Machine learning tools for biomarker discovery

Chloé-Agathe Azencott

MACHINE LEARNING TOOLS FOR BIOMARKER DISCOVERY

Chloé-Agathe Azencott

Habilitation à Diriger des Recherches

Déposée à l'UFR Ingénierie de Sorbonne Université
Spécialité Informatique

January 30, 2020

Membres du jury

M. Mario Marchand (Université Laval, Canada)
Rapporteur

Mme Nataša Pržulj (University College London, UK, and Barcelona Supercomputing Center, Spain)
Rapporteuse

M. Bertrand Thirion (Inria Saclay, France)
Rapporteur

M. Michel Blum (Université Grenoble Alpes, France)
Examinateur

M. Grégory Nuel (Sorbonne Université, France)
Examinateur

Mme Marylyn Ritchie (University of Pennsylvania, USA)
Examinatrice

RÉSUMÉ

Mes travaux de recherche s'inscrivent dans le cadre du développement de techniques d'apprentissage statistique (« machine learning ») pour la recherche thérapeutique.

Ils visent en particulier à proposer des outils informatiques permettant d'exploiter des jeux de données pour en extraire des hypothèses biologiques expliquant au niveau génomique ou moléculaire les différences entre échantillons observées à un niveau macroscopique. De tels outils sont nécessaires à la mise en œuvre de la médecine de précision, qui requiert d'identifier les caractéristiques, génomiques ou autres, expliquant les différences de pronostic ou de réponse thérapeutique entre patients présentant les mêmes symptômes.

Ces questions peuvent souvent être formulées comme des problèmes de sélection de variables. Les jeux de données utilisés, cependant, contiennent généralement largement plus de variables que d'échantillons, ce qui pose des difficultés statistiques. Pour répondre à ces défis, mes travaux s'orientent autour de trois axes.

Premièrement, les connaissances accumulées sur les systèmes biologiques peuvent souvent être représentées sous la forme de réseaux biologiques. Sous l'hypothèse que les variables connectées par ces réseaux sont susceptibles d'agir conjointement sur un phénotype, nous proposons d'utiliser ces réseaux pour guider un algorithme de sélection de variables. Il s'agit ici d'utiliser des contraintes qui encouragent les variables sélectionnées à être connectées sur un réseau donné. La formulation que nous avons proposée, qui s'inscrit dans le cadre plus large de ce que j'appelle la pertinence régularisée, permet de résoudre efficacement le problème de sélection de variables sur des jeux de données comportant des centaines de milliers de variables.

Deuxièment, pour compenser le faible nombre d'échantillons disponibles, les méthodes dites multitâches résolvent simultanément plusieurs problèmes, ou tâches, proches. Nous avons étendu la pertinence régularisée à ce contexte. Je me suis aussi intéressée au cas où il est possible de définir une similarité entre tâches, afin d'imposer que les variables sélectionnées pour deux tâches soient d'autant plus similaires que les deux tâches sont semblables. Ces approches sont pertinentes dans le cas de l'étude de la réponse à différents traitements médicamenteux : on peut alors utiliser la similarité entre les structures moléculaires de ces médicaments, sujet que j'ai étudié pendant ma thèse.

Enfin, la plupart des approches de sélection de variables utilisées dans le contexte de la génomique ne peuvent expliquer le phénomène d'intérêt que par des effets linéaires. Cependant, de nombreux travaux indiquent que les régions du génome peuvent interagir de façon non-linéaire. Modéliser de telles interactions, que l'on qualifie d'épistatiques, aggrave cependant les problèmes statistiques déjà rencontrés précédemment, et crée aussi des problèmes computationnels : il devient difficile d'évaluer toutes les combinaisons possibles de variables. Mes travaux portent aussi bien sur les difficultés calculatoires que sur les difficultés statistiques rencontrées dans la modélisation d'interactions quadratiques entre paires de régions du génomes. Plus récemment, nous avons aussi développé des approches permettant la modélisation d'interactions plus complexes grâce à des méthodes à noyaux.

ACKNOWLEDGMENTS

This document summarizes the results of fourteen years of work in machine learning applications to bioinformatics and chemoinformatics. The number of people that should be thanked is staggering, and I will therefore have to omit a large number of them.

First and foremost, I am deeply indepted to my many collaborators, without whom none of the work nor ideas I present here would have happened. It is with great pleasure that I thank Fabian Aicheler, Tero Aittokallio, André Altmann, Nadine Andrieu, Pierre Baldi, Annalisa Barla, Víctor Bellón, Valentina Boeva, Karsten M. Borgwardt, Lawrence Cayton, Clément Chatelain, Jonathan H. Chen, Héctor Climente-González, Kenny Daily, Mark Daly, Ahmed Debit, Stefani Dritsa, Laramie Duncan, Diane Duroux, Fajwel Fogel, Samuele Fiorini, Stephen Friend, Xavier Gidrol, Udo Gieraths, Anna Goldenberg, Dominik Grimm, Anne-Sophie Hamy-Petit, Tony Kam-Thong, Nazanin Karbalai, Samuel Kaski, Yoshinobu Kawahara, Alexandre Ksikes, Matthew A. Kayala, Christophe Le Priol, Fabienne Lesueur, Ting-Wan Lin, Felipe Llinares López, Christine Lonjou, Daniel MacArthur, Thibaud Martinez, Bertram Müller-Myhsok, Asma Nouira, Thea Norman, Charlotte Proudhon, Benno Pütz, Liva Ralaivola, Antonio Rausell, Antoine Recanati, Fabien Reyal, Daniel Rovera, Sushmita Roy, Kaitlin Samocha, Philipp G. Sämann, Bernard Schölkopf, Nino Shervashidze, Carl-Johann Simon-Gabriel, Lotfi Slim, Jordan Smoller, Gustavo Stolovitzky, Véronique Stoven, Mahito Sugiyama, S. Joshua Swamidass, Sheryl Tsai, Kristel Van Steen, Jean-Philippe Vert, Makoto Yamada, and Weiyi Zhang.

My work was in part inspired by two DREAM challenges. I hence also wish to acknowledge all of the many people involved in the various stages of these challenges; all members of the NIEHS-NCATS-UNC DREAM Toxicogenetics Collaboration and The Rheumatoid Arthritis Challenge Consortium; and more particularly, Christopher Bare, Jing Cui, Federica Eduati, Andre O. Falcao, Mike Kellen, Lara M. Mangravite, Michael P. Menden, Gaurav Pandey, Dimitrios Pappas, Abhishek Pratap, Julio Saez-Rodriguez, Solveig K. Siebert, Eli Stahl, Christine Suver, and Yang Xie.

Along the years, my work has been supported by Agence Nationale pour la Recherche, the Alexander von Humboldt Stiftung, the Deutsche Forschungsgemeinschaft, and the European Research Council, as well as by Sancare and Sanofi-Adventis. I am grateful for their funding.

I am very happy to thank all current and past members of my current lab, the Centre for Bioinformatics of Mines ParisTech. I would particularly like to emphasize the role of Véronique Stoven, Jean-Philippe Vert, and Thomas Walter, for their warm welcome in their group, for their constant support, for all I have learned and am still learning from them, and for the friendly atmosphere they kindle at CBIO.

With this HDR, I am expected to demonstrate my ability to conduct research independently, and to supervise trainees. I could not show the latter without the trust that my (co-supervised) trainees I have placed in me, and I therefore thank Víctor Bellón, my first PhD student; Christophe Le Priol, Héctor Climente-González, Lotfi Slim, and Asma Nouira, who followed suit; as well as Jean-Daniel Granet, Thibaud Martinez, Killian Poulaud,

Antoine Récanati, Athénaïs Vaginay and Weiyi Zhang. Among former CBIO members, Marine Le Morvan, Benoît Playe, Erwan Scornet, and Nelle Varoquaux deserve special thanks as well, for scientific and non-scientific conversations alike. I am also pleased to thank Emmanuel Barillot, who makes our inclusion in the U900 team of Institut Curie possible.

The Machine Learning and Computational Biology lab led by Karsten Borgwardt and formerly in Tübingen is where I started becoming an independent researcher. Warm thanks go to all the team, particularly Karsten; but also Aasa Feragen, Dominik Grimm, Theofanis Karaletsos, Christoph Lippert, Barbara Rakitsch, Nino Shervashidze, Oliver Stegle, and Mahito Sugiyama, whom I am happy to count as friends and colleagues. Thank you also to Fabian Aicheler, Udo Gieraths, and Valeri Velkov, who were the first students whose research I supervised.

I do not discuss here much of the work I conducted at UC Irvine for my PhD thesis – there are already two hundred pages or so that you can read on the topic. Still, I am thankful to Pierre Baldi for the environment he provided, and to Jon Chen, Kenny Daily, Matt Kayala, Ramzi Nasr, Vishal Patel, and Josh Swamidass, for both hard work and friendship.

For the two years of sisterhood, fighting together to increase the visibility of women in machine learning and data science, I thank past and current members of the Paris WiMLDS team: Chiara Biscaro, Natalie Cerneka, Caroline Chavier, Fanny Riols, and Marina Vinyes.

Many scientists have supported me along the years, helping me build confidence and regain my footing in difficult situations. I cannot thank them all, but I would like to mention Annalisa Barla, Danielle Belgrave, Matthew Blashko, Kevin Bleakley, Dean Bodenham, Laurence Calzone, Igor Carron, Émilie Chautru, Veronika Cheplygina, Florence d'Alché-Buc, Alexandre Gramfort, Søren Hauberg, Laurent Jacob, Neil Lawrence, Tijana Milenkovic, Richard Neher, Cheng Soon Ong, Elizabeth Purdom, Franck Rapaport, Magnus Rattray, Charlotte Truchet, and Detlef Weigel. Some of you may be surprised to appear in this list; but believe me, you were here at the right time, with the right words, and this made all the difference.

