{\mathcal{T}}_1 = \{G_6, G_{10}, \bar{G}_{10}\}$.

The completed hierarchical trees are denoted by $\bar{\mathcal{T}}_1, \ldots, \bar{\mathcal{T}}_d$.

In addition, applying the HTP procedure from [Meinshausen, 2008] also requires to summarize each group within each hierarchical tree by a representative variable. This is done by the MLGL package by computing the first principal component of each group. The new corresponding trees are denoted by $\dot{\mathcal{T}}_1, \ldots, \dot{\mathcal{T}}_d$. Therefore the HTP procedure of [Meinshausen, 2008] is applied to $\dot{\mathcal{T}}_1, \ldots, \dot{\mathcal{T}}_d$ (see Pseudo-code 3).

Controlling the FWER level With the same notation as Section 2.3.3, let us define the cardinality of any hierarchical tree as the number of leaves it contains, and set $m = |\mathcal{S}| + \sum_{i=1}^{d} |\dot{\mathcal{T}}_i|$, where $|A|$ denotes the cardinality of the set A. Then, the HMT procedure implemented in the MLGL package controls the FWER of the tree $\dot{\mathcal{T}}_i$ (Pseudo-code 3) at level $\frac{\alpha|\dot{\mathcal{T}}_i|}{m}$, and that of the set $\mathcal{S}$ at level $\frac{\alpha|\mathcal{S}|}{m}$. It results that the global HMT procedure described by Pseudo-code 4 truly controls the FWER at the overall prescribed level $0 < \alpha < 1$.

Pseudo-code 3 Hierarchical testing procedure for one tree

Input: Any $\mathcal{T} \in \{\mathcal{T}_1, \ldots, \mathcal{T}_d\}$.

Complete hierarchical trees Add missing groups to the hierarchical tree $\mathcal{T}$ to get a *complete* tree $\bar{\mathcal{T}}$.

Summarize the influence of each group Compute the first principal component of each group in the tree $\bar{\mathcal{T}}$. The resulting hierarchical tree is denoted by $\dot{\mathcal{T}}$.

Hierarchical testing Apply the HTP procedure of [Meinshausen, 2008] to the tree $\dot{\mathcal{T}}$ for a prescribed level of control.

Output: Selected groups from $\dot{\mathcal{T}}$.

Pseudo-code 4 Hierarchical multiple testing (HMT) for a given regularization level

Input: List of groups selected after the group-Lasso step for a given $\lambda \in \Lambda$
(Λ: set of candidate regularization parameters).

Define hierarchical trees Split the groups into hierarchical trees $\mathcal{T}_1, \ldots, \mathcal{T}_d$ and the set $\mathcal{S}$. Set $m = |\mathcal{T}_1| + \cdots + |\mathcal{T}_j| + |\mathcal{S}|$.

Testing procedure for hierarchical trees For each hierarchical tree $\mathcal{T}_i$ for $i = 1, \ldots, d$, apply Pseudo-code 3 to get the global control level $\frac{\alpha \times |\mathcal{T}_i|}{m}$.

Testing procedure for groups not belonging to a tree For the set $\mathcal{S}$, apply Pseudo-code 2 to get the global control level $\frac{\alpha \times |\mathcal{S}|}{m}$.

Avoiding over-fitting In order to avoid overfitting, it is necessary to use different individuals for using group Lasso and applying the hierarchical testing procedure.

The set $\mathcal{I} = \{1, \ldots, n\}$ of indices associated with individuals is randomly split into two parts of equal size, say $\mathcal{I}_1$ and $\mathcal{I}_2$. The hierarchical clustering of the variables is first performed from the set $\mathcal{I}_1$. Then group-Lasso is applied from the individuals in $\mathcal{I}_2$ and the previously computed hierarchy. Finally, the whole HMT procedure (namely Pseudo-code 4) is applied for the individuals from $\mathcal{I}_1$. In order to ease the understanding, the whole procedure consisting of "AHC+gLasso+HMT" is summarized in Psuedo-code 5.

Pseudo-code 5 AHC+gLasso+HMT

1. Randomly split the sample indexed by I into two subsets of equal cardinality: $\mathcal{I}_1$ and $\mathcal{I}_2$.
2. Perform AHC of candidate variables from $\mathcal{I}_1$.
3. Perform group-Lasso (9) from $\mathcal{I}_2$.
4. Apply the HMT procedure (namely Pseudo-code 4) from $\mathcal{I}_1$.

2.4 Selecting the final groups by choosing λ

The groups output at the previous steps of the MLGL package (AHC+gLasso+HMT) depend on the value of the regularization parameter $\lambda \in \Lambda$, which is a crucial choice. Several papers have raised the problem of choosing the value of λ in penalized regression frameworks

[Fan and Tang, 2013, Sun et al., 2013]. For instance, resampling-based approaches have been suggested. Among them, choosing the value of λ which yields the most stable selected variables have been explored by [Meinshausen and Bühlmann, 2010], which intensively relies on bootstrap. An alternative consists in tuning λ by means of V-fold cross validation [Arlot and Celisse, 2010]. However both these approaches are highly time-consuming due to the multiple executions they require. Moreover V-fold cross-validation is more suited to the estimation/prediction purpose than to the identification/selection of influential variables. This aspect arises more clearly in difficult settings where the signal-to-noise ratio becomes small. Then, V-fold cross-validation tends to include superfluous variables (false positives). Furthermore information criteria such as AIC [Akaike, 1974] and BIC [Schwarz, 1978] need an estimator of both the degrees of freedom and the unknown variance σ^2 [Giraud et al., 2007]. However if the number of candidate variables is larger than the number of observations, such a consistent estimator of σ^2 is difficult to design [Fan et al., 2012].

One important feature of the procedures implemented in the MLGL package is that the FWER is kept under control whatever the value of $\lambda \in \Lambda$. Furthermore since the proposed procedure turns out to be conservative (from our empirical experiments), the MLGL package chooses the value of λ maximizing the number of rejections. The simulation results discussed in Section 4 seem to support this choice since maximizing the number of rejections turns out to maximize in the same time the number of true positives (while keeping the number of false positives under control).

3 Usage of the MLGL package

The main function of the MLGL package is `fullProcess`. It enables to run the whole procedure consisting in AHC+gLasso+HMT.

For illustration purpose, we generate simulated data with the function `simuBlockGaussian`. In what follows, $n = 50$ individuals and $p = 60$ candidate variables are simulated from a multivariate Gaussian $\mathcal{N}(0,\Sigma)$ distribution. The covariance matrix Σ has a block-diagonal structure where each block of 5 variables has 1 on the diagonal and $\rho = 0.7$ elsewhere, that is

```
X <- simuBlockGaussian(n= 50, nBlock=12,sizeBlock= 5, rho= 0.7)
```

Two probabilistic models are considered in the MLGL package: the linear and the logistic ones.

- With the linear model, let us simulate

  ```
  y <- drop(X[,c(2,7,12)]%*%c(2,2,-2) + rnorm(50, 0, 0.5))
  ```

  Then, applying the function `fullProcess` is done by means of:

  ```
  res <- fullProcess(X, y)
  ```

- With the logistic model, binary observations are generated by

  ```
  y <- 2*(rowSums(X[,1:4])>0)-1
  ```

  Then, the function `fullProcess` can be processed by:

  ```
  res <- fullProcess(X, y, loss = "logit", test = partialChisqtest)
  ```

In addition to this main function, the MLGL package contains functions enabling to perform different steps of the procedure. For instance, the `MLGL` function computes the path of candidate groups output after AHC+gLasso.

Alternative procedures to HMT are also implemented in the MLGL package to select final groups. For instance, `cv.MLGL` and `stability.MLGL` can be applied to choose λ by respectively V-fold cross-validation and bootstrap. More precisely, the first one returns the mean cross-validation error (mean squared error for the linear case or area under the ROC curve for the logistic case) for a prescribed sequence of regularization parameter values. Instead, the second one performs the stability selection procedure [Meinshausen and Bühlmann, 2010] where the probability of selecting each group is estimated for every value of the prescribed sequence of regularization parameter values. Let us also mention that the paths returned by these two functions can be independently generated by the functions `plot.MLGL`, `plot.cv.MLGL`, and `plot.stability.MLGL` (see Figure 3):

```
res <- MLGL(X, y)
plot(res)
res.cv <- cv.MLGL(X, y, loss = "logit")
plot(res.cv)
res.stab <- stability.MLGL(X, y, loss = "logit")
plot(res.stab)
```

4 Comparison of MLGL to other selection procedures

In the present section, the solution paths output by different procedures will be compared to that one provided by the MLGL package by plotting the number of true positives versus the number of false positives.

Let us generate n realizations of independent and identically distributed random variables $X_1, \ldots, X_n \in \mathbb{R}^p$ from a multivariate Gaussian distribution $\mathcal{N}(0_p, \Sigma)$, where Σ is a $p \times p$ covariance matrix with a block-diagonal structure. The common size of the blocks is l, and all the blocks have 1 on their diagonal and ρ everywhere else.

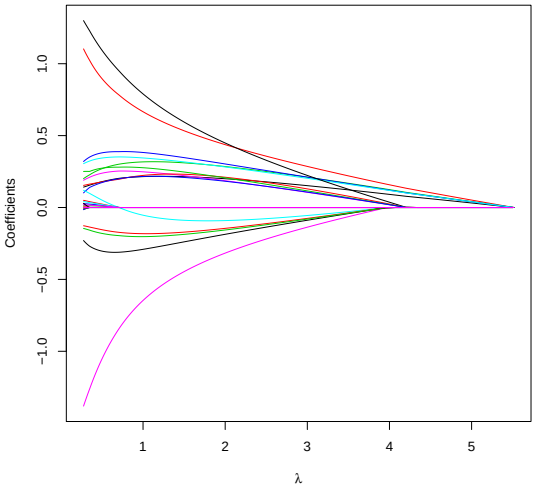
The response variable is generated from the model $y = X\beta^* + \epsilon$, where $\beta^* \in \mathbb{R}^p$ is a sparse vector with 1s for K elements corresponding to different blocks of Σ, and ϵ denotes a random Gaussian variable. Note that the noise level is set such that the signal-to-noise ratio has a value of 2.

In the present simulation design, a selected group is called *true positive* if it contains exactly one variable belonging to the support of the true solution β^*, as well as other variables that are correlated with this one but do not belong to the support of β^*. Conversely a group is termed as a *false positive* if it contains either no variable belonging to the support of β^*, or several (uncorrelated) variables belonging to the true support.

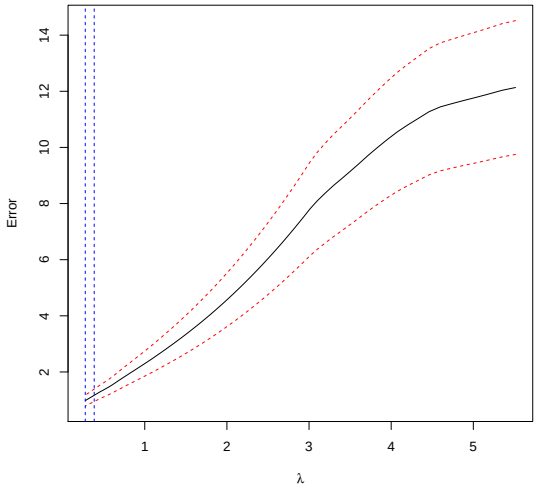
4.1 Comparison of *Multi-Layer Group-Lasso* with group-Lasso

The output of the MLGL package is first compared to that of the classical group-Lasso which essentially focuses on only one level of the hierarchy.