For their love, and for their ability to listen to me talk about my work for hours, and for helping me snapping out of it when necessary, I thank Alix, my sweetheart; Nicole, my mother; and my friends Alexandre, Andrey, Auréliane, Camille, Fabio, Gabriel, and Louise. Garance is too young to pay attention when I talk about work, but still, she is the most wonderful little girl on earth.

During my PhD I danced; since then I've leaned more and more towards playing music. Several orchestras – and their members – have kept me sane along the years, whether through rehearsals, concerts, tours, or post-rehearsal drinks. You are all welcome, particularly those of you who donated so much of your time to the logistics; those who have become friends; and of course, all the violists and conductors.

My final thanks are for Michel Blum, Mario Marchand, Grégory Nuel, Nataša Pržulj, Marylyn Ritchie, and Bertrand Thirion, who have accepted to read this document and sit on my HDR committee. Thank you so much for your time and enthusiasm. My thoughts go to François Laviolette, who had to step down from this committee for health reasons.

CONTENTS

1

CONTEXT

Differences in disease predisposition or response to treatment can be explained in great part by genomic differences between individuals [218]. In consequence, there is a growing interest for incorporating genomic data into *precision medicine*, that is, tailoring disease treatment and prevention strategies to the individual genomic characteristics of each patient [4].

To be able to use genetic characteristics in precision medicine, we need to identify genetic features, which I call here *biomarkers*, associated with disease risk, diagnostic, prognosis, or response to treatment. This endeavor hence depends on collecting considerable amounts of molecular data for large numbers of individuals. It is enabled by thriving developments in genome sequencing and other high-throughput experimental technologies, thanks to which it is now possible to accumulate millions of genomic descriptors for thousands of individuals.

Unfortunately, we still lack effective mathematical methods to reliably detect, from these high-dimensional data, which of these genomic descriptors (or *features*, or *variables*) determine a phenotype such as disease predisposition or response to treatment [142, 252].

In this chapter, I will briefly give some background on both omics and health (Section 1.1) and on statistical methods for feature selection in high dimension, before higlighting some of the challenges of extracting relevant features from omics data (Section 1.2). I will also introduce biological networks and a few concepts of network science in Section 1.3. Finally, I will outline my contributions to this domain (Section 1.4).

1.1 Omics and health

Most of my recent scientific contributions belong to the domain of precision medicine. In this section, I will briefly introduce this field, as well as genome-wide association studies, on which I have focused much of my efforts.

1.1.1 *Precision medicine*

"Precision medicine" – sometimes also "personalized medicine" – is a term used to describe using information beyond the patient's symptoms to diagnose or treat their disease. These information can be clinical (age, sex, blood test results) or genetic. The social and economical potentials of precision medicine are huge, particularly in cancer applications. This is underlined by several recent large-scale initiatives, such as The Cancer Genome Atlas (TCGA)[1], the recent opening of the UC San Francisco Precision Cancer Medicine Building, President Obama's 2015 Precision Medicine Initiative, or, in France, the Médecine France Génomique 2025 plan.

1 http://cancergenome.nih.gov/

Early examples of the usage of genomic information in precision medicine include the breast cancer drug trastuzumab (Herceptin), which dramatically improves the prognosis of patients whose tumor overexpresses the HER-2 gene, or the colon cancer drugs cetuximab (Erbitux) and panitumumab (Vectibix), which are known to have little effect on patients that have a mutation in the KRAS gene. How can we further encourage such discoveries?

1.1.2 *Genome-wide association studies*

Thanks to thriving developments in high-throughput experimental technologies, it is now rather easy to collect tens of millions of genomic descriptors for thousands of biological samples. These descriptors are diverse in nature, and contain data types such as single-point mutations, DNA methylation patterns, or gene expression levels.

Gene expression data, in which the messenger RNA levels of tens of thousands of genes are measured either thanks to RNA microarray or, more recently, RNA sequencing technologies, are possibly the most widespread of these molecular data types.

Genome-Wide Association Studies. In most of my work over the past eight years, however, I have focused on a specific type of molecular data, collected in the context of *Genome-Wide Association Studies*, or *GWAS*. GWAS are one of the prevalent tools for detecting genetic variants associated with a phenotype. They consist in collecting, for a large cohort of individuals, the alleles they exhibit across of the order of 250 000 to several millions of *Single Nucleotide Polymorphisms*, or *SNPs*. SNPs are individual locations across the genome where nucleotide variations can occur. The same individuals are also phenotyped, meaning that a trait of interest is recorded for each of them. This trait can be binary, such as disease status, or continuous, such as age of onset or reduction in tumoral burden. The goal of these studies is to identify which of the SNPs are associated with the phenotype of interest.

Missing heritability. While GWAS have provided novel insights into the pathways underpinning many common human diseases, a number of frustrating results have also been reported [255]. Indeed, most of the genotype-to-phenotype associations they have detected are weak, many of their findings have failed to be replicated in other studies, and the genetic variants they uncovered often fall short of explaining all of the phenotypic variation that is known to be inheritable. This last phenomenon is often referred to as the *missing heritability* problem [158].

Many reasons have been advanced for these shortcomings [158, 171]. Among those, I am particularly interested in the lack of statistical power that is due to the relatively small number of samples compared to that of features (see Section 1.2), as well as the failure to account for nonlinear effects. I find it important to note, however, that phenotypic inheritability can be due to genomic variation other than SNPs, such as epigenetics or DNA copy number, and intertwined with environmental effects; this suggests that the proportion of phenotypic variation that can be explained from GWAS is possibly smaller than previously thought.

These issues are not limited to SNP data, and indeed span the breadth of data-driven biology. For example, state-of-the-art approaches on gene expression data yield disconcertingly disparate molecular signatures for the same phenotype [65]. Hence, although I

frequently focus on GWAS for simplicity, the methods I describe can often be applied to all sorts of molecular data sets.

Linkage disequilibrium The difficulty of feature selection in high dimension is exacerbated by the presence of correlation between features. Unfortunately, molecular features tend to present high levels of correlations. In particular, although recombination can be expected to break down non-random genetic associations between genetic loci, such associations exist, and are refered to as *linkage disequilibrium*, or *LD*. Linkage disequilibrium can be due to genetic linkage, that is to say, the tendency for variants located nearby on the same chromosome to be transmitted together through meiosis, therefore being highly correlated. In addition, LD patterns can be due to natural selection, which may favor alleles that together affect reproductive fitness; genetic drift; or population bottlenecks, to name a few reasons.

Linkage disequilibrium is often addressed by *LD pruning*, which is a preprocessing step consisting in sequentially scanning the genomes for pairs of correlated SNPs, keeping only the one with the higher minor allele frequency. Pruning may leave regions of the genomes without any representative SNP. By contrast, *LD clumping* is performed after the association tests, which ensures that the SNPs with the strongest test statistics are not filtered out [199].

1.2 Feature selection in high dimension

In most of my work, I cast biomarker discovery as a *feature selection* problem. In this section, I will sketch a few approaches for feature selection, in particular statistical tests – which will be considered filtering approaches in machine learning – and embedded methods, with regularized linear regressions. I will also discuss the notion of stability of a feature selection method, and the extension of these approaches to nonlinear models.

1.2.1 *Biomarker discovery as a feature selection problem*

The fields of statistics, machine learning, and data mining, which are central to the analysis of genomic data, have dramatically progressed in the last twenty-five years. *Feature selection*, which aims at identifying the most important features in a data set and discarding those that are irrelevant or redundant [92], is of particular interest for identifying biologically relevant features.

However, too few of the recent efforts in this area have been focused on the challenges that molecular data represent, as they exhibit few samples in high dimension. Indeed, there is a broadening gap between the number of features we are able to measure for a given sample (easily reaching tens of millions with current technologies) and the number of samples we can collect (more commonly in the order of thousands).

This high-dimensional, low sample-size situation drastically limits the power of general-purpose statistical and machine learning approachess [44, 114]. In sharp contrast with the current “big data” vision, we cannot expect this problem to disappear with improvements in technology: the number of individuals with a given condition that we can sequence will never outgrow the millions of features that can be collected about them. This issue, far from being restricted to genomic data, is of broad interest in a variety of domains ranging from medical imaging to quantitative finance and climate science.

In what follows, I will review classical approaches to feature selection and their limitations when it comes to biomarker discovery. I find it important to note here that, while the presence of statistical associations in data can help generate new hypotheses, corroborations on independent data sets and in-depth investigations of the underlying molecular mechanisms are necessary to substantiate these statistical findings.

1.2.2 *Notations*

Throughout this document, unless otherwise noted, the data we work with will be represented by a couple $(\boldsymbol{X}, \boldsymbol{y}) \in \mathbb{R}^{n \times m} \times \mathbb{R}^n$. n is the number of samples or individuals, and m the number of genomic features that have been measured for each of them. $\boldsymbol{X}$ is the genotype data matrix, and $\boldsymbol{y}$ represents the phenotype. In the case of a qualitative (case-control) phenotype, $\boldsymbol{y}$ belongs to $\{0, 1\}^n$. I will denote by x_{ip} (sometimes X_{ip}) the entry at the i-th line and p-th column of $\boldsymbol{X}$, that is to say, the genotype at the p-th measurement (gene expression, SNP, methylation status, etc) of sample i. y_i will denote the phenotype of sample i. The m-dimensional vector describing sample i is denoted as $\boldsymbol{x}_i$. The underlying assumption is that the n couples $(\boldsymbol{x}_i, y_i)$ are realizations of two random variables X, which is m-dimensional, and Y, which is either binary or real-valued.

In the most general case, the measurements along the genome are real-valued number. However, in human genetics, SNPs are ternary variables, as an individual can either be homozygous in the major (most frequent) allele, homozygous in the minor (least frequent) allele, or heterozygous. Different encodings of the SNPs correspond to different biological models. For example, the *dosage encoding*, in which the SNP is encoded by its number of minor alleles (0 for homozygous major, 1 for heterozygous, 2 for homozygous minor), supposes a different effect for all three possibilities. By contrast, the *dominant encoding*, where the SNP is encoded by 0 in the absence of minor allele and 1 otherwise, supposes a dominant effect. Conversely, in the *recessive encoding*, the SNP is encoded by 1 if it is homozygous in the minor allele, and 0 otherwise.

1.2.3 *Filtering approaches: statistical tests*

Filtering approaches to feature selection consist in considering the features one by one, and evaluating for each of them, independenty of the others, whether it is correlated, or associated, with the outcome of interest. Statistical tests are the most common approaches to biomarker discovery from molecular data.

Statistical tests for gene expression data. One of the most common analysis of gene expression data consists in running statistical tests to determine the genes that are differentially expressed between two conditions. Differential variability analysis is also garnering interest, as several studies have identified differentially variable genes involved in cancer [62, 103] or in neurological disorders [159, 281]. Most statistical tests to compare gene expression between two conditions, such as edgeR [208] and DESeq [3, 154] for differential expression and MDSeq [203] and DiPhiSeq [143] for differential dispersion, are based on the negative binomial distribution. While this topic is outside the scope of the present document, we recently applied these approaches to the detection of genes with a differential expression dispersion in cancer [198].

Statistical tests for GWAS. In GWAS, statistical tests are run to detect associations between the SNPs and the phenotype. These statistical tests can account for linkage disequilibrium [50, 243], leverage linear mixed models to correct for sample relatedness – which invalidates the assumption of population homogeneity underlying most tests [188, 201], or assess the joint contribution of multiple genetic loci, either additively [268, 282] or multiplicatively for pairwise interactions [104, 262, 284]. An overview of classical GWAS techniques can be found in Bush and Moore [34] or Gumpinger et al. [91].

1.2.4 *Embedded approaches: regularized linear regression*

A major drawback of filtering approaches is that features are selected independently from each other. By contrast, so-called embedded approaches [92] consider all features jointly.

Embedded approaches offer a way to detect combinations of variants that are associated with a phenotype. Indeed, they learn which features contribute best to the accuracy of a machine learning model (a classifier in the case of case/control studies, or a regressor in the case of a quantitative phenotype), while it is being built.

Within this framework, the leading example is that of linear regression [95]. A linear regression model assumes that the phenotype can be explained as a linear function of the biomarkers:

$$y_i = \sum_{p=1}^{m} x_{ip}\beta_p + \epsilon_i, \tag{1.1}$$

where the regression weights $\beta_1, \ldots, \beta_m$ are unknown parameters and ϵ_i is an error term. Note that we can equally assume that the mean of y is 0, or that the first of the m biomarkers is a mock feature of all ones that will serve to estimate the bias of the model. The least-squares methods provides estimates of $\beta_1, \ldots, \beta_m$ by minimizing the least-square objective function (or data-fitting term) given in matrix form by Eq. (1.2):

$$\underset{\beta \in \mathbb{R}^m}{\arg\min} ||\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}||_2^2. \tag{1.2}$$

Regularization When $m \gg n$, as it is the case in most genome-wide biomarker discovery datasets, Eq. (1.2) has an infinite set of solutions. In order to *regularize* the estimation procedure, one can add to the least-square objective function a *penalty term*, or *regularization term*, that will force the regression weights to respect certain constraints. Eq. (1.2) becomes

$$\underset{\beta \in \mathbb{R}^m}{\arg\min} ||\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}||_2^2 + \lambda\, \Omega(\boldsymbol{\beta}), \tag{1.3}$$

where $\Omega : \mathbb{R}^m \to \mathbb{R}$ is a regularizer and $\lambda \in \mathbb{R}^+$ is a parameter which controls the balance between the relevance and the regularization terms, and is typically set by cross-validation.

Ridge regression The more well-known exemple of such a regularized regression is probably the *rigde regression*, for which

$$\Omega_{\text{ridge}}(\boldsymbol{\beta}) = ||\boldsymbol{\beta}||_2. \tag{1.4}$$

The ridge regression estimator of $\boldsymbol{\beta}$, obtained by solving Eq. (1.3) with Ω given by Eq. (1.4), is biased, unlike the ordinary least squares estimator obtained by solving Eq. (1.2), but it has a lower variance. The resulting linear model is less likely to overfit.

Lasso A very popular regularizer for feature selection is the ℓ_1-norm of $\boldsymbol{\beta}$, $||\boldsymbol{\beta}||_1 = \sum_{p=1}^{m} |\beta_p|$, which has the effect of shrinking the β_p coefficients and setting a large number of them to zero. The resulting model is called *sparse* [97]. The features with zero weights do not enter the model and can hence be rejected; only the features with non-zero weights are considered to be selected. This results in the *lasso* [247], which estimates the regression weights by solving Eq. (1.5). The reason for using the ℓ_1-norm, rather than the ℓ_0-norm which counts the number of variables that enter the model and hence directly enforces sparsity, is that with the ℓ_0-norm the resulting objective function would be non-convex, making its minimization very challenging computationally.

$$\underset{\boldsymbol{\beta}\in\mathbb{R}^m}{\arg\min} ||\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}||_2^2 + \lambda\, ||\boldsymbol{\beta}||_1. \tag{1.5}$$

1.2.5 *Stability*

The *stability* (or *robustness*) of feature selection procedures, meaning their ability to retain the same features upon minor perturbations of the data, remains a major predicament in the high-dimensional, low sample-size setting. Indeed, current approaches tend to focus on the prediction error of the models they build, and finding the relevant features is much harder than finding those that give optimal predictivity [182]. Current algorithms are typically highly unstable, often yielding widely different results for different sets of samples relating to the same question [57]. In practice, filtering approaches based on t-test scores often still yields the most stable selection [99, 129]. This high variability implies that these algorithms capture idiosyncrasies rather than truly relevant features. This casts doubts on the reliability of predictive algorithms built on the selected features and impedes interpreting these features to yield novel biological insights.

Elastic net One of the reasons of the unstability of the lasso in high-dimensional settings is that, in such settings, features are correlated, either by nature (and this is the case in most molecular data sets) or merely by chance (as the number of samples is relatively small). The lasso will then randomly select one of a group of several correlated features. To avoid this, the *elastic net* [69, 289] uses a mixed ℓ_1 and ℓ_2 regularizer which will tend to select all of the correlated features that explain the outcome:

$$\underset{\boldsymbol{\beta}\in\mathbb{R}^m}{\arg\min} ||\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}||_2^2 + \lambda\, \left(\eta||\boldsymbol{\beta}||_1 + (1-\eta)||\boldsymbol{\beta}||_2^2\right). \tag{1.6}$$

While elastic net solutions tend to be more stable than lasso ones, they remain too unstable for interpretability.

Stability selection for the lasso Recent efforts to use multiple repetitions of the procedure on subsamples of the data to make feature selection algorithms more stable [164, 220], are yielding encouraging results. However, they are computationally intensive and their theoretical guarantees do not hold in high dimensions. Applications of these methods to biomarker detection remain rare.

Consistency index Several ways of measuring the stability of a feature selection method have been proposed [183]. Among them, the Kuncheva consistency index [128], which we generalized to the comparison of sets of selected features of different sizes [11], seems

the better adapted to our setting [183]. The consistency index between two feature sets $\mathcal{S}$ and $\mathcal{S}'$ is defined relative to the size of their overlap:

$$I_C(\mathcal{S}, \mathcal{S}') := \frac{\text{Observed}(|\mathcal{S} \cap \mathcal{S}'|) - \text{Expected}(|\mathcal{S} \cap \mathcal{S}'|)}{\text{Maximum}(|\mathcal{S} \cap \mathcal{S}'|) - \text{Expected}(|\mathcal{S} \cap \mathcal{S}'|)} \tag{1.7}$$

where $\text{Maximum}(|\mathcal{S} \cap \mathcal{S}'|) = \min(|\mathcal{S}|, |\mathcal{S}'|)$, and $\text{Expected}(|\mathcal{S} \cap \mathcal{S}'|)$ is the expectation of obtaining $|\mathcal{S} \cap \mathcal{S}'|$ features in $\mathcal{S}$ when randomly picking $|\mathcal{S}'|$ features out of m under the hypergeometric distribution:

$$\text{Expected}(|\mathcal{S} \cap \mathcal{S}'|) = \frac{|\mathcal{S}||\mathcal{S}'|}{m}. \tag{1.8}$$

Finally,

$$I_C(\mathcal{S}, \mathcal{S}') = \frac{m\,|\mathcal{S} \cap \mathcal{S}'| - |\mathcal{S}||\mathcal{S}'|}{n \min(|\mathcal{S}|, |\mathcal{S}'|) - |\mathcal{S}||\mathcal{S}'|}. \tag{1.9}$$

For an experiment with K folds, the consistency is computed as the average of the $\frac{K(K-1)}{2}$ pairwise consistencies between sets of selected features:

$$I_C(\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_K) = \frac{K(K-1)}{2} \sum_{k=1}^{K} \sum_{l=i+1}^{K} I_C(\mathcal{S}_k, \mathcal{S}_l). \tag{1.10}$$

1.2.6 *Nonlinearities and kernels*

In very high dimensions, feature selection approaches are generally limited to contemplating only additive effects between the variables, although many biological phenomena are nonlinear. While a variety of statistical tests have been developed to characterize so-called *epistatic* effects, most of those are limited to quadratic models involving only two SNPs at a time [170, 181, 207]. Indeed, the statistical problems that arise when considering more features than samples are aggravated in this context; if one has 500 000 SNPs to test, then the number of pairs of SNPs becomes of the order 125 billions.