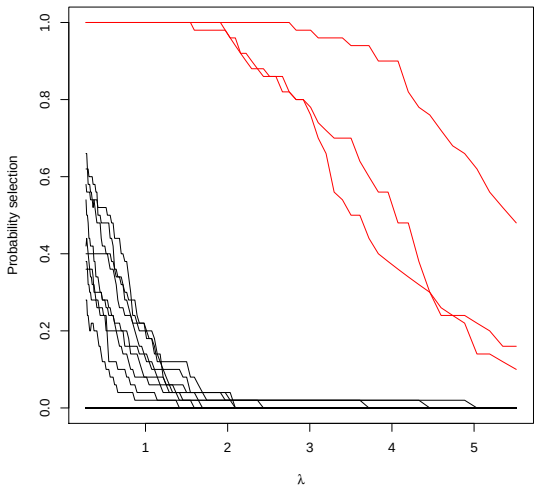
The AHC step is performed based on the Euclidean distance and Ward's criterion. The highest jump rule selects the partition of the candidate variables (level of the hierarchy) that is taken as an input of the classical group-Lasso. The MLGL package uses the weights defined in Eq. (8), which (also) involves the highest jump rule and allows for selecting groups from different levels of the hierarchy.



(a) Solution path (`plot.MLGL`)

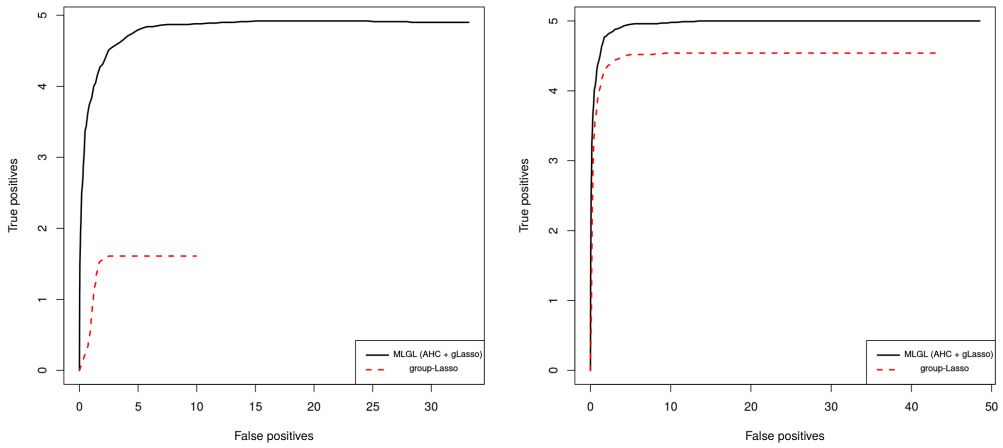


(b) CV error (`plot.cv.MLGL`)



(c) Probability selection (`plot.stability.MLGL`)

Figure 3: Plots generated by `plot.MLGL`, `plot.cv.MLGL` and `plot.stability.MLGL`. The plot generated by `plot.MLGL` represents the solution path of MLGL with each curve corresponding to the estimated coefficients of a variable according to the regularization parameter. The cross-validation error is the output of `plot.cv.MLGL`; the vertical lines correspond to the λ which minimizes the cross-validation error and the largest value of λ such that error is within one standard error of the minimum. `plot.stability.MLGL` shows the probability selection for the different groups, the red curves being the selected groups.



(a) $n = 100$, $p = 500$, $\rho = 0.5$, $l = 5$, $K = 5$ (b) $n = 100$, $p = 500$, $\rho = 0.7$, $l = 5$, $K = 5$

Figure 4: Number of true positives versus the number of false positives in the solution path output by the MLGL package before hierarchical multiple testing (black solid line) and classical group-Lasso (red dashed line). The curves represent the mean calculated over 100 replicates.

Figure 4 displays the number of true and false positives along the solution path output by the MLGL package and the classical group-Lasso. For a given number of false positives, more true positives are provided by the two first steps of the MLGL package (AHC+gLasso) than by the classical group-Lasso.

The gap between the two solution paths can be explained by the way the partition used by the group-Lasso is chosen. From Figure 5 (left panel), it arises that the highest jump rule fails to recover the optimal partition which has 100 groups in the present simulation experiments. In such cases, group-Lasso selects groups among poor candidates whereas the MLGL package is less sensitive to such a bad preliminary choice.

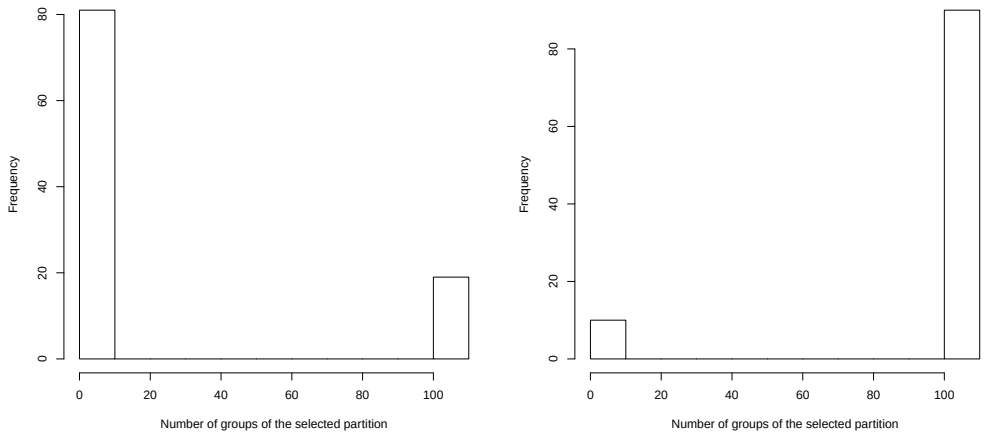
4.2 Comparison to alternative approaches combining clustering and selection

The performance of the MLGL package is now compared to that of alternative procedures combining clustering and selection: Hierarchical Clustering and Averaging for Regression (HCAR) [Park et al., 2007], Supervised Group-Lasso (SGL) [Ma et al., 2007], Cluster Representative Lasso (CRL) and Cluster Group-Lasso (CGL) [Bühlmann et al., 2013]. Note that all these procedures combine a clustering step (hierarchical clustering or k-means) with a selection step (Lasso, group-Lasso, or standardized group-Lasso [Bühlmann and van de Geer, 2011, Simon and Tibshirani, 2011]).

For all these methods a clustering is performed based on the Euclidean distance and Ward's criterion. When the method requires only one partition, this one is chosen by the highest jump rule. For HCAR, $\hat{\lambda}$ is chosen by cross-validation and only the corresponding solution path is output.

Figure 6 displays the number of true and false positives along the solution path of the competing procedures for different values of the parameters.

The MLGL package turns out to provide results among the best ones since the maximal num-



(a) $n = 100$, $p = 500$, $\rho = 0.5$, $l = 5$, $K = 5$ (b) $n = 100$, $p = 500$, $\rho = 0.7$, $l = 5$, $K = 5$

Figure 5: Size (in number of groups) of the partition selected by the highest jump rule.

ber of true positives ($K = 5$ or 10) is reached with only a few false positives. It is noticeable that *Cluster Representative Lasso* and *Supervised Group-Lasso* exhibit similar performances (schemes b, c and d).

When the correlation ρ rises from 0.5 to 0.9 between Figures 6a and 6b, the performance of *HCAR* and *CGL* heavily deteriorates whereas the other procedures remains almost unchanged.

Between Figures 6b and 6c, the number of variables in the support of the true response increases from 5 to 10. The MLGL package still provides among the best results. But more selected groups turn out to be false positives when reaching the maximal number of true positives.

When the size of the diagonal-blocks is decreased from 10 to 5 between Figures 6b and 6d, all procedures perform similarly (even if the correlation is set at 0.9). It seems that dealing with large blocks with highly correlated variables is a difficult settings for *HCAR* and *CGL*.

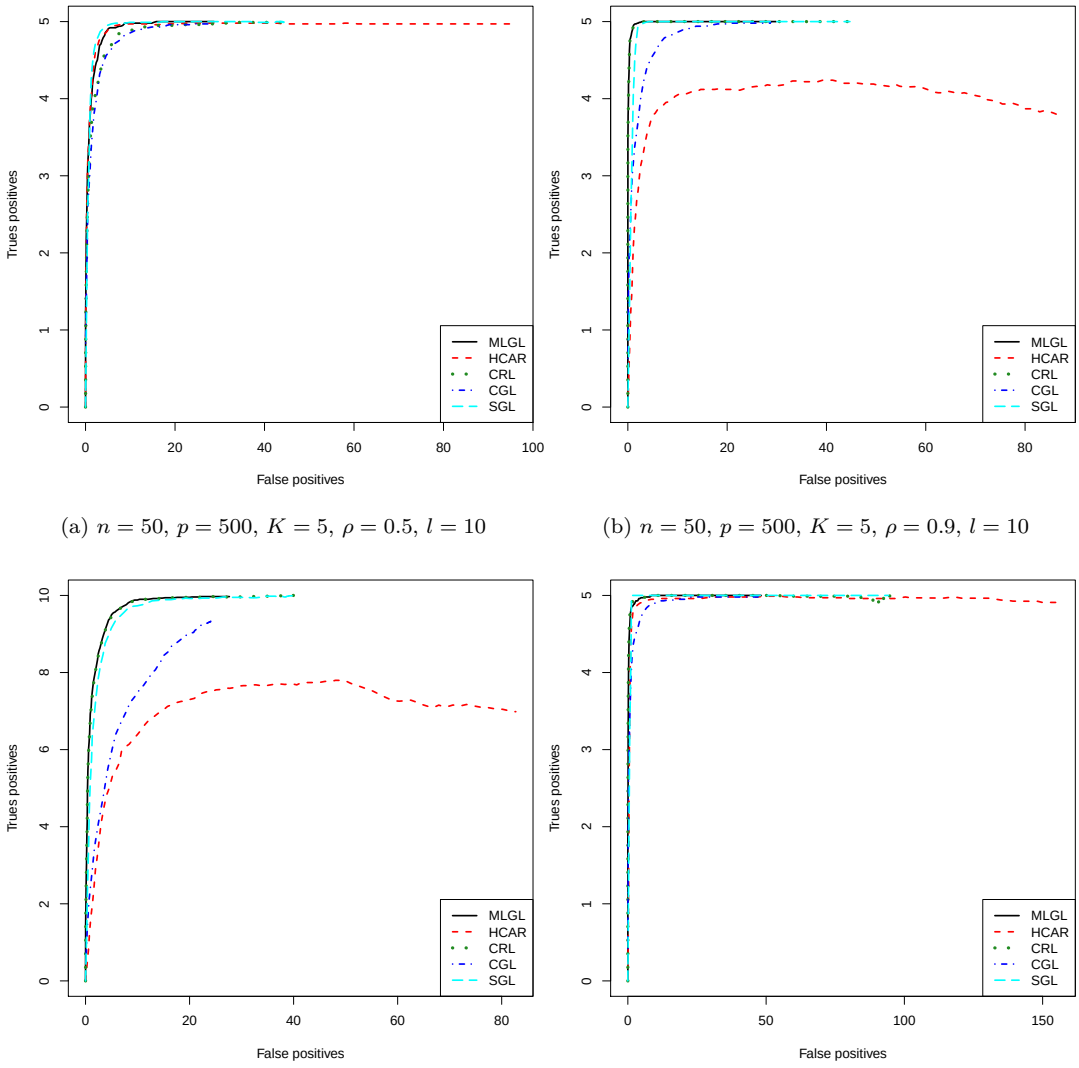
The procedure implemented in the MLGL package seems to have better results when the size of blocks is increased and the correlation strength is greater, which has the effect of reducing the effective dimension of the problem.