Among the tools to model nonlinearities, *kernels* are dot products in complex, sometimes infinite-dimensional feature spaces, that can encode many types of interactions between the features originally describing the data in their input space.

Given an input space $\mathcal{X}$ (here, $\mathcal{X} = \mathbb{R}^m$), a kernel is a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, and such that there exists a function $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space, such that $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{x}, \boldsymbol{x}' \rangle_{\mathcal{H}}$. Here $\langle ., . \rangle_{\mathcal{H}}$ denotes the dot product on $\mathcal{H}$.

In addition, the Moore—Aronszajn theorem [5] states that any symmetric function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that verifies that, for any $n \in \mathbb{N}$, $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, $c_1, c_2, \ldots, c_n \in \mathbb{R}$,

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0, \tag{1.11}$$

there exists a Hilbert space $\mathcal{H}$ and a function $\phi : \mathcal{X} \to \mathcal{H}$ such that $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{x}, \boldsymbol{x}' \rangle_{\mathcal{H}}$, even if we do not have access to $\mathcal{H}$ or ϕ.

Because kernels can easily be computed on the input space, machine learning algorithms that only rely on dot products between objects, such as principal component analysis,

ridge regression, or support vector machines (SVMs), are amenable to the so-called *kernel trick*: they can be applied in feature space very efficiently, as all computations are done in input space.

In statistical genetics, kernels have long been used to compute the similarity of individuals based on their genomes [131, 145, 146]. Among the most frequently used kernels for this purpose, let us mention

- the *weighted linear kernel*, $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{p=1}^{m} w_p x_{ip} x_{jp}$. The weights $(w_1, w_2, \ldots, w_m)$ can be all identical, or set in such a way as to give more importance to some loci (such as rare variants, or variants likely to be deleterious);
- the *Identical By State (IBS) kernel* [131]: $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{2m} \sum_{p=1}^{m} \text{IBS}(x_{ip}, x_{jp})$, where $\text{IBS}(x_{ip}, x_{jp})$ denotes the number of alleles (0, 1, or 2) shared identical by state (meaning without any information as to whether they are identical by descent or just by chance) between individuals i and j at locus p. The kernel can also be weighted as above;
- the *weighted quadratic kernel*, $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left(\sum_{p=1}^{m} w_p x_{ip} x_{jp} + 1 \right)^2$, which models both additive and quadratic terms of interaction between variants. The weights are chosen as for the weighted linear kernel.

SKAT. The most well-known test based on kernels, the *Sequence Kernel Association Test*, or *SKAT*, is quite popular thanks to its flexibility. SKAT [268] is intended to compute the association between a set of variants and a phenotype; hence making it possible to include rare variants data in the association score between a genomic region and a phenotype. SKAT is a score-based variance-component test, meaning that it contrasts the variance of each observed allele with its expected variance. It can be used for both case-control and quantitative phenotypes, can account for covariates, and uses kernels to model potential non-linearities between SNPs. While the kernels presented above were computed over the entire genotype (all m SNPs), in SKAT the kernels use only the set of variants to be tested.

SKAT first fits a linear model between the non-genetic covariates of interest and the phenotye, and then computes a variance component score statistic as

$$Q = \widetilde{\boldsymbol{y}}^\top K \widetilde{\boldsymbol{y}}, \tag{1.12}$$

where $\widetilde{\boldsymbol{y}}$ is the residual of $\boldsymbol{y}$ under the aforementioned linear model and $K \in \mathbb{R}^{n \times n}$ the kernel matrix such that $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Under the null hypothesis, Q follows a mixture of chi-square distributions, which makes it possible to compute p-values for this statistical test analytically.

Hilbert-Schimdt Independence Criterion. Kernels have also been used for feature selection beyond genetics. In particular, measuring the dependence between two random variables X and Y can be achieved by the *Hilbert-Schimdt Independence Criterion*, or *HSIC* [84]:

$$\begin{aligned} \text{HSIC}(X, Y) = {} & \mathbb{E}_{\boldsymbol{x}, \boldsymbol{x}', \boldsymbol{y}, \boldsymbol{y}'}[k(\boldsymbol{x}, \boldsymbol{x}') l(\boldsymbol{y}, \boldsymbol{y}')] + \mathbb{E}_{\boldsymbol{x}, \boldsymbol{x}'}[k(\boldsymbol{x}, \boldsymbol{x}')] \mathbb{E}_{\boldsymbol{y}, \boldsymbol{y}'}[l(\boldsymbol{y}, \boldsymbol{y}')] \\ & - 2\mathbb{E}_{\boldsymbol{x}', \boldsymbol{y}} \left[\mathbb{E}_{\boldsymbol{x}'}[k(\boldsymbol{x}, \boldsymbol{x}')] \mathbb{E}_{\boldsymbol{y}'}[l(\boldsymbol{y}, \boldsymbol{y}')] \right], \end{aligned} \tag{1.13}$$

where $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ and $l : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ are positive definite kernels, and $\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}',\boldsymbol{y},\boldsymbol{y}'}$ denotes the expectation over independent pairs $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{x}', \boldsymbol{y}')$ drawn from the distribution of (X, Y). HSIC(X, Y) is equal to 0 if X and Y are independent, and is non-negative otherwise. Several estimators have been proposed to compute HSIC from observations $(\boldsymbol{g}_j, \boldsymbol{y})$ [86, 232], where $\boldsymbol{g}_j \in \mathbb{R}^n$ contains n observations for one feature and corresponds to a column of $\boldsymbol{X}$. These can be used to design statistical tests of independence [85] or to perform feature selection by ranking the features by descending value of HSIC [233].

1.3 Network biology

Biological systems are complex systems composed of many interacting entities. In particular, the molecular features we consider, whether genetic sequences, genes, or mutations, do not act in isolation. *Biological networks*, which are mathematical representations of these interactions as graphs, are an important tool to give context to these features and model their relationships. A large part of my work consists in developing ways of using these biological networks to guide biomarker discovery. In this section, I give some background on both biological networks and their mathematical modelization.

1.3.1 *Biological networks*

Biological systems are often represented as networks, which capture relationships or interactions between biological entities, such as genes, metabolites, or proteins. Examples include metabolic networks; cell signaling networks; gene regulatory networks; protein-protein interaction networks; disease-gene interaction networks, which connect diseases to genes that, when mutated, contribute to the disease; or drug-protein interaction networks, linking drugs to their protein target.

Gene-gene interaction networks can play an important role in biomarker discovery as they encode information about which regions of the genome "work" together, or against each other, towards a particular function. In particular, a genetic aberration can have a negative effect on the function of genes that have no mutation, but are connected to this mutated gene [17]. Conversely, the impact of a mutation can be negated by functional redundancy. In additions, proteins involved in the same disease tend to interact with each other [187], and genes linked to diseases with similar phenotypes tend to interact [80]. This suggests that a disease phenotype is rarely the consequence of a single genetic anomaly, but rather a perturbation of a whole network of interacting molecules [18, 75, 111].

Ressources for gene-gene interaction networks include the STRING database [240], which contains physical and functional interactions, both computationally predicted and experimentally confirmed, for over 2 000 organisms, or BioGRID [36], which includes interactions, chemical associations, and post-translational modifications from the literature. In addition, systems biologists are building specialized networks, focused on the pathways involved in a particular disease. One example of such networks is ACSN [130], a comprehensive map of known molecular mechanisms implicated in cancer.

1.3.2 *Network science*

Networks, or *graphs*, have attracted considerable attention in the data mining and machine learning communities. Beyond biological systems, they may represent chemical compounds, ecological systems, functional connectivity in the brain, or social networks,

both on and off the web. Being able to manipulate these objects and, in particular, to determine which part of such systems is responsible for a particular outcome, is a modeling question that does not concern only biomarker discovery. For example, neuroscientists search for subgraphs in brain connectivity networks from functional MRI screens that correlate with certain types of behavior or cognitive tasks [1, 167].

In what follows, I define a few terms commonly used in network science that I will use throughout this document.

Graph / Network. A graph (network) $(\mathcal{V}, \mathcal{E})$ consists of a set of *vertices* (*nodes*) $\mathcal{V}$ and a set of *edges* (*links*) $\mathcal{E}$ made of pairs of vertices. If the pair is ordered, then the edge is *directed*; otherwise, it is *undirected*. A graph with no directed edge is called undirected; unless otherwise specified, this is the type of graph we consider here. We use the notation $p \sim q$ to denote that vertex p and vertex q form an edge in the graph considered.

Adjacency matrix. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, its *adjacency matrix* is a square matrix $\boldsymbol{W} \in \mathbb{R}^{d \times d}$, where $d = |\mathcal{V}|$ is the number of vertices, and $W_{pq} \neq 0$ if and only if there is an edge between the p-th and the q-th elements of $\mathcal{V}$. $W_{pq} \in \mathbb{R}$ represents the weight of edge (i, j). If all non-zero entries of $\boldsymbol{W}$ are equal to 1, the graph is said to be *unweighted*.

Network module. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a graph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is said to be a subgraph of $\mathcal{G}$ if and only if $\mathcal{V}'$ is a subset of $\mathcal{V}$ and $\mathcal{E}'$ is a subset of $\mathcal{E}$. In systems biology, the term *network module* refers to a subgraph of a biological network whose nodes work together to achieve a specific function. Examples of modules include transcriptional modules, which are sets of co-regulated genes that share a common function, or signaling pathways, that is to say chains of interacting proteins that propagate a signal through the cell. In the context of biomarker discovery, we are interested in finding modules of a given biological network that are associated with the phenotype under study.

Graph Laplacian. Given a graph $\mathcal{G}$ of adjacency matrix $\boldsymbol{W} \in \mathbb{R}^{d \times d}$, the Laplacian [165] of $\mathcal{G}$ is defined as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D}$ is the *degree matrix*, that is to say a $d \times d$ diagonal matrix with diagonal entries $D_{pp} = \sum_{q=1}^{d} W_{pq}$. The graph Laplacian is analog to the Laplacian operator in multivariable calculus, and similarly measures to what extent a graph differs at one vertex from its values at nearby vertices. Given a function $f : \mathcal{V} \mapsto \mathbb{R}$, $f^\top \boldsymbol{L} f$ quantifies how "smoothly" f varies over the graph [230].

1.4 Contributions

In this document, I will give an overview of my contributions to the field of machine learning for biomarker discovery. Those contributions follow three axes:

- The integration of prior biological knowledge, encoded as networks, to machine learning methods for biomarker discovery (Chapter 2);
- The development of multitask algorithms, which alleviate the data scarcity by jointly fitting models for related problems (Chapters 3 and 4);

- The development of methods for nonlinear feature selection, allowing the field of biomarker discovery to depart from single-feature or additive models (Chapters 5 to 7).

The work I present in this HDR thesis fall in the framework of biomarker discovery cast as a feature selection problem, but this is not the sole focus of my research. In Appendix A, the samples (and not the features) are genes, and the problem of biomarker discovery, or rather disease gene prioritization, is cast as a semi-supervised learning problems. In Appendix B, I discuss the applications of multitask learning to the prediction of drug-protein binding.

I will conclude with some lessons drawn not only from this work but, more generally, from my experience developing and applying machine learning to therapeutic research at large (Chapter 8), before sketching a few perspectives (Chapter 9) for my work.

2

NETWORK-GUIDED BIOMARKER DISCOVERY

One way to address the limited power of classic feature selection methods on high-dimensional biological data sets is to incorporate prior biological knowledge to the procedure. Because genes do not work in isolation, but rather cooperate through their interaction (physical, regulatory, or through co-expression) in cellular pathways and molecular networks, this prior knowledge is often available in a structured way, and in particular under the form of networks (see Section 1.3.1).

These gene-gene interaction networks can be used to define networks between genomic descriptors, by mapping these descriptors to genes, using for instance in the case of SNPs a fixed-size window over the genetic sequence, and connecting together all descriptors mapped to the same gene, and all descriptors mapped to either of two interacting genes (see Section 2.3.7).

In this chapter, I make the assumption that genetic features that are linked on such a network are more likely to work jointly towards explaining the phenotype of interest, and that such effects would otherwise be missed when considering them individually. Compared to pathway-based approaches, which assess whether predefined sets of genes are associated with a given trait, network-based approaches introduce flexibility in the definition of associated gene sets.

In what follows, I will review three families of approaches, namely post-hoc analyses (Section 2.1), regularized regression (Section 2.2), and penalized relevance (Section 2.3). This last family of approaches is one we proposed in [11] with the SConES (Selecting Connected Explanatory SNPs) algorithm, and I will also summarize some experimental results we have obtained with this method (Section 2.4).

Appendix A, in which I describe how to cast the problem of disease-gene prediction as a graph node labeling problem, review existing methods and describe how to use recent graph neural network approaches, also relies on this assumption.

The contents of this chapter have been published in

- Chloé-Agathe Azencott, Dominik Grimm, Mahito Sugiyama, Yoshinobu Kawahara, and Karsten M. Borgwardt. Efficient network-guided multi-locus association mapping with graph cuts. *Bioinformatics*, 29(13):i171–i179, 2013. Proceedings of the 21st Annual International Conference on Intelligent Systems for Molecular Biology (ISMB 2013).
- Chloé-Agathe Azencott. Network-guided biomarker discovery. In *Machine Learning for Health Informatics*, number 9605 in Lecture Notes in Computer Science. Springer, 2016.

2.1 Network-based post-analysis of association studies

Most of the methods developed to integrate biological networks to the analysis of GWAS results are post-hoc methods. They start from a classic, single-SNP GWAS, to obtain the association of each SNP with the phenotype of interest. The SNP p-values are then converted to gene p-values, These p-values are assigned to the nodes of an existing biological network. The goal of network-based post-analysis is to find modules of this network that concentrate more small p-values than would be expected by chance.

These methods can therefore leverage state-of-the-art statistical tests that, for example, account for sample relatedness [245], address issues related to correlation between markers (linkage disequilibrium) [148], or are tailored to the discovery of rare variants [138]. In addition, they can easily be applied without access to raw data, only on the basis of published summary statistics, which makes them particularly appealing.

Summarizing SNP statistics into gene statistics requires to first map SNPs to genes. This is typically achieved by *physical mapping*, that is to say, based on distance to the gene on the genomic sequence. Gene p-values are then obtained using the minimum, maximum, or average p-value. A popular alternative consists in using VEGAS, which accounts for linkage disequilibrium between markers [147].

Several search methods have been proposed to find modules of significantly associated genes from the resulting networks. dmGWAS [112] uses a greedy approach [42] to identify modules that locally maximize the proportion of low p-value genes. Several variants of this approach, using different greedy statistics, have been proposed. A prominent example, first proposed in [19] and refined in PINBPA [260], relies on a simulated annealing search called JActiveModule and first proposed for the discovery of regulatory pathways in protein-protein interaction networks [107]. Finally, GrandPrixFixe [242] uses a genetic algorithm for its search strategy.

Because exact searches are prohibitively expensive in terms of calculations, these approaches rely on heuristic searches that do not guarantee that the top-scoring module is found. Methods such as that proposed by Mitra et al. [169] could be used to identify top-scoring modules exactly, but are too computationally intensive to have been applied to GWAS at this point. Computational cost also limit the application of these post-hoc methods to networks defined over genes rather than directly over biomarkers.

2.2 Regularized linear regression

Rather than considering each SNP individually and then trying to combine evidences through a biological network, the regularized linear regression framework (see Section 1.2.4) allows to consider all SNPs joinlty in a linear model.

Many regularizers have been proposed, to satisfy a variety of constraints on the regression weights, and have led to many contributions for the analysis of GWAS data [39, 225, 269, 285, 287]. In particular, it is possible to design regularizers that force the features that are assigned non-zero weights to follow a given underlying structure [106, 166]. In the context of network-guided biomarker discovery, we will focus on regularizers $\Omega(\boldsymbol{\beta})$ that penalize solutions in which the selected features are not connected over a given network.

These regularizers include the *overlapping group lasso* [108], which encourages the selection of biomarkers belonging to the same group (or set) of features. This method can be applied to network-guided biomarker discovery if each network edge defines a group of two biomarkers.

Another example is the *generalized fused lasso* [248], which smoothes regression weights along the edges of the graph thanks to a term in $\sum_{p\sim q} |\beta_p - \beta_q|$, where $p \sim q$ denotes that p and q are connected on the graph. I am not aware of any application of this approach to biomarker discovery from genetic data, but [272] successfully applied it to Alzheimer's disease diagnostic from brain images.

Alternatively, based on work on regularization operators [230], *Grace* [140, 141] uses a penalty based on the graph Laplacian L of the biological network, which encourages the coefficients $\boldsymbol{\beta}$ to be smooth on the graph structure. This regularizer is given by Eq. (2.1), and yields a special case of the generalized elastic net [231]. It penalizes coefficient vectors $\boldsymbol{\beta}$ that vary a lot over nodes that are linked in the network.

$$\Omega_{\text{grace}}(\boldsymbol{\beta}) = \boldsymbol{\beta}^\top L \boldsymbol{\beta} + \mu ||\boldsymbol{\beta}||_1 = \sum_{p,q} W_{pq}(\beta_p - \beta_q)^2 + \mu \sum_p |\beta_p|. \tag{2.1}$$

Here one assumes the weights along the edges of the adjacency matrix are all positive.

Finally, while the previous approaches require to build a network over biomarkers, the *graph-guided group lasso* [263] encourages genes connected on the network to be selected in and out of the model together (graph penalty), and biomarkers attached to a given gene to be either selected together or not at all (group penalty).

In practice, we found that the computational burden was a severe limitation to applying either the overlapping group gasso or Grace to the analysis of more than a hundred thousand markers [11]. On a similar note, the experiments presented in Yang et al. [279] used at most 8 000 genes; the graph-guided group lasso Wang and Montana [263] used 1 000 SNPs only; and the work in Xin et al. [272] used 3 000 voxels to describe brain images. It is therefore unclear whether these methods can scale up to several hundreds of thousands of markers.

While these computational issues might be addressed by using more powerful solvers or parallel versions of the algorithms, regularized linear regression approaches are also typically highly unstable (Section 1.2.5). Empirically, structural regularizers can help alleviate this issue, but rather partially to date.