4.3 Hierarchical multiple testing procedure

Let us now assess the quality of the solution path before and after applying the HMT procedure. Figure 7 shows the number of true and false positives among the groups output by AHC+gLasso before and after applying the HMT procedure.

One striking aspect of these experimental results is that the set of groups output by AHC+gLasso contains more false than true positives for small values of λ. But the two curves quickly cross each other as λ grows. This strengthens the need for a multiple testing procedure discarding false groups. It is also noticeable that the number of false positives immediately drops after using the HMT procedure, no matter the level α at which the multiple testing correction is applied.

With only $K = 5$ true groups, most of the true positives are kept after applying HMT, unlike what happens when the number of true groups is $K = 10$ (Figure 7c). However in presence of highly correlated variables (within groups), the performance of the MLGL package strongly



(a) $n = 50$, $p = 500$, $K = 5$, $\rho = 0.5$, $l = 10$

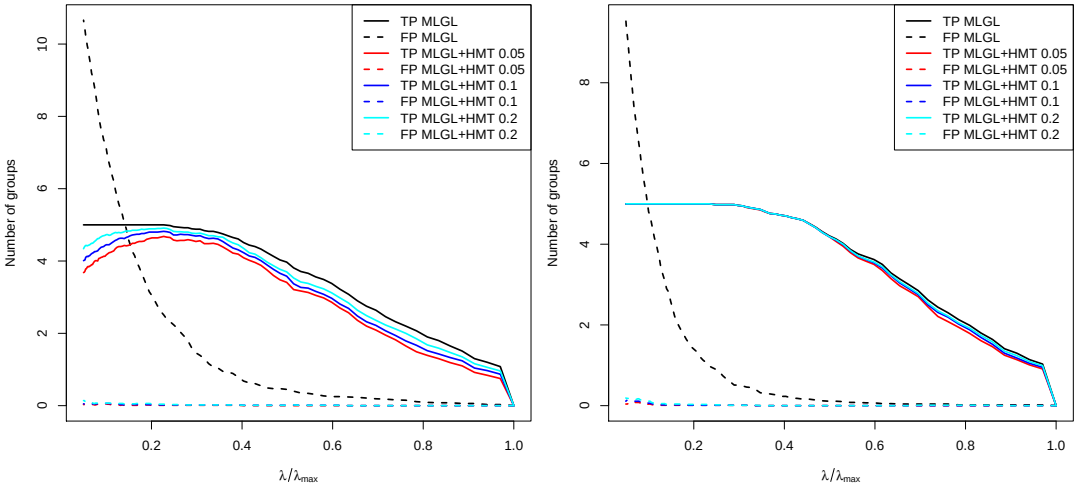
(b) $n = 50$, $p = 500$, $K = 5$, $\rho = 0.9$, $l = 10$

(c) $n = 50$, $p = 500$, $K = 10$, $\rho = 0.9$, $l = 10$

(d) $n = 50$, $p = 500$, $K = 5$, $\rho = 0.9$, $l = 5$

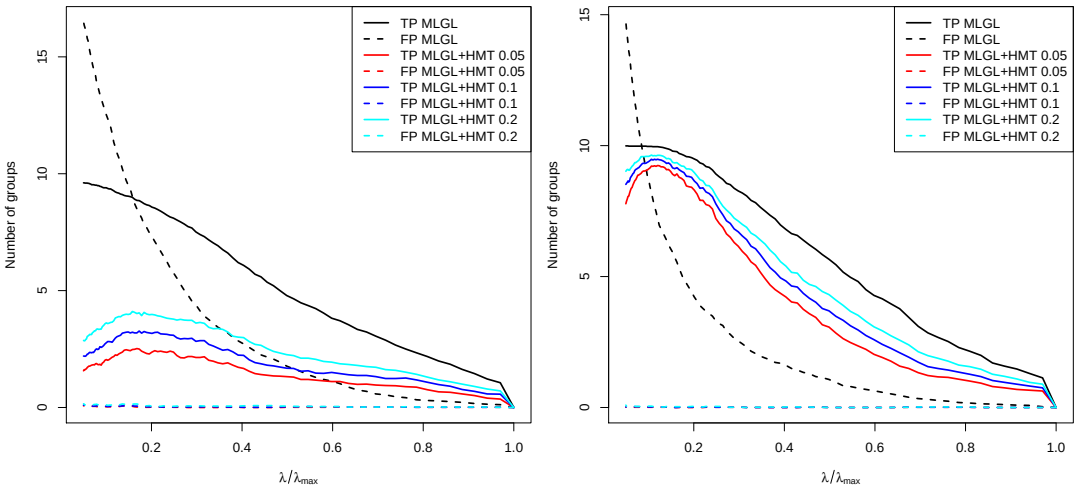
Figure 6: Number of true positives versus the number of false positives along the solution path of *Multi-Layer Group-Lasso* before hierarchical multiple testing (MLGL, black), *Hierarchical Clustering and Averaging for Regression* (HCAR, red), *Cluster Representative Lasso* (CRL, green), Cluster Group-Lasso (CGL, blue) and *Supervised Group-Lasso* (SGL, cyan). Each curve represents the average of 100 trials. Between the Figure 6a and 6b, the correlation ρ rises from 0.5 to 0.9. Between the Figures 6b and 6c, the number of true groups K rises from 5 to 10. Between the Figures 6b and 6d, the size l of blocks reduces from 10 to 5.

improves (Figure 7d) since on average, more than 9 (out of 10) true positives can be recovered at best. By contrast when the correlation decreases, the performance sharply drops (Figure 7c). In this situation, the maximum number of true positives is rather small (only 4 out of 10 when $\alpha = 0.20$).



(a) $n = 50$, $p = 500$, $K = 5$, $\rho = 0.7$, $l = 10$

(b) $n = 50$, $p = 500$, $K = 5$, $\rho = 0.9$, $l = 10$



(c) $n = 50$, $p = 500$, $K = 10$, $\rho = 0.7$, $l = 10$

(d) $n = 50$, $p = 500$, $K = 10$, $\rho = 0.9$, $l = 5$

Figure 7: Number of true and false positives along the solution path of *Multi-Layer Group-Lasso* before (MLGL, black) and after applying the hierarchical multiple testing procedure (MLGL + HMT) with $\alpha \in \{0.05, 0.1, 0.2\}$. In these figures, *MLGL* stands for *ACH + gLasso*. Each curve represents the average of 100 trials. The upper figures show the case $K = 5$ whereas the bottom figures show the case $K = 10$. From left to right, the correlation increases from 0.7 to 0.9.

Table 1: Number of true (TP) and false positives (FP) for different values of regularization parameters for $n = 100$ and $p = 500$. $\hat{\lambda}_{\mathrm{RM}}$ (resp. $\hat{\lambda}_{\mathrm{TPM}}$) denotes the value maximizing the number of rejections (resp. true positives). K, l et ρ are the different parameters of the simulated data. K is the size of the support of β^*, l the size of blocks and ρ the within-block correlation. In the HMT procedure, $\alpha = 0.05$.

		$K = 5$				$K = 10$			
		$l = 5$		$l = 10$		$l = 5$		$l = 10$	
		TP	FP	TP	FP	TP	FP	TP	FP
$\rho = 0.9$	ACH + gLasso + HMT + $\hat{\lambda}_{\mathrm{RM}}$	4.91	0.28	4.78	0.8	4.15	0.44	4.75	1.28
	ACH + gLasso + HMT + $\hat{\lambda}_{\mathrm{TPM}}$	4.95	0	4.86	0.05	4.27	0.14	4.86	0.57
$\rho = 0.7$	ACH + gLasso + HMT + $\hat{\lambda}_{\mathrm{RM}}$	2.95	0.51	3.95	0.27	1.59	0.64	3.39	0.54
	ACH + gLasso + HMT + $\hat{\lambda}_{\mathrm{TPM}}$	3.02	0.13	3.97	0.14	1.65	0.23	3.4	0.39
$\rho = 0.5$	ACH + gLasso + HMT + $\hat{\lambda}_{\mathrm{RM}}$	2.89	0.32	2.6	0.53	1.68	0.41	1.53	0.63
	ACH + gLasso + HMT + $\hat{\lambda}_{\mathrm{TPM}}$	2.95	0.04	2.7	0.13	1.72	0.13	1.58	0.26

From the different pictures of Figure 7, the overall conclusion owing to the calibration of λ is that choosing the value of λ maximizing the number fo rejections provides the best results in terms of the ratio between true and false positives. This clearly arises from the remark that the number of false positives is almost constant in our experimental results compared to the strong variations in the true positives curve. However this should be clear that this is likely to be a by-product of the high conservativeness of the HMT procedure implemented in the MLGL package.

4.4 Tuning the parameter λ

Let us now illustrate the performance of the procedure implemented in the MLGL package which yields the final selected groups.

Maximizing the number of rejections. Based on the previous remarks made in Section 4.3, the default value of λ recommended in the MLGL package is the one maximizing the number of rejections, which is denoted by $\hat{\lambda}_{\mathrm{RM}}$) in what follows.

However it should be clear that the number of rejections can include some false positives, which would be suboptimal. Therefore, an *oracle* choice for the parameter λ is the one maximizing the number of true rejections, called $\hat{\lambda}_{\mathrm{TPM}}$. Since the number of false positives in our simulation experiments only slowly increases, this choice should provide the best possible performance in terms of the ratio between true and false positives. All of this is illustrated by Table 1, which collects the results obtained with $\alpha = 0.05$. From Table 1, the main idea is that choosing $\lambda = \hat{\lambda}_{\mathrm{RM}}$ as the value maximizing the number of rejections is almost optimal since, whatever the experimental conditions, both the numbers of true and false rejections remain close to the ones of the oracle rule $\hat{\lambda}_{\mathrm{TPM}}$.

There is a drop of the number of true positives (both for $\hat{\lambda}_{\mathrm{RM}}$ and $\hat{\lambda}_{\mathrm{TPM}}$) as the number K of true groups increases from 5 to 10. This phenomenon is somewhat balanced by the increase of the correlation level (at least when $\rho = 0.9$) since in this case, we keep almost the same results.

Another interesting idea is that increasing the size l of the blocks in presence of a strong

Table 2: Comparison of different methods of choice of the regularization parameter. Stability selection is used with a threshold of 0.75. TP and FP correspond to true positives and false positives. K, l et ρ are the different parameters of the simulated data. K is the size of the support of β^*, l the size of blocks and ρ the within-block correlation.

		$K=5$				$K=10$			
		$l=5$		$l=10$		$l=5$		$l=10$	
		TP	FP	TP	FP	TP	FP	TP	FP
$\rho=0.9$	proposed method	4.91	0.28	4.78	0.8	4.15	0.44	4.75	1.28
	Kappa	3.66	2.14	4.3	2.64	5.78	13.25	5.84	8.06
	5-f cv	4.96	24.37	4.87	23.4	8.15	30.46	7.74	27.21
	stability	4.99	0.15	5	0.4	7.4	0.22	9.92	0.22
$\rho=0.7$	proposed method	2.95	0.51	3.95	0.27	1.59	0.64	3.39	0.54
	Kappa	2.59	1.68	3.86	1.21	2.76	4.36	6.22	3.29
	5-f cv	3.73	6.32	4.36	5.33	3.35	6.35	6.46	4.55
	stability	4.52	0.61	5	1.79	3.63	0.84	9.8	1.58
$\rho=0.5$	proposed method	2.89	0.32	2.6	0.53	1.68	0.41	1.53	0.63
	Kappa	3.19	3.83	3.08	3.32	2.93	5.50	3.56	5.34
	5-f cv	3.55	6.73	3.49	6.28	3.34	7.67	3.72	5.71
	stability	3.17	1.17	4.85	1.61	2.54	1.79	8.01	1.51

enough correlation level improves the results. For instance if $\rho = 0.5$, increasing l from 5 to 10 reduces the number of groups, but does not improve the results. By contrast, as long as $\rho \geq 0.7$, enlarging the blocks reduces the effective dimension of the problem, which leads to better results.

Performance of HMT+$\hat{\lambda}_{RM}$. An important question is to determine the influence of the procedure HMT+$\hat{\lambda}_{RM}$ on the quality of the final selected groups. To address this question, a comparison is carried out between the selection procedure of λ implemented in the MLGL package and alternative ones such as 5-fold cross-validation, kappa [Sun et al., 2013], and stability selection [Meinshausen and Bühlmann, 2010]. Let us emphasize that 5-fold cross-validation aims at selecting a $\hat{\lambda}$ which minimizes the prediction error, whereas Kappa and stability selection mainly focus on selecting groups with the highest possible stability. However all these procedures are time-consuming since they require multiple executions of the whole procedure. Table 2 collects the experimental results.

Firstly, 5-fold cross-validation uniformly selects more true positives, but at the price of including by far more false positives than any other competitor. This is in line with the trend of cross-validation to favor estimation/prediction rather than identification/selection.

Secondly, the best overall performance is achieved by the stability selection which always provides the largest number of true positives and only a small (averaged) number of false positives. This remarkable conclusion has to be balanced with the higher computational cost suffered by this time-consuming procedure.

Thirdly, the procedure implemented in the MLGL package yields close (but somewhat smaller) numbers of true positives compared to stability selection. However, the number of false positives is almost equal to (or lower than) the one of stability selection, which results from the conservativeness of our HMT procedure.

Finally the Kappa selection procedure performance stays in between that of 5-fold cross-

validation and the one of the MLGL package, for a higher computational price.

In conclusion, choosing the regularization parameter as the one maximizing the number of rejections gives reliable results which remain close to optimal ones according to our simulation experiments. The procedure implemented in the MLGL package seems conservative. But it does not require any intensive re-sampling and selects only a few false positives.

5 Conclusions

We designed a selection procedure implemented in the MLGL package, MLGL standing for *Multi-Layer Group-Lasso*. This procedure aims at selecting groups of correlated variables according to a response variable. It combines hierarchical clustering and group-Lasso. It differs from classical group-Lasso-based strategies by allowing to use simultaneously different levels of the hierarchy provided by the hierarchical clustering step. A weight for each level of the hierarchy is introduced to favor a priori "good" levels (according to a quality measure). From our empirical experiments, it results that the MLGL package performs almost the same as or improves upon alternative procedures combining hierarchical clustering and group-Lasso.

Possible improvements of the procedure in the MLGL package could be made, for instance by optimizing the weight function used at the group-Lasso step. Developing a more flexible weight function or using the results of several hierarchical clustering distances are interesting lines of research to explore.

In the MLGL package, the optimal value of the regularization parameter is chosen by maximizing the number of rejections. This results from the conservativeness of the involved HMT procedure. This HMT procedure has nevertheless the merit of taking into account the possible hierarchical trees and provides a FWER control of the selected groups. A way to improve the results is to provide tighter bounds on the FWER control to get a refined p-value correction. Nevertheless, the main advantage of the MLGL package over alternative approaches is that it provides close to optimal results while requiring a by far smaller computation time.

Acknowledgements

We thank Direction Générale de l'Armement (DGA) and Inria for a financial support of Quentin Grimonprez's PhD, and the CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015-2020.

References

[Akaike, 1974] Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

[Arlot and Celisse, 2010] Arlot, S. and Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79.

[Barber et al., 2015] Barber, R. F., Candès, E. J., et al. (2015). Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43(5):2055–2085.

[Benjamini and Hochberg, 1995] Benjamini, Y. and Hochberg, Y. (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300.

[Bühlmann et al., 2013] Bühlmann, P., Rütimann, P., van de Geer, S., and Zhang, C.-H. (2013). Correlated variables in regression: clustering and sparse estimation. *Journal of Statistical Planning and Inference*, (143):1835–3871.

[Bühlmann and van de Geer, 2011] Bühlmann, P. and van de Geer, S. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer Publishing Company, Incorporated.

[Dunn, 1959] Dunn, O. J. (1959). Estimation of the medians for dependent variables. *Ann. Math. Statist.*, 30(1):192–197.

[Fan et al., 2012] Fan, J., Guo, S., and Hao, N. (2012). Variance estimation using refitted cross-validation in ultrahigh dimensional regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(1):37–65.

[Fan and Tang, 2013] Fan, Y. and Tang, C. Y. (2013). Tuning parameter selection in high dimensional penalized likelihood. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(3):531–552.

[Giraud et al., 2007] Giraud, C., Baraud, Y., and Huet, S. (2007). Gaussian Model Selection with Unknown Variance.

[Jacob et al., 2009] Jacob, L., Obozinski, G., and Vert, J.-P. (2009). Group Lasso with Overlap and Graph Lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 433–440, New York, NY, USA. ACM.

[Jain et al., 1999] Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data Clustering: A Review. *ACM Comput. Surv.*, 31(3):264–323.

[Jamshidian et al., 2007] Jamshidian, M., Jennrich, R. I., and Liu, W. (2007). A study of partial f tests for multiple linear regression models. *Comput. Stat. Data Anal.*, 51(12):6269–6284.

[Jenatton et al., 2011] Jenatton, R., Audibert, J.-Y., and Bach, F. (2011). Structured variable selection with sparsity-inducing norms. *J. Mach. Learn. Res.*, 12:2777–2824.

[Liu and Zhang, 2009] Liu, H. and Zhang, J. (2009). Estimation consistency of the group lasso and its applications. In *JMLR*.

[Ma et al., 2007] Ma, S., Song, X., and Huang, J. (2007). Supervised group lasso with applications to microarray data analysis. *BMC Bioinformatics*, 8(1):60.

[Meinshausen, 2008] Meinshausen, N. (2008). Hierarchical testing of variable importance. *Biometrika*, 95(2):265–278.

[Meinshausen and Bühlmann, 2010] Meinshausen, N. and Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473.

[Park et al., 2007] Park, M. Y., Hastie, T., and Tibshirani, R. (2007). Averaged gene expressions for regression. *Biostatistics*, 8(2):212–227.

[Schwarz, 1978] Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

[Simon and Tibshirani, 2011] Simon, N. and Tibshirani, R. (2011). Standardization and the group lasso penalty. Technical report.

[Sun et al., 2013] Sun, W., Wang, J., and Fang, Y. (2013). Consistent Selection of Tuning Parameters via Variable Selection Stability. 14:3419–3440.

[Tibshirani, 1994] Tibshirani, R. (1994). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288.

[Tibshirani et al., 2005] Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B*, pages 91–108.

[Wainwright, 2009] Wainwright, M. J. (2009). Sharp Thresholds for High-dimensional and Noisy Sparsity Recovery Using L1-constrained Quadratic Programming (Lasso). *IEEE Trans. Inf. Theor.*, 55(5):2183–2202.

[Wasserman and Roeder, 2009] Wasserman, L. and Roeder, K. (2009). High-dimensional variable selection. *Ann. Statist.*, 37(5A):2178–2201.

[Witten et al., 2014] Witten, D. M., Shojaie, A., and Zhang, F. (2014). The cluster elastic net for high-dimensional regression with unknown variable grouping. *Technometrics*, 56(1):112–122.

[Yang and Zou, 2015] Yang, Y. and Zou, H. (2015). A fast unified algorithm for solving group-lasso penalized learning problems. *Statistics and Computing*, 25(6):1129–1141.

[Yuan and Lin, 2006] Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67.

[Zhao and Yu, 2006] Zhao, P. and Yu, B. (2006). On Model Selection Consistency of Lasso. *J. Mach. Learn. Res.*, 7:2541–2563.

Appendix

A Proof of Lemma 1

Let β denote a solution of the group-Lasso (equation (6)) for a value of λ, then β must check $\forall i = 1, \ldots, g$:

$$X_{G_i}^T(y - X\beta) = \lambda w_i s_{G_i} \tag{10}$$

with s_{G_i} belonging to subdifferential of the function $\|\,.\,\|_2$ at θ_{G_i},

$$s_{G_i} \in \begin{cases} \left\{\frac{\beta_{G_i}}{\|\beta_{G_i}\|_2}\right\} & \text{if } \beta_{G_i} \neq 0_{|G_i|} \\ \left\{z \in \mathbb{R}^{|G_i|} \;\middle|\; \|z\|_2 \leq 1\right\} & \text{if } \beta_{G_i} = 0_{|G_i|} \end{cases}$$

The subdifferential of a function $f : U \to \mathbb{R}$ with U a convex subset of $\mathbb{R}^p$ contains the subgradients of f. A vector $v \in U$ is a subgradient of f at x_0 if $\forall x \in U : f(x) - f(x_0) \geq \langle v, x - x_0 \rangle$.

From Karush-Kuhn-Tucker (KKT) conditions, we can deduce that if $\|X_{G_i}^T(y - X\theta)\|_2 < \lambda w_i$ then $\theta_{G_i} = 0_{|G_i|}$.