Finally, it is interesting to note that biomarkers are often represented as categorical variables (such as the presence or absence of a mutation, or the number of minor alleles observed in the case of SNPs). Applying linear (or logistic) regressions in this context, although not entirely meaningless, can be considered an unsatisfying choice, unless one uses a one-hot encoding vector, which increases the number of variables.

2.3 Penalized relevance

By contrast, we proposed in [11] to combine statistical tests with regularization. The resulting method, which we called SConES for Selecting Connected Explanatory SNPs, can be seen as a specific instance of a more general framework that I call *penalized relevance*. SConES is based on a minimum cut reformulation of the problem of selecting features

under sparsity and connectivity constraints, which can be solved exactly and rapidly. It is therefore an efficient method to discover sets of genetic loci that are maximally associated with a phenotype, while being connected in an underlying network.

2.3.1 *General framework*

Let us assume that the data is described over a set $\mathcal{V}$ of m features. We propose to carry out feature selection by identifying the subset $\mathcal{S}$ of $\mathcal{V}$ that maximizes the sum of a data-driven *relevance function* and a domain-driven *regularizer*.

The relevance function $R : 2^{\mathcal{V}} \to \mathbb{R}$ quantifies the importance of a set of features with respect to the task under study. It can be derived from a measure of correlation, or a statistical test of association between groups of features and a phenotype.

Our objective is to find the set of features $\mathcal{S} \subseteq \mathcal{V}$ that maximizes R under structural constraints, which we model, as previously, by means of a regularizer $\Phi : 2^{\mathcal{V}} \to \mathbb{R}$, which promotes sparsity patterns that are compatible with a priori knowledge about the feature space. A simple example of regularizer computes the cardinality of the selected set. We hence want to solve the following problem:

$$\arg\max_{\mathcal{S} \subseteq \mathcal{V}} R(\mathcal{S}) - \lambda \Phi(\mathcal{S}). \tag{2.2}$$

Here again, $\lambda \in \mathbb{R}^+$ is a parameter which controls the balance between the relevance and the regularization terms.

This formulation is close to that of the regularized linear regression presented in Section 1.2.4. However, lasso-like approaches focus on the minimization of an empirical risk (or prediction error), while the penalized relevance framework shifts the emphasis to the maximization of feature importance with respect to the question under study. As with the approaches presented in Section 2.1, this formulation makes it possible to leverage a large body of work from statistical genetics to define relevance based on appropriate statistical tests. Moreover, in this framework, optimization is done directly over the power set of $\mathcal{V}$ (also noted as $2^{\mathcal{V}}$), rather than over $\mathbb{R}^m$. This presents the conceptual advantage of yielding sparsity formulations that can be optimized without resorting to convex relaxation.

Although convex optimization tends to be more efficient than combinatorial optimization, some choices of relevance and regularization result in better computational efficiency in very high dimension.

In particular, relevance functions derived from linear models are *modular*, meaning that the relevance of a set of biomarkers is computed as the sum of the relevances of the individual biomarkers in this set.

More specifically, given a set $\mathcal{V}$, a function $\Phi : 2^{\mathcal{V}} \to \mathbb{R}$ is said to be *submodular* if for any $\mathcal{S}, \mathcal{T} \subseteq \mathcal{V}$, $\Phi(\mathcal{S}) + \phi(\mathcal{T}) \geq \Phi(\mathcal{S} \cup \mathcal{T}) + \Phi(\mathcal{S} \cap \mathcal{T})$. This property is also referred to as that of diminishing returns. Given a graph G and its adjacency matrix W, an example of submodular function is the function $\Phi : \mathcal{S} \mapsto \sum_{p \in \mathcal{S}} \sum_{q \notin \mathcal{S}} W_{pq}$. In the case of equality, i.e. $\Phi(\mathcal{S}) + \phi(\mathcal{T}) = \Phi(\mathcal{S} \cup \mathcal{T}) + \Phi(\mathcal{S} \cap \mathcal{T})$ for any $\mathcal{S}, \mathcal{T} \subseteq \mathcal{V}$, Φ is said to be *modular*. In this case, the value of Φ over a set is equal to the sum of its values over items of that set.

Submodular functions play an important role in optimization [74] and machine learning [15]. In particular, a number of submodular, structure-enforcing regularizers can be derived from sparsity-inducing norms [14]. As the sum of submodular functions is

submodular, if R is modular and Φ submodular, solving Eq. (2.2) becomes a *submodular minimization problem* and can be solved in polynomial time.

Unfortunately, algorithms to minimize arbitrary submodular functions are slow, with a computational complexity in $\mathcal{O}(m^5c + m^6)$ where c is the cost of one function evaluation [186].

However, faster algorithms exist for specific classes of submodular functions. In particular, *graph cut functions* can be minimized much more efficiently in practice with maximum flow approaches [83], a particularity that has long been exploited in the context of energy minimization in computer vision [125].

2.3.2 *SConES: Selecting Connected Explanatory SNPs*

One of the submodular, structure-enforcing regularizers that can be derived from sparsity-inducing norms [14] is the Laplacian-based graph regularizer, which encourages the selected features to be connected on a predefined graph defined by its adjacency matrix $\boldsymbol{W}$. It is very similar to Ω_{grace} in Eq. (2.1), and given by

$$\Phi_{\text{Laplacian}} : \mathcal{S} \mapsto \sum_{p \in \mathcal{S}} \sum_{q \notin \mathcal{S}} W_{pq}. \tag{2.3}$$

In order to perform network-guided feature selection in the penalized relevance context, we propose to use SKAT with a linear kernel (see Section 1.2.6) for R, and Φ by the sum of a cardinality constraint $\eta|\mathcal{S}|$ and the Laplacian-based regularizer $\Phi_{\text{Laplacian}}$ defined above. The SKAT test statistic gives us a modular relevance function, that is to say, for any $\mathcal{S} \subseteq \mathcal{V}$ $R(\mathcal{S}) = \sum_{p \in \mathcal{S}} R(\{p\})$ We therefore obtain the optimization problem given by Eq. (2.4):

$$\underset{\mathcal{S} \subseteq \mathcal{V}}{\arg\max} \sum_{p \in \mathcal{S}} R(\{p\}) - \eta|\mathcal{S}| - \lambda \sum_{p \in \mathcal{S}} \sum_{q \notin \mathcal{S}} W_{pq}. \tag{2.4}$$

2.3.3 *Maximum flow solution*

The submodular minimization problem in Eq. (2.4) can be cast as a graph-cut problem and solved very efficiently. Indeed, we showed in Azencott et al. [11] that it is equivalent to finding an s/t min-cut on the graph, depicted in Figure 2.1, whose vertices are that of $\mathcal{G}$, augmented by two additional nodes s and t, and whose edges are given by the adjacency matrix $\boldsymbol{A}$, where $A_{pq} = \lambda W_{pq}$ for $1 \leq p, q \leq m$ and

$$A_{sp} = \begin{cases} R(\{p\}) - \eta & \text{if } R(\{p\}) > \eta \\ 0 & \text{otherwise} \end{cases} \quad \text{and } A_{tp} = \begin{cases} \eta - R(\{p\}) & \text{if } R(\{p\}) < \eta \\ 0 & \text{otherwise} \end{cases}$$
$$(p = 1, \ldots, m).$$

This also holds if $\boldsymbol{W}$ is a weighted adjacency matrix, and therefore the min-cut reformulation can also be applied to a weighted network. The original graph $\mathcal{G}$ can be directed or undirected.

It is therefore possible to use maximal flow algorithms to efficiently optimize the objective function defined in Eq. (2.4) and select a small number of connected SNPs maximally

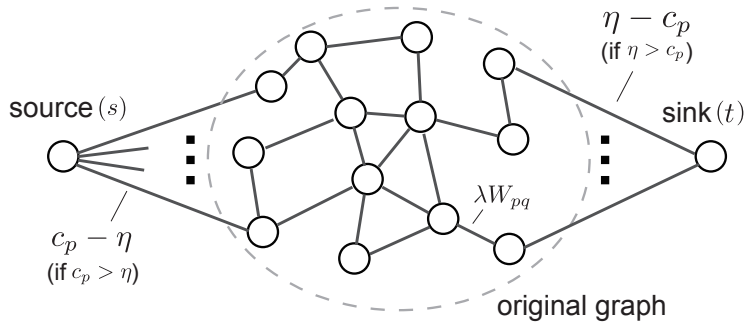


Figure 2.1: Graph for which finding the minimum cut is equivalent to maximizing the objective function in Eq. (2.4). $c_p = R(\{p\})$ denotes the relevance of biomarker p, and W_{pq} is the weight of the edge connecting biomarker p to biomarker q in the given network.

associated with a phenotype. In our implementation, we use the Boykov-Kolmogorov algorithm [27]. Although its worst case complexity is in $\mathcal{O}(n^2 n_E n_C)$, where n_E is the number of edges of the graph and n_C the size of the minimum cut, it performs much better in practice, particularly when the graph is sparse. We refer to this method as SConES, for Selecting CONnected Explanatory SNPs.

SConES is available as a Matlab implementation[1], as well as part of the `sfan` Python package[2] and the Bioconducter package `martini` [45, 47].

2.3.4 *Spectral formulation*

Let us denote by $\boldsymbol{f} \in \{0,1\}^m$ the indicator vector of a subset $\mathcal{S} \subset \mathcal{V}$: f_p is set to 1 if $p \in \mathcal{S}$ and 0 otherwise. We also denote by $\boldsymbol{r} \in \mathbb{R}^{|\mathcal{V}|}$ the vector composed of values $R(\{p\})$.