Proof 1 (Lemma 1)**.** *Suppose that $G_1 = G_2$ and $w_2 > w_1 > 0$. Let θ denote a solution of group-Lasso (equation (6)). We want to show that we have $\theta_{G_2} = 0_{|G_2|}$.*

- *Let $\theta_{G_1} = 0_{|G_1|}$. We show that $\theta_{G_2} = 0_{|G_2|}$.*
  If $\theta_{G_1} = 0_{|G_1|}$, from KKT conditions, we have:

$$\begin{aligned} &\|X_{G_1}^T(y - X\theta)\|_2 \leq \lambda w_1 \\ &\|X_{G_2}^T(y - X\theta)\|_2 \leq \lambda w_1 \text{ because } X_{G_1} = X_{G_2} \\ &\|X_{G_2}^T(y - X\theta)\|_2 < \lambda w_2 \text{ because } w_1 < w_2 \end{aligned}$$

  So, $\theta_{G_2} = 0_{|G_2|}$.

- *If $\theta_{G_1} \neq 0_{|G_1|}$. We show that $\theta_{G_2} = 0_{|G_2|}$.*
  If $\theta_{G_1} \neq 0_{|G_1|}$, from KKT conditions, we have:

$$\begin{aligned} X_{G_1}^T(y - X\theta) &= \lambda w_1 \frac{\theta_{G_1}}{\|\theta_{G_1}\|_2} \\ \|X_{G_1}^T(y - X\theta)\|_2 &= \left\|\lambda w_1 \frac{\theta_{G_1}}{\|\theta_{G_1}\|_2}\right\|_2 \\ \|X_{G_1}^T(y - X\theta)\|_2 &= \lambda w_1 \\ \|X_{G_2}^T(y - X\theta)\|_2 &= \lambda w_1 \text{ because } X_{G_1} = X_{G_2} \\ \|X_{G_2}^T(y - X\theta)\|_2 &< \lambda w_2 \text{ because } w_1 < w_2 \end{aligned}$$

  So, $\theta_{G_2} = 0_{|G_2|}$.
  We have shown that $\theta_{G_2} = 0_{|G_2|}$, the lemma is proved.

Chapter 4

Perspectives

Mes travaux de recherche actuels et ceux des années à venir sont dans la continuité des développements méthodologiques présentés dans les chapitres précédents. Ayant acquis une expertise sur l'analyse statistique de différents niveaux -omiques, il paraissait naturel de participer aux réflexions de la communauté scientifique sur l'intégration de données -omiques et cliniques. Ceci représente un domaine très vaste et je présenterai simplement deux axes de recherche auxquels je m'intéresse, à savoir:

- l'influence de la taille d'effet et du rapport nombre d'individus/nombre de variables dans l'intégration de données -omiques et cliniques.
- l'intégration de données -omiques provenant de différentes technologies à haut débit.

Par ailleurs, l'intégration de données englobant aussi les approches d'intégration de résultats issus d'analyses séparées de différents niveaux -omiques, il est important de continuer à contribuer aux analyses ne concernant qu'un seul niveau -omique. Mon troisième axe de recherche actuel est:

- la prise en compte d'une structure temporelle dans l'analyse statistique de données d'expériences à haut débit.

Ces trois axes de recherche sont présentés dans les sections suivantes.

4.1 Influence de la taille d'effet et du rapport nombre d'individus/nombre de variables dans l'intégration de données -omiques et cliniques

Avec la réduction du coût des expériences à haut débit, il est maintenant classique, notamment en médecine de précision, de vouloir intégrer des variables -omiques dans une construction de score de recherche clinique. L'un des enjeux de cette construction de scores est de développer des méthodes permettant de sélectionner un petit nombre de variables pertinentes (principe de parcimonie) dans un contexte de données hétérogènes (données -omiques et cliniques). L'intégration de toutes les données dans le même modèle statistique n'est pas facile : les tailles d'effet et le rapport nombre d'individus/nombre de variables

sont très différents selon le type de données. Le nombre de sujets inclus dans les expériences à haut débit est souvent guidé par la faisabilité technique ou un calcul de coût, sachant que ces expériences sont exploratoires. Le nombre de sujets inclus est inférieur au nombre de sujets nécessaires pour une étude de validation de biomarqueurs. Les variables cliniques ont souvent été pré-sélectionnées en fonction du contexte de l'étude et présentent généralement des effets plus forts que ceux des variables -omiques. La plupart des approches statistiques utilisées en clinique reposent sur l'hypothèse qu'il y a plus d'individus que de variables, tandis qu'en analyse de données -omiques, il est courant d'avoir beaucoup plus de variables que d'individus. Plusieurs approches ont été proposées dans la littérature pour intégrer des données -omiques et cliniques. Une partie du travail de thèse d'Hélène Sarter vise à comparer différentes approches sur des données simulées et sur des données réelles pour une réponse binaire. Dans les prochaines années, je prévois aussi de poursuivre ce travail de comparaison pour un critère de jugement censuré ou répété. Les résultats attendus sont de produire des règles de bonnes pratiques pour l'intégration de données cliniques et -omiques, en particulier sur le nombre de variables à inclure pour chaque type de données et le choix de la méthode appropriée.

4.2 Intégration de données -omiques provenant de technologies à haut débit différentes

L'utilisation de technologies à haut débit différentes (par exemple puces à ADN, séquençage à haut débit) peut nécessiter des traitements statistiques particuliers plus compliqués que l'inclusion d'un simple effet technologique dans le modèle statistique. En effet, le choix de la loi statistique utilisée pour modéliser les données à haut débit est crucial lorsque le nombre d'individus statistiques est faible, ce qui est le cas dans les analyses exploratoires de recherche de biomarqueurs potentiels. En particulier, il est commun d'utiliser des lois de Poisson ou négatives binomiales pour les comptages issus du séquençage à haut débit et des lois normales pour les intensités issues des puces à ADN. Les méthodes initialement développées pour l'analyse de données de puces à ADN ne sont pas directement applicables pour les données de séquençage. Plusieurs transformations co-existent et ont déjà été étudiées pour la classification non supervisée de données de séquençage. Cependant, il n'existe pas de critère statistique de sélection de modèle permettant de s'affranchir de l'effet technologie pour regrouper des variables provenant de jeux de données issus à la fois de puces à ADN et de séquençage à haut débit. Un objectif à moyen terme du travail en cours consiste à proposer un critère qui permette de sélectionner simultanément la meilleure transformation possible pour chacun des jeux intégrés et la classification résultant de ces transformations.

4.3 Prise en compte d'une structure temporelle dans l'analyse statistique de données d'expérience à haut débit

Dans le cadre étudié, on suppose qu'il n'y a pas suffisamment de mesures pour utiliser les techniques d'analyse fonctionnelle (où les individus sont des courbes) mais suffisamment pour souhaiter prendre en compte la structure au cours du temps. Habituellement, lorsque des données -omiques sont mesurées au cours du temps sur les mêmes individus, la recherche de biomarqueurs se fait souvent pour chaque instant puis, si plusieurs temps sont présents, les biomarqueurs sélectionnés pour les différents instants sont comparés. En raison du nombre de variables présentes dans ces expériences à haut débit, il est fréquent d'observer des problèmes de corrélation qui induisent des sélections différentes à chaque instant. Cela nécessite un post-traitement de l'union des marqueurs sélectionnés aux différents instants pour observer leurs profils et leurs corrélations. L'un des objectifs de la thèse de Wilfried Heyse est de développer une nouvelle méthode statistique de sélection de variables dans un contexte prédictif où les mesures sont répétées dans le temps sur des milliers de variables simultanément. Il s'agit d'adapter la méthodologie existante en sélection de variables à d'autres méthodes d'apprentissage, qui prennent en compte la structure temporelle. Les modèles linéaires mixtes pourront être particulièrement appropriés pour ces mesures répétées, sous réserve d'une adaptation à un contexte dit de grande dimension (nombre de variables largement supérieur au nombre d'individus). En parallèle, une classification multivariée non supervisée pourra être utilisée pour modéliser la structure temporelle. Cette structure pourrait guider le choix de la matrice de "design" du modèle de régression et la conception d'une pénalité appropriée. Le gain concernant l'interprétabilité des variables sélectionnées par cette nouvelle approche sera comparé aux résultats des analyses temps à temps.

Ces trois axes majeurs de recherche sont nés de questions récurrentes que des biologistes m'ont posée, notamment dans le cadre de mon encadrement scientifique pour des projets de la plateforme bilille. Le chapitre suivant présente la genèse de cette plateforme et quelques travaux d'ingénierie que j'ai pu encadrer.

Chapter 5

Activités de support à la recherche en biologie-santé: la plateforme bilille

5.1 Présentation

Bilille est une plateforme de bioinformatique, qui travaille en proximité avec les unités de recherche en biologie-santé de la métropole lilloise. Ses missions couvrent l'accompagnement de projets (plus de 20 par an), la formation, la mise à disposition de moyens de calcul, le développement de nouveaux outils logiciels, bases de données et chaînes de traitement, l'animation scientifique et technologique. Ses domaines d'activité vont de l'analyse de données -omiques à l'annotation, la phylogénie, la bioinformatique structurale, la biologie intégrative. Pour répondre aux besoins des utilisateurs, trois profils principaux interagissent dans bilille: des biologistes ayant un master de bioinformatique, des biostatisticiens ayant un cursus initial en mathématiques ou en physique et des informaticiens ayant un master de bioinformatique ou d'intelligence artificielle.

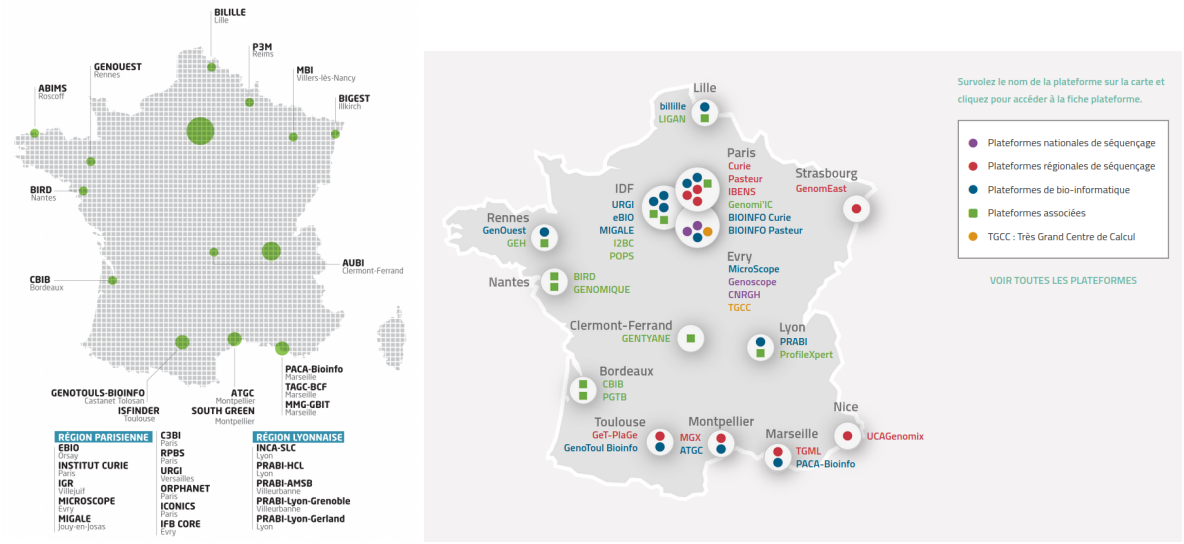
D'un point de vue institutionnel, la plateforme bilille a rejoint au 1er janvier 2020 l'UMS 2014 - US 41 PLBS (Plateformes lilloises en biologie santé), dont les tutelles sont l'Université de Lille, le CNRS, l'Inserm, l'Institut Pasteur de Lille et le CHU de Lille. Cette UMS - US regroupe des plateformes de génomique, d'imagerie cellulaire et d'imagerie du vivant, de criblage, de glycomique et protéomique et de ressources expérimentales.