Eq. (2.4) can be rewritten using the *Laplacian* $\boldsymbol{L}$ of the network: the relevance term $\sum_{p\in\mathcal{S}} R(\{p\})$ can be written as $\boldsymbol{r}^\top \boldsymbol{f}$, while $\eta|\mathcal{S}| = ||\boldsymbol{f}||_0$ and $\sum_{p\in\mathcal{S}} \sum_{q\notin\mathcal{S}} W_{pq} = \boldsymbol{f}^\top \boldsymbol{L} \boldsymbol{f}$. This term is equivalent to the Laplacian graph regularizer used in Grace (Eq. (2.1)). Hence Eq. (2.4) is equivalent to:

$$\underset{\boldsymbol{f}\in\{0,1\}^m}{\arg\max}\, \boldsymbol{r}^\top \boldsymbol{f} - \eta||\boldsymbol{f}||_0 - \lambda \boldsymbol{f}^\top \boldsymbol{L} \boldsymbol{f}. \tag{2.5}$$

Note how this formulation uses directly the ℓ_0 norm, without relaxing it to ℓ_1.

2.3.5 *Setting the hyperparameters*

As for regularized linear regression, we propose to set the regularization hyperparameters λ and η through cross-validation grid-search experiments. As stability is a concern, rather than selecting the hyperparameters leading to the best predictive model, we pick as optimal the parameters leading to the most stable selection according to the consistency index defined in Section 1.2.5, and report as finally selected the features selected in all folds. Finding a good balance between predictivity and stability, however, can prove difficult, as approaches that systematically select all (or none) of the features will have high stability but poor predictivity – and little interest. Upper bounding the number of features that can be selected and excluding solutions that select no features allows us to address this problem to some extent.

1 `https://github.com/chagaz/scones`
2 `https://github.com/chagaz/sfan`

In addition, regularization parameter η is *anti-monotonous* with respect to the number of selected features: if we denote the selected features for each η by $\mathcal{S}(\eta)$, we have $\mathcal{S}(\eta) \subseteq \mathcal{S}(\eta')$ if and only if $\eta > \eta'$.

Moreover, we can easily check that our formulation satisfies all assumptions to apply the *parametric maximum flow algorithm* [76]. With this algorithm, we can obtain the entire *regularization path* [96] along with the changes in η without increasing the time complexity.

In practice, this property of η is particularly interesting when we are given cardinality constraints *a priori* over the size of the set of selected features. Then we can directly pick from the regularization path the solutions that fulfill these constraints.

2.3.6 *Group penalties*

Rather than using biological networks to define constraints, one may wish to use predefined sets (or groups) of features, such as regulatory or metabolic pathways. The overlapping group lasso [108], mentioned in Section 2.2, was developed for this purpose. Given a set $\mathcal{P} = \{P_1, P_2, \ldots, P_r\}$ of such groups of features, the *coverage regularizer* defined in Eq. (2.6) below encourages the selected features to belong to a small number of the groups of $\mathcal{P}$.

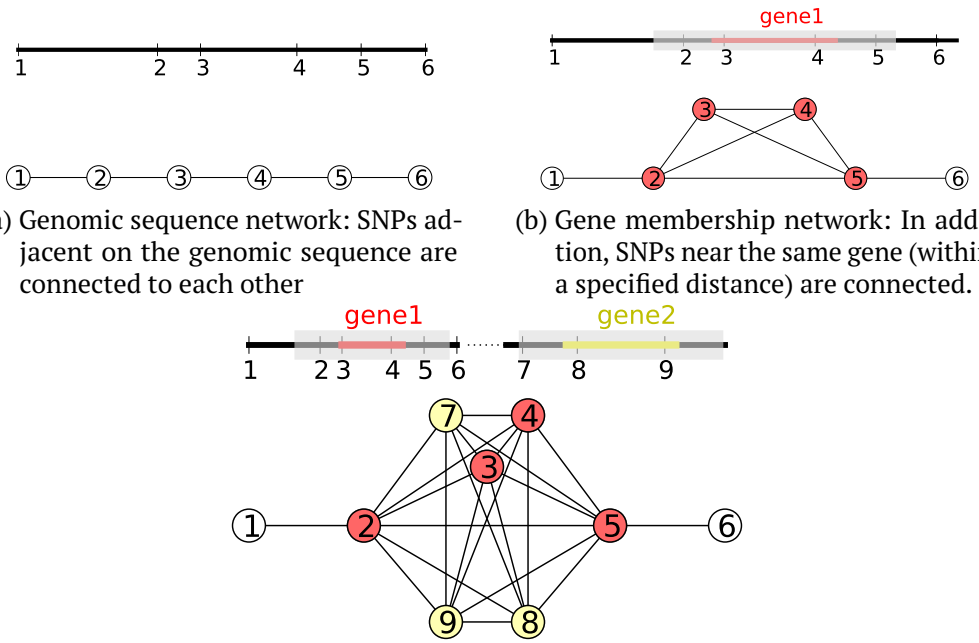
$$\Phi : \mathcal{S} \mapsto \sum_{P \in \mathcal{P}, P \cap \mathcal{S} \neq \varnothing} w_P, \tag{2.6}$$

where $w_P \in \mathbb{R}^+$ denotes a predefined weight associated with group P. This regularizer can once more be combined with a cardinality constraint $\eta|\mathcal{S}|$, and like the one used in SConES, is a cut function, and therefore amenable to a similar maximum flow reformulation.

2.3.7 *Building SNP-SNP networks from gene-gene networks*

Most efforts to build biological networks are targeted towards the construction of gene-gene networks, where nodes correspond to genes and links correspond to a relationship between those gene, such as co-expression in a specific tissue, or physical interaction of the proteins they code for. Building SNP-SNP networks from gene-gene networks is not straightforward. In most of my work, I have used the three following approaches, schematically explained on Figure 2.2:

- *Genomic sequence network* (GS): SNPs adjacent on the genomic sequence are linked together. In this setting we aim at recovering sub-sequences of the genomic sequence that correlate with the phenotype.
- *Gene membership network* (GM): SNPs are connected as in the sequence network described above; in addition, SNPs near the same gene are linked together as well. Usually, a SNP is considered to belong to a gene if it is either located inside said gene ore within a pre-defined distance of this gene. In this setting we aim more particularly at recovering genes that correlate with the phenotype.
- *Gene interaction network* (GI): SNPs are connected as in the gene membership network described above. In addition, supposing we have a gene-gene interaction network (derived, for example, from protein-protein interaction data or gene expression correlations), SNPs belonging to two genes connected in the gene network are linked together. In this setting, we aim at recovering potential pathways that explain the phenotype.



(a) Genomic sequence network: SNPs adjacent on the genomic sequence are connected to each other

(b) Gene membership network: In addition, SNPs near the same gene (within a specified distance) are connected.

(c) Gene-interaction network: In addition, SNPs near two interacting genes are connected.

Figure 2.2: Small examples of the three types of SNP networks that can be built.

We are currently considering additional ways of building SNP-SNP networks based on gene-gene networks, based on different SNP-to-gene mappings. One of those is based on known eQTLs (expression quantitative trait loci, which link SNPs to genes whose expression they partially regulate), and the other on chromatin interaction matrices, which allow to link SNPs to genes they are physically in contact with in live cells. While neither type of information was available in data bases at the time we developed SConES, this has now changed and both approaches have been used for the functional interpretation of GWAS results [264].

2.4 Experimental results

I will now briefly summarize the results of our experiments applying SConES to both simulated and real-world data.

2.4.1 *Runtime of SConES*

We evaluated the runtime of SConES, for a number of simulated scenarios, over a single CPU over a single AMD Opteron CPU (2 048 KB, 2 600 MHz) with 512 GB of memory. Our findings indicate that SConES is typically one order of magnitude faster than the overlapping group lasso [108] (see Section 2.2) and two orders of magnitude faster than Grace [140] (see Eq. (2.1)). We were able to implement a faster version of Grace (see Section 3.2), but it required more memory and could not run for more than 150 000 SNPs.

2.4.2 *Ability of SConES to recover relevant features*

On our simulations, SConES was systematically better than its state-of-the-art comparison partners at leveraging structural information to retrieve the connected SNPs that were

causal. The performance of SConES, as that of other graph-regularized approaches, is strongly negatively affected when the network is entirely unrelated to the problem.

We also applied SConES to a large collection of 17 *Arabidopsis thaliana* flowering times phenotypes from Atwell et al. [9] (up to 194 individuals, 214 051 SNPs). The SNPs selected by SConES are at least as informative (in a ridge regression model) as those selected by other graph-guided approaches. For the phenotypes for which the lasso outperforms SConES, it also outperforms all other graph-guided approaches, suggesting that the network information is irrelevant in these cases. Finally, all graph-guided approaches retrieve similar numbers of SNPs, but the SNPs identified by SConES tend to be more spread out in the network, and to cover a larger number of genes previously known or suspected to be associated with flowering time (as listed in Brachi et al. [29]).

2.4.3 *Robustness to missing edges*

Furthermore, we observe that removing a small fraction (1–15%) of the edges between causal features does not significantly alter the performance of SConES. This means that SConES is robust to missing edges, an important point when the biological network used is likely to be incomplete.

2.4.4 *Comparison of network-guided biomarker discovery approaches*

We are currently comparing the application of various network-guided biomarker discovery approaches to a breast cancer GWAS data set, composed of 1 282 French women affected by familial breast cancer, and 1 272 unaffected women from the general population.

Our preliminary results [48] indicate that network methods find SNPs close to genes well known to be related to breast cancer susceptibility, such as FGFR2, TOX3, or NEK10. With two orders of magnitudes fewer samples, network methods recover 23% of the SNPs found significant in the most exhaustive breast cancer susceptibility meta-analysis to date.