La plateforme bilille a trois implémentations géographiques. Ayant des bureaux depuis 2016 à la fois sur les Campus Cité Scientifique et Campus Hospitalo-Universitaire, elle a aussi un bureau sur le Campus Pasteur depuis 2018. La plateforme a connu une croissance très forte en 4 ans, passant de 2 CDD temps plein à 8 CDD aujourd'hui.

5.2 La naissance de bilille

La plateforme bilille a pris son nom lors de la naissance de l'Institut Français de Bioinformatique et de France Génomique, deux infrastructures nationales qui ont vu le jour grâce aux financements "Investissement d'Avenir" dans le cadre du projet "Infrastructures nationales en Biologie et Santé". Bilille apparaissait alors sur les cartes de ces deux infrastructures en tant que plateforme membre (cf. Figure 5.1), c'est-à-dire ceux qu'elle a pu bénéficier de financements directs de ces infrastructures pour répondre à des besoins précis

Figure 5.1: Cartes des plateformes membres de l'Institut Français de Bioinformatique (à gauche) et de France Génomique (à droite) en 2016



Sources: france-bioinformatique.fr et france-genomique.org

de la communauté nationale, où les compétences locales apportaient une valeur ajoutée au niveau national.

Localement, bilille bénéficiait du plan pluri-formation (PPF) bioinfo de l'Université Lille 1, permettant de financer 2 à 3 journées d'animation scientifique par an et le contrat d'un ingénieur répondant aux besoins du campus Cité Scientifique. Cet ingénieur bénéficiait de l'environnement de recherche en bioinformatique de l'équipe-projet Inria BONSAI, au sein du laboratoire CRIStAL de la faculté actuelle des Sciences et Technologies (ex-Lille 1).

Fin 2015, un comité institutionnel regroupant les tutelles actuelles de l'UMS-US et Inria invitait les plateformes des différents sites lillois à se regrouper. C'est à ce moment-là que Maude Pupin, alors responsable de bilille, démissionna de sa mission de responsable et qu'Hélène Touzet et moi-même décidâmes d'allier les forces des campus Hospitalo-Universitaire(HU) et Cité Scientifique pour donner une nouvelle forme à la plateforme bilille et devînmes co-responsables de bilille. Cette nouvelle forme devait intégrer les différents plateaux de bioinformatique lillois identifiés par le comité institutionnel, la plupart travaillant sur des plateformes de génomique ou offrant des services de proximité uniquement à l'unité de recherche qui les intégrait. Plusieurs permanents ont accepté de consacrer entre 5 et 20 % de leur temps à la nouvelle forme de bilille, ce qui a permis de maintenir une animation scientifique et technologique, relancer un réseau métier ingénieurs et développer une offre de formation. Quant à Hélène Touzet et moi-même, nous avons consacré entre 40 et 50% de notre temps pour développer bilille avec peu de moyens humains (seulement deux ingénieurs temps plein en CDD au départ) et répondre aussi bien que possible aux besoins des unités de recherche en biologie santé de la métropole lilloise. Notre investissement important porte ses fruits cette année, avec l'intégration de bilille à l'UMS-US qui permet l'ouverture de deux postes au concours cette année (l'un par l'Université de Lille,

l'autre par le CNRS), permettant de recruter un responsable opérationnel et pérenniser, nous l'espérons, au moins l'un des 8 CDD de la plateforme.

5.3 Comment développer bilille avec peu de moyens humains?

En 2016, seuls deux ingénieurs (un statisticien sur le campus HU et une bioinformaticienne sur le campus Cité scientifique) pouvaient être recrutés en CDD sur la plateforme pour répondre aux besoins des unités de recherche en biologie santé de la métropole lilloise. Afin de recenser les besoins et planifier le travail de ces deux ingénieurs, nous avons lancé un "appel à projets bilille" offrant du temps ingénieur pour des projets de petite dimension (missions de 1 à 2 mois): tests d'outils, analyse de données, petit développement. Les besoins étant très importants, nous n'avons malheureusement pas pu répondre à tous les besoins mais cela a permis d'initier des collaborations qui ont ensuite débouché sur des financements permettant de renforcer les moyens humains de la plateforme.

5.4 Quelques exemples de projets bilille pour illustrer la différence entre la théorie et la pratique ...

Ayant encadré personnellement 24 projets en 4 ans pour la plateforme bilille, les exemples donnés dans ce paragraphe ne seront pas exhaustifs et ne représenteront pas nécessairement la variété des projets adressés à bilille. J'ai simplement souhaité illustrer la difficulté de travailler sur des données réelles, l'importance du dialogue avec les biologistes, et du test de méthodes existantes avant tout nouveau développement biostatistique.

5.4.1 Analyse de données transcriptomiques

Une des questions très classiques en analyse de données transcriptomiques est de savoir quels sont les gènes ou transcrits différentiellement exprimés entre deux conditions (e.g. tissu sain/tissu tumoral), c'est-à-dire dont l'expression moyenne varie entre ces deux conditions (cf chapitre 2). Les plateformes de génomique qui génèrent ces données savent maintenant traiter ces questions standard, par exemple en utilisant limma [Ritchie et al., 2015] pour les puces à ADN, DESeq2 [Love et al., 2014] ou edgeR [Robinson et al., 2010] pour le RNA-Seq. Ainsi, les projets qui arrivent à bilille ne sont pas toujours les plus faciles à analyser. Dans certains cas, le biologiste a fait appel à un prestataire qui ne faisait pas l'analyse de données pour payer moins cher. On a ainsi vu des projets de séquençage dont le coût global incluant l'analyse bioinformatique était bien plus cher que celui où tout aurait été traité par une plateforme de génomique ayant en son sein des bioinformaticiens, à cause d'une mauvaise qualité de données. Parfois, les données ont déjà été analysées mais les effets biologiques recherchés sont si faibles qu'ils n'ont pas été détectés par les analyses classiques. Quand des effets relatifs à un plan d'expérience donné n'ont pas été pris en compte, bilille améliore parfois les résultats mais cela nécessite une ré-analyse complète. Bilille est aussi souvent sollicitée quand il s'agit d'intégrer des données de transcriptomique et de protéomique. Là encore, ce n'est pas la tâche la plus aisée, sachant que les résultats

trouvés sur les niveaux transcriptomiques ou protéomiques séparément sont parfois plus faciles à publier qu'une analyse intégrée. Ce sont cependant ces projets d'intégration de données qui m'ont apporté de belles questions méthodologiques et donné l'envie d'encadrer une thèse sur ce sujet pour pouvoir donner des recommandations pratiques.

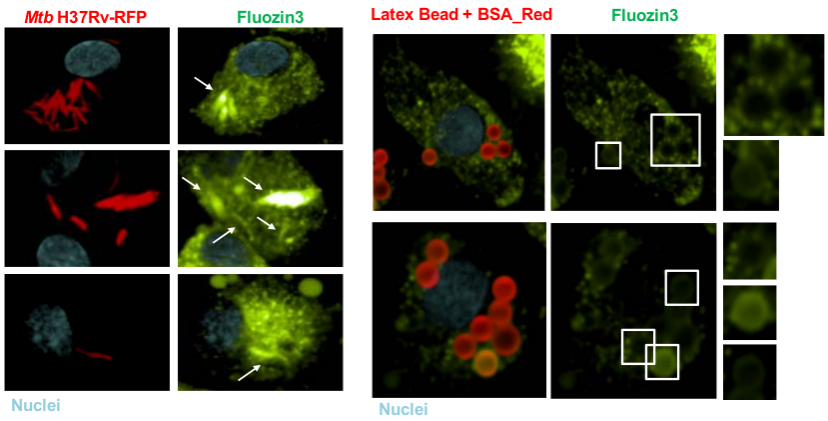
Plusieurs projets de transcriptomique déposés à bilille m'ont donné l'occasion de tester le package blockcluster, développé par des collègues de mon équipe-projet Inria MODAL. Ce package permet de faire la classification croisée à la fois de lignes et de colonnes, de façon complètement non supervisée. Sa publication originale [Bhatia et al., 2017] illustre brillamment son application sur des jeux de données réelles de segmentation d'image et de classification de documents. Il paraissait naturel d'utiliser ce package pour rechercher des sous-groupes de patients et sous groupes de gènes pour des études de co-expression. Un des premiers problèmes pratico-pratiques a été de savoir comment centrer réduire les données, avant l'utilisation du package. Fallait-il commencer par centrer-réduire les lignes ou les colonnes? Rappelons qu'en analyse différentielle, on évite de réduire par gène car on utilise souvent des approches bayésiennes empiriques pour modéliser la variance de ces gènes (cf chapitre 2). En classification classique, on centre et réduit les variables quand on cherche à classer les individus et on centre et réduit les individus quand on cherche à classer les variables. Les deux possibilités se justifiant dans la classification croisée, nous avons testé à chaque fois les deux approches, et les résultats étaient différents. Les visualisations fournies par blockcluster mettaient en évidence des sous-groupes homogènes relativement distincts. Malheureusement, quand nous croisions les variables cliniques à disposition avec les sous groupes de patients ainsi constitués, il était très difficile d'interpréter les groupes. Ces tests n'ont donc jamais donné lieu à publication dans des journaux de biologie. En revanche, ils ont généré une question de recherche méthodologique relative à la recherche de sous-labels dans une classification semi-supervisée en très grande dimension.

5.4.2 Criblage à haut contenu

Le criblage à haut-débit (High Throughput Screening - HTS) désigne les techniques visant à étudier et identifier des molécules aux propriétés nouvelles, parmi un grand nombre de composés pré-déterminés. Ces techniques sont très utilisées en médecine et pharmacologie pour créer de nouveaux médicaments. Pour accélérer la phase de test, le criblage peut utiliser la robotique ou être virtuel (in silico). Le criblage virtuel consiste à travailler avec des modèles mathématiques de molécules ou protéines, en créant toutes les combinaisons possibles, y compris des formes qui ne pourraient pas être synthétisées ou viables dans la nature. Les projets bilille pour lesquels j'ai encadré Franck Bonardi concernaient essentiellement des projets de criblage à haut contenu (High Content Screening - HCS). Le criblage à haut contenu, parfois connu aussi sous le nom de cellomique, est lié au criblage à haut débit dans la mesure où des milliers de composés sont testés en parallèle, mais implique des tests de phénotypes cellulaires plus complexes [Wikipedia contributors, 2020]. Les données fournies étant des images de microscopie par fluorescence, on parle d'imagerie cellulaire. L'imagerie est capable de détecter des changements à un niveau sub-cellulaire (cytoplasme, noyau, ...). Une revue du criblage phénotypique à haut contenu pour la chémobiologie et ses enjeux est disponible dans [Brodin et al., 2015]. L'originalité des expériences HCS réalisées sur l'Equipex Imaginex Biomed lillois est l'utilisation de billes

pour enfermer le pathogène, ce qui rend plus facile l'observation grâce à la forme circulaire, cf figure 5.2.

Figure 5.2: Utilisation de billes dans une expérience HCS



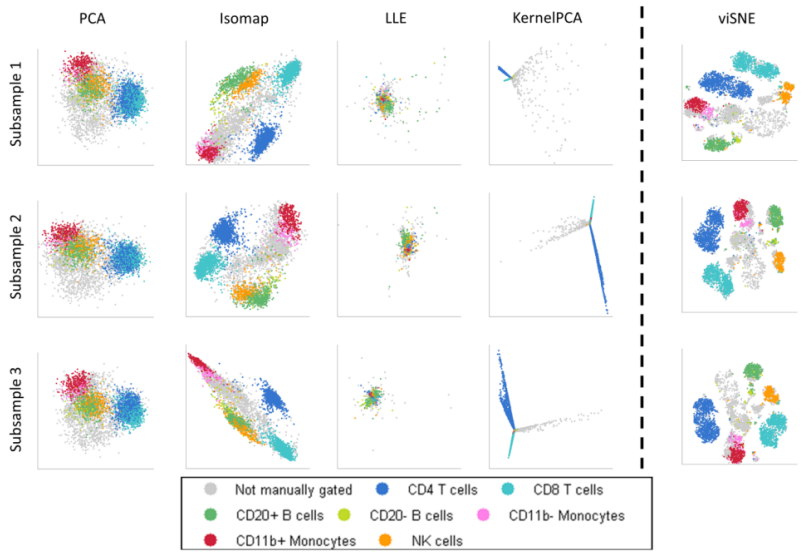
©P. Brodin

Le premier projet HCS confié à bilille était un projet de Priscille Brodin (CIIL: Center for Infection and Immunity of Lille) étudiant les mécanismes biologiques sous-jacents à la résistance aux traitements de mycobactéries tuberculeuses à l'intérieur des cellules. Le criblage étudiait une banque de 15291 petits ARN interférents (siRNA: small interfering RNA). Initialement, l'analyse paraissait très proche d'une analyse de puces à ADN où la fluorescence est aussi utilisée pour mesurer l'expression des gènes, chaque siRNA testé étant dans un puit. Cependant, les premières réunions ont montré des différences notables: il n'y avait pas de réplicats dans l'expérience et la question scientifique n'était pas une analyse différentielle entre deux groupes ou plus mais plutôt une recherche de "hits", siRNA qui se distinguaient des autres siRNA par des mesures atypiques. Il y avait plus de paramètres à prendre en compte que la simple fluorescence. La plus grosse difficulté a été la normalisation des données, et plusieurs packages R ont été envisagés: CellHTS2 [Boutros et al., 2006], sights [Garg et al., 2016] et HTSvis [Scheeder et al., 2017]. Cependant, devant l'absence de réplicats, le faible nombre de contrôles (qui en plus étaient positionnés sur les bords de chaque plaque et non aléatoirement), nous avons normalisé les données par une analyse de la variance incluant un effet donneur et un effet plaque et gardé les résidus pour l'analyse des hits. Une analyse en composantes principales a ensuite ciblé les paramètres d'imagerie qui différenciaient le plus les siRNA. Nous nous sommes ensuite concentrés sur ces paramètres pour rechercher des hits. Les résultats biologiques sont en cours d'analyse. Cet exemple illustre à quel point une méthode historique comme l'Analyse en Composantes Principales reste une alternative intéressante quand les conditions expérimentales ne permettent pas d'utiliser les derniers packages développés. Le paragraphe suivant donne un exemple d'expérience où cette même technique d'analyse en composantes principales n'est pas appropriée.

5.4.3 Cytométrie - quand les statisticiens ont à apprendre de la communauté *machine learning*

La cytométrie est une technique permettant de caractériser individuellement des cellules, en quantifiant plusieurs marqueurs sur les cellules, avec la possibilité de trier simultanément des sous-populations d'intérêt. L'article [Barlogie et al., 1983] décrit l'utilisation de la cytométrie en flux pour des applications diagnostiques en cancérologie. J'ai encadré Léa Fléchon et Marie Fourcot sur des projets déposés à la plateforme bilille, au moment où la plateforme de cytométrie de Lille souhaitait automatiser ses analyses, la cytométrie de masse permettant de mesurer plus de marqueurs que la cytométrie en flux. La thèse [Amir, 2014] illustre les défauts de l'ACP qui ne parvient pas à capter fidèlement les relations non linéaires présentes dans ce type de jeu de données. Cette thèse présente le développement d'un outil de visualisation (viSNE) adapté à l'algorithme t-Distributed Stochastic Neighbor Embedding (t-SNE) [Maaten and Hinton, 2008] issu de la communauté apprentissage automatique, qui rassemble à la fois des informaticiens, probabilistes et statisticiens. La troisième partie du deuxième chapitre de cette thèse présente une comparaison très intéressante de méthodes de réduction de dimension sur un jeu de données réelles. La figure 5.3 est issue de ce chapitre.

Figure 5.3: Comparaison de méthodes de réduction de dimension pour identifier des sous-populations d'intérêt



©El-ad David Amir

On remarque sur cette figure que l'ACP à noyaux, qui aurait pu capter des relations non linéaires, donne de plus mauvais résultats que l'ACP, regroupant les cellules en trois à cinq diagonales. L'algorithme t-SNE, en revanche, est particulièrement performant. La visualisation avec viSNE en a fait l'outil indispensable pour l'analyse de données de cytométrie.

5.4.4 Stratification de patients - classification supervisée ou non supervisée?

La question de départ paraissait classique. A partir de données cliniques d'une cohorte de plus de 1400 patients, l'équipe de F. Pattou (UMR1190) souhaitait retrouver des variables prédictives de la forme la plus sévère de la maladie du foie gras non alcoolique (NASH). Estelle Chatelain, ingénieure bilille, a alors mis en œuvre plusieurs techniques de classification supervisée, sous l'encadrement de Philippe Preux (Equipe-projet Inria Sequel), ainsi que des méthodes que nous utilisions classiquement dans bilille telles que les régressions logistiques pénalisées par lasso. Malheureusement, même si certaines variables étaient régulièrement sélectionnées par les forêts aléatoires et régressions pénalisées et semblaient donc pertinentes, les taux d'erreurs étaient relativement élevés après construction des scores. Comme le pourcentage de NASH était très faible (environ 10%), Estelle a ensuite rééquilibré les groupes de malades et non malades, sans observer d'amélioration notable. Plusieurs individus non NASH étaient très proches des NASH. C'est alors que nous avons décidé d'utiliser des approches de classification non supervisée pour définir des sous-groupes dans cette cohorte et voir si certains sous-groupes contenaient plus de NASH que d'autres. Cette stratégie s'est avérée très pertinente puisque cela permet de redéfinir une classification clinique regroupant les NASH et d'autres patients très proches n'ayant pas été qualifiés NASH mais présentant les mêmes complications cliniques. Dans cette étude, nous avons pu discuter de la difficulté pour les médecins de mettre certains labels, ce qui rend ensuite l'utilisation de méthodes supervisées plus compliquées. Cela redonne aussi de la place aux méthodes de classification non supervisées, y compris pour des études où l'objectif final sera prédictif.

5.5 Perspectives

Bilille a signé la lettre d'engagement des plateformes membres de l'Institut Français de Bioinformatique, c'est à dire que bilille s'est engagée à développer des services ouverts à l'extérieur, une tarification, une démarche qualité, une participation périodique aux analyses de coût complet. Le recrutement de deux nouveaux ingénieurs de recherche arrive donc à point nommé pour participer à ces démarches. Du côté scientifique, cette labellisation de l'Institut Français de Bioinformatique permet aussi de participer à son projet déposé Equipex+ MuDIS4LS, notamment au travers de travaux autour de la biologie intégrative.

Par ailleurs, bilille a rejoint l'UMS2014-US41 au 1er janvier 2020. Bilille avait déjà des liens très forts avec les plateformes de génomique de cette unité de service et s'est engagée à développer des liens avec les autres plateformes, notamment celles d'imagerie cellulaire et d'imagerie du vivant. Cela apportera d'autres beaux projets scientifiques, où il faudra tester les méthodes historiques, celles développées pour telle ou telle technologie, voire même apporter de nouveaux sujets de recherche en statistique.

Conclusion

Mes travaux de recherche s'inscrivent au sein de l'axe "évaluation clinique" de l'Unité Labellisée de Recherche (ULR) 2694 METRICS "Evaluation des technologies de santé et des pratiques médicales". Comme le mentionne le projet scientifique de l'ULR présenté à l'HCERES, l'évaluation des technologies de santé a une finalité applicative (éclairer la décision) mais repose sur un ensemble de méthodes qui évoluent et se perfectionnent. L'axe "évaluation clinique" est un axe où la construction de scores cliniques est mise en avant. Les méthodes à mettre en œuvre pour le développement et la validation d'un score clinique sont très bien balisées pour des plateformes de support méthodologique. En revanche, il existe encore des questions méthodologiques liées à l'utilisation de données provenant de nouvelles technologies comme les puces à ADN, les spectromètres de masse ou le séquençage à haut débit. Ces techniques ont introduit de nouvelles questions statistiques pour l'analyse de données biologiques. Elles permettent en effet de mesurer simultanément plusieurs milliers de variables -omiques, dont le nombre est largement supérieur au nombre d'individus étudiés. En recherche médicale, ces expériences à haut débit sont souvent utilisées comme approche exploratoire pour pré-sélectionner des facteurs d'intérêt pour l'étude d'une maladie. On peut par exemple rechercher des anomalies le long du génome (comme des variations du nombre de copies d'ADN), des gènes différentiellement exprimés (c'est à dire présentant une différence d'expression entre des cellules saines et tumorales par exemple) avant de rechercher des facteurs prédictifs (qui permettent de prédire une rechute par exemple). Les approches statistiques considérées dépendent de la question biologique posée. Ces dernières années, mes développements méthodologiques ont concerné ces grandes questions biologiques, reposant sur trois types d'approches statistiques : approches bayésiennes empiriques (cf. chapitre 2), détection de ruptures à base de noyaux (cf. chapitre 3) et régressions pénalisées, point commun aux trois thèses co-encadrées (cf. chapitres 3 et 4). Ces recherches ont introduit mes travaux actuels autour de la construction de scores, l'un des thèmes privilégiés de l'axe 2 (évaluation clinique) de mon unité de recherche METRICS. Des perspectives ont ainsi été données dans le chapitre 4. Mes développements méthodologiques sont toujours accompagnés d'outils logiciels, afin d'assurer la diffusion la plus large possible de ces nouvelles approches statistiques. Enfin, mon implication importante dans la plateforme bilille m'a permis d'interagir avec différentes communautés et j'espère que d'autres collaborations interdisciplinaires fructueuses naîtront dans les années à venir.

Bibliography

R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422(6928):198–207, 2003. URL https://www.nature.com/articles/nature01511.