We also found that although the multiplicity of subnetworks returned by PINBPA and dmGWAS makes interpretation more challenging, it provides useful complementary information. We conclude that dmGWAS and PINBPA are interesting for generating hypotheses about the etiology of the disease, while SConES' strength is biomarker discovery.

2.5 Conclusion and perspectives

We can hardly hope to understand the biology underlying complex diseases without considering the molecular interactions that govern entire cells, tissues or organisms. The approaches we discussed offer a principled way to perform biomarker discovery in a systems biology framework, by integrating knowledge accumulated in the form of interaction networks into studies associating genomic features with a disease or response to treatment. While these methods are still in their infancy, I believe that they can become powerful tools in the realization of precision medicine.

The approaches I have discussed in this chapter are limited by (1) not accounting for nonlinear effects between network nodes; (2) the lack of statistical techniques for the evaluation of the significance of the associations they detect; and (3) the refinement and choice of appropriate network data.

While most network-guided biomarker discovery studies make use of generic gene-gene interaction networks such as STRING or BioGRID, many other possibilities are starting to open up. They include disease-specific networks such as ACSN, but we can also imagine using for example eQTL networks based on previous studies [250], or three-dimensional chromatin interaction networks [213]. Methods that integrate these multiple types of networks may be needed; that the regularized regression or penalized relevance methods we discussed can all accomodate weighted networks (either directly or through simple modifications) that will facilitate these developments.

The Laplacian regularizer was originally designed to smooth real-valued feature weights along the graph. In the regularized relevance framework, it has the undesirable tendency to essentially want to grab all the neighbors of a selected node. There is therefore a need to develop more appropriate graph regularizers, based for example on approximations to the number of connected components formed by the selected features [37] or on random walks [230]. By penalizing disconnections between faraway vertices more than between closeby ones, regularizers based on random walks will provide an elegant way to compensate for missing edges in the graph.

Finally, no serious progress the field of biomarker discovery can be made without proper validation, at the very least in different data sets pertaining to the same trait, of the pertinence of the modules identified by these various methods. Because this often requires that modelers convince the owners of other data sets to run experiments for their own benefits, this is often hard to implement outside of large consortium collaborations.

3

MULTITASK NETWORK-GUIDED BIOMARKER DISCOVERY

Machine learning applications to biomarker discovery are severely limited by the scarcity of data to learn from. To alleviate the statistical challenges resulting from having many more features than samples to learn from, I have described in Chapter 2 how to integrate prior knowledge encoded as networks to feature selection algorithms, thereby reducing the space of possible solutions. To further address these challenges, the multitask learning framework proposes to jointly learn models for different but related tasks by sharing information between those tasks.

Multitask learning is driven by the assumption that there are benefits to be gained from jointly learning on related tasks. In the multitask framework, data is available for several related but different problems (or tasks). While such data cannot be pooled together to form a single, large data set, the idea is to leverage all the available information to jointly learn related but separate models for each of the tasks.

A number of biomarker discovery settings lend themselves well to this approach. Examples include GWAS over multiple related traits (such as ovarian and breast cancers, or asthma and chronic obstructive pulmonary disease); studying the response of patients to different drugs, as was done in the DREAM Rheumatoid Arthritis Responder Challenge in which I took part [223]; or toxicogenomics, where one studies the response of cell lines to exposure to various chemicals, as was done in the DREAM Toxicogenetics Challenge in which I also took part [63].

In each of those settings, multitask feature selection approaches reduce the features-to-sample ratio of the data, while keeping the particularities of each data set. In the two following chapters, I will describe several contributions to multitask feature selection: In this chapter, I will present Multi-SConES, a multitask extension of the SConES algorithm (Section 2.3); in the following, I will make the assumption that task descriptors are available to further constrain the problem.

This chapter starts with a brief overview of existing methods for multitask feature selection with structured regularizers (Sections 3.1 and 3.2). I will then describe in Section 3.3 the Multi-SConES algorithm and explain how it can be solved exactly and efficiently by a maximum flow algorithm. Finally, in Section 3.4, I will briefly summarize experimental results on both simulated and real data, showing that MultiSconES is better at recovering causal features than other state-of-the-art methods.

This chapter is based on work published as

> Mahito Sugiyama, Chloé-Agathe Azencott, Dominik Grimm, Yoshinobu Kawahara, and Karsten M. Borgwardt. Multi-task feature selection on multiple networks via maximum flows. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 199–207, 2014.

3.1 Multitask feature selection with structured regularizers

Several multitask versions of the lasso have been proposed, starting with the multitask lasso [184], in which an ℓ_2-norm on each weight across all tasks rewards solutions where the same features are selected for all tasks.

The graph-guided fused lasso [119] extends this idea by coupling the weight vectors of correlated tasks: solutions in which correlated tasks have similar weight vectors are rewarded. Chen et al. [40] further extends this to include a network regularization on the input features similar to that of the generalized fused lasso 2.2. Along the same lines, Lee and Xing [139] uses mixed ℓ_1/ℓ_2-norm to enforce the selection of either none or all of the features in predefined groups of correlated features, for predefined groups of correlated tasks. Finally, [71] uses network regularization to infer task relationship in multitask learning.

There has been little interest, however, in focusing on network-regularized multitask approaches where the network information describes feature relationships only. For this reason, we proposed a simple multitask extension of Grace [140], which we refer to as Multi-Grace and which is described in Section 3.2.

Furthermore, most existing approaches assume that the same features should be selected across all task. While this is reasonable for many application domains, one can think of numerous contexts where this assumption is violated. For example, lung diseases such as asthma and chronic obstructive pulmonary disease (COPD) may be linked to a set of common mutations, but there is no indication that the exact same mutations are causal in both diseases. Existing multitask approaches based on regularized linear regression will attempt to select the union of both sets of relevant mutations, and may or may not assign a weight of zero to a feature that is specific to one task in the models of other tasks.

In addition, multitask approaches that incorporate structured regularizers do not make it possible to consider different structural constraints for different tasks. However, we may want to use different biological networks for different related diseases. For example, one may want to study ovarian and breast cancers simultaneously using a tissue-specific co-expression network for each of these phenotypes.

To address these issues, as well as to benefit from the advantages of the penalized relevance framework, we proposed Multi-SConES, a multitask formulation of SConES. Multi-SConES uses multiple network regularizers to improve feature selection in each task by combining and solving multiple tasks simultaneously.

3.2 Multi-Grace

Given data described as $(\boldsymbol{X}, \boldsymbol{Y}) \in \mathbb{R}^{n \times m} \times \mathbb{R}^{n \times T}$, representing m features for n samples and the T associated outputs, the multitask lasso [184] solves

$$\underset{\boldsymbol{B} \in \mathbb{R}^{m \times T}}{\arg\min} \sum_{t=1}^{T} ||\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{y}_t||_2^2 + \lambda ||\boldsymbol{B}||_{\ell_1/\ell_2}, \tag{3.1}$$

where the t-th column of $\boldsymbol{B}$ is the parameter vector $\boldsymbol{\beta}_t$ of the corresponding task, and the ℓ_1/ℓ_2-norm of $\boldsymbol{B}$ is given by $||\boldsymbol{B}||_{\ell_1/\ell_2} = \sum_{p=1}^{m} ||\boldsymbol{\beta}_p||_2$. Note that we are here in the *multi-output* settings, where all outputs (here, phenotypes) are available for each sample.

Given a network over the m features, described by its graph Laplacian $\boldsymbol{L}$, we formulate Multi-Grace as:

$$\underset{\boldsymbol{B}\in\mathbb{R}^{m\times T}}{\arg\min} \sum_{t=1}^{T} \left(||\boldsymbol{X}\boldsymbol{\beta}_t - \boldsymbol{y}_t||_2^2 + \lambda_1 ||\boldsymbol{B}||_{\ell_1/\ell_2} + \lambda_2 \boldsymbol{\beta}_t^\top \boldsymbol{L} \boldsymbol{\beta}_t \right). \tag{3.2}$$

Following the reasoning in Lemma 1 of Li and Li [140], this formulation is equivalent to the followig multitask lasso problem:

$$\underset{\boldsymbol{B}\in\mathbb{R}^{m\times T}}{\arg\min} \sum_{t=1}^{T} \left(||\boldsymbol{X}^*\boldsymbol{\beta}_t - \boldsymbol{y}_t^*||_2^2 + \gamma ||\boldsymbol{B}^*||_{\ell_1/\ell_2} \right), \tag{3.3}$$

where $\gamma = \lambda_1/\sqrt{1+\lambda_2}$ and, for each task, $(\boldsymbol{X}^*, \boldsymbol{y}_t^*)$ is obtained as

$$\boldsymbol{X}^* = (1+\lambda_2)^{-1/2} \begin{pmatrix} \boldsymbol{X} \\ \sqrt{\lambda_2}\boldsymbol{S}^\top \end{pmatrix}, \boldsymbol{y}_t^* = \begin{pmatrix} \boldsymbol{y}_t \\ \mathbf{0_m} \end{pmatrix},$$

and $\boldsymbol{S}$ is such that $\boldsymbol{S}\boldsymbol{S}^\top = \boldsymbol{L}$. If $(\hat{\boldsymbol{\beta}}_t^*)_{t=1,\dots,T}$ is the solution to this multitask lasso problem, then the solution to Eq. (3.2) is given by $\hat{\boldsymbol{\beta}}_t = \hat{\boldsymbol{\beta}}_t^*/\sqrt{1+\lambda_2}$.

If the different tasks use different networks