E.-a. D. Amir. *viSNE and Wanderlust, two algorithms for the visualization and analysis of high-dimensional single-cell data*. PhD thesis, Columbia University, 2014. URL https://doi.org/10.7916/D8SB43VK.

S. Arlot, A. Celisse, and Z. Harchaoui. A Kernel Multiple Change-point Algorithm via Model Selection. *arXiv:1202.3878*, 2012. URL http://arxiv.org/abs/1202.3878.

C. Audebert, D. Hot, Y. Lemoine, and S. Caboche. Le séquençage haut-débit: Vers un diagnostic basé sur la séquence complète du génome de l'agent infectieux. *médecine/sciences*, 30(12): 1144–1151, 2014. URL http://www.medecinesciences.org/10.1051/medsci/20143012018.

B. Barlogie, M. N. Raber, J. Schumann, T. S. Johnson, B. Drewinko, D. E. Swartzendruber, W. Göhde, M. Andreeff, and E. J. Freireich. Flow Cytometry in Clinical Cancer Research. *Cancer Research*, 43(9):3982–3997, 1983. URL https://cancerres.aacrjournals.org/content/43/9/3982.

P. S. Bhatia, S. Iovleff, and G. Govaert. blockcluster: An R Package for Model-Based Co-Clustering. *Journal of Statistical Software*, 76(1):1–24, 2017. URL https://www.jstatsoft.org/index.php/jss/article/view/v076i09.

L. Birgé and P. Massart. Minimal Penalties for Gaussian Model Selection. *Probability Theory and Related Fields*, 138(1):33–73, 2007. URL https://doi.org/10.1007/s00440-006-0011-8.

S. Blanck and G. Marot. SMAGEXP: a galaxy tool suite for transcriptomics data meta-analysis. *GigaScience*, 8(2), 2019. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6354025/.

M. Boutros, L. P. B. L, and W. Huber. Analysis of cell-based RNAi screens. *Genome Biology*, 7 (7):R66, 2006. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1779553/.

P. Brodin, E. DelNery, and E. Soleilhac. Criblage phénotypique à haut contenu pour la chémobiologie et ses enjeux. *Médecine sciences*, 31:187–196, 2015. URL http://www.ipubli.inserm.fr/handle/10608/8571.

I. Curie contributors. Le cancer, une maladie des gènes | Institut Curie, 2020. URL https://curie.fr/dossier-pedagogique/le-cancer-une-maladie-des-genes.

K. Dettmer, P. A. Aronov, and B. D. Hammock. Mass spectrometry-based metabolomics. *Mass Spectrometry Reviews*, 26(1):51–78, 2007. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mas.20108.

D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, Sept. 1994. URL https://academic.oup.com/biomet/article/81/3/425/256924.

E. Garg, C. Murie, and R. Nadon. *sights: Statistics and dIagnostic Graphs for HTS*, 2016. URL https://www.bioconductor.org/packages/devel/bioc/vignettes/sights/inst/doc/sights.html. R package version 1.10.0.

M. Giacofci, S. Lambert-Lacroix, G. Marot, and F. Picard. Wavelet-Based Clustering for Mixed-Effects Functional Models in High Dimension. *Biometrics*, 69(1):31–40, 2013. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1541-0420.2012.01828.x.

Q. Grimonprez, A. Celisse, S. Blanck, M. Cheok, M. Figeac, and G. Marot. MPAgenomics: an R package for multi-patient analysis of genomic markers. *BMC Bioinformatics*, 15(1):394, 2014. URL https://doi.org/10.1186/s12859-014-0394-y.

A. B. Haidich. Meta-analysis in medical research. *Hippokratia*, 14(Suppl 1):29–37, 2010. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3049418/.

C. Herbaux, E. Bertrand, G. Marot, C. Roumier, N. Poret, V. Soenen, O. Nibourel, C. Roche-Lestienne, N. Broucqsault, S. Galiègue-Zouitina, E. M. Boyle, G. Fouquet, A. Renneville, S. Tricot, F. Morschhauser, C. Preudhomme, B. Quesnel, S. Poulain, and X. Leleu. BACH2 promotes indolent clinical presentation in Waldenström macroglobulinemia. *Oncotarget*, 8(34): 57451–57459, 2016. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5593656/.

L. Hood, R. Balling, and C. Auffray. Revolutionizing medicine in the 21st century through systems approaches. *Biotechnology journal*, 7(8):992–1001, 2012. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3962497/.

F. Jaffrézic, G. Marot, S. Degrelle, I. Hue, and J.-L. Foulley. A structural mixed model for variances in differential gene expression studies. *Genetics Research*, 89(1):19–25, 2007. URL https://doi.org/10.1017/S0016672307008646.

R. Killick, P. Fearnhead, and I. A. Eckley. Optimal Detection of Changepoints With a Linear Computational Cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012. URL https://www.jstor.org/stable/23427357.

E. Lebarbier. Detecting multiple change-points in the mean of Gaussian process by model selection. *Signal Processing*, 85(4):717–736, 2005. URL http://www.sciencedirect.com/science/article/pii/S0165168404003196.

M. I. Love, W. Huber, and S. Anders. Moderated estimation of fold change and dispersion for rna-seq data with DESeq2. *Genome Biology*, 15:550, 2014. URL https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0550-8.

L. v. d. Maaten and G. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

G. Marot and R. Bruyère. Using metaMA for differential gene expression analysis from multiple studies, 2015. URL https://cran.r-project.org/web/packages/metaMA/vignettes/metaMA.pdf.

G. Marot, J.-L. Foulley, C.-D. Mayer, and F. Jaffrézic. Moderated effect size and P-value combinations for microarray meta-analyses. *Bioinformatics*, 25(20):2692–2699, 2009. URL https://academic.oup.com/bioinformatics/article/25/20/2692/192916.

N. Martin, C. Salazar-Cardozo, C. Vercamer, L. Ott, G. Marot, P. Slijepcevic, C. Abbadie, and O. Pluquet. Identification of a gene signature of a pre-transformation process by senescence evasion in normal human epidermal keratinocytes. *Molecular Cancer*, 13(1):151, 2014. URL https://doi.org/10.1186/1476-4598-13-151.

N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2010.00740.x.

D. A. Mogilenko, J. T. Haas, L. L'homme, S. Fleury, S. Quemener, M. Levavasseur, C. Becquart, J. Wartelle, A. Bogomolova, L. Pineau, O. Molendi-Coste, S. Lancel, H. Dehondt, C. Gheeraert, A. Melchior, C. Dewas, A. Nikitin, S. Pic, N. Rabhi, J.-S. Annicotte, S. Oyadomari, T. Velasco-Hernandez, J. Cammenga, M. Foretz, B. Viollet, M. Vukovic, A. Villacreces, K. Kranc, P. Carmeliet, G. Marot, A. Boulter, S. Tavernier, L. Berod, M. P. Longhi, C. Paget, S. Janssens, D. Staumont-Sallé, E. Aksoy, B. Staels, and D. Dombrowicz. Metabolic and Innate Immune Cues Merge into a Specific Inflammatory Response via the UPR. *Cell*, 177(5): 1201–1216.e19, 2019. URL https://doi.org/10.1016/j.cell.2019.03.018.

J.-Y. Nau. Pour en finir avec le dogme central de la biologie moléculaire (3), 2003. URL https://www.revmed.ch/RMS/2003/RMS-2430/1012.

B. Phipson, S. Lee, I. J. Majewski, W. S. Alexander, and G. K. Smyth. Robust hyperparameter estimation protects against hypervariable genes and improves power to detect differential expression. *The annals of applied statistics*, 10(2):946–963, 2016. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5373812/.

F. Picard, S. Robin, M. Lavielle, C. Vaisse, and J.-J. Daudin. A statistical approach for array CGH data analysis. *BMC Bioinformatics*, 6(1):27, 2005. URL https://doi.org/10.1186/1471-2105-6-27.

S. Poulain, C. Roumier, A. Venet-Caillault, M. Figeac, C. Herbaux, G. Marot, E. Doye, E. Bertrand, S. Geffroy, F. Lepretre, O. Nibourel, A. Decambron, E. M. Boyle, A. Renneville, S. Tricot, A. Daudignon, B. Quesnel, P. Duthilleul, C. Preudhomme, and X. Leleu. Genomic Landscape of CXCR4 Mutations in Waldenström Macroglobulinemia. *Clinical Cancer Research*, 22(6):1480–1488, 2016. URL http://clincancerres.aacrjournals.org/content/22/6/1480.

A. Renneville, R. B. Abdelali, S. Chevret, O. Nibourel, M. Cheok, C. Pautas, R. Duléry, T. Boyer, J.-M. Cayuela, S. Hayette, E. Raffoux, H. Farhat, N. Boissel, C. Terre, H. Dombret, S. Castaigne, and C. Preudhomme. Clinical impact of gene mutations and lesions detected by SNP-array karyotyping in acute myeloid leukemia patients in the context of gemtuzumab ozogamicin treatment: results of the ALFA-0701 trial. *Oncotarget*, 5(4):916–932, 2014. URL https://www.oncotarget.com/article/1536/text/.

G. Rigaill. A pruned dynamic programming algorithm to recover the best segmentations with 1 to K_{max} change-points. *arXiv:1004.0887*, 2015. URL http://arxiv.org/abs/1004.0887.

M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, 43(7):e47, 2015. URL https://doi.org/10.1093/nar/gkv007.

M. D. Robinson, D. J. McCarthy, and G. K. Smyth. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, 2010. URL https://academic.oup.com/bioinformatics/article/26/1/139/182458.

C. Scheeder, F. Heigwer, and M. Boutros. HTSvis: a web app for exploratory data analysis and visualization of arrayed high-throughput screens. *Bioinformatics*, 33(18):2960–2962, 2017. URL https://academic.oup.com/bioinformatics/article/33/18/2960/3827333.

G. K. Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3:Article3, 2004. URL https://doi.org/10.2202/1544-6115.1027.

R. Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. URL https://www.jstor.org/stable/2346178.

D. Valour, I. Hue, S. A. Degrelle, S. Déjean, G. Marot, O. Dubois, G. Germain, P. Humblot, A. A. Ponter, G. Charpigny, and B. Grimard. Pre- and Post-Partum Mild Underfeeding Influences Gene Expression in the Reproductive Tract of Cyclic Dairy Cows. *Reproduction in Domestic Animals*, 48(3):484–499, 2013. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/rda.12113.

C. D. M. van Karnebeek, S. B. Wortmann, M. Tarailo-Graovac, M. Langeveld, C. R. Ferreira, J. M. van de Kamp, C. E. Hollak, W. W. Wasserman, H. R. Waterham, R. A. Wevers, T. B. Haack, R. J. Wanders, and K. M. Boycott. The role of the clinician in the multi-omics era: are you ready? *Journal of Inherited Metabolic Disease*, 41(3):571–582, 2018. URL https://doi.org/10.1007/s10545-017-0128-1.

Wikipedia contributors. High-content screening. https://en.wikipedia.org/w/index.php?title=High-content_screening&oldid=951578283, 2020.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006. URL https://doi.org/10.1111/j.1467-9868.2005.00532.x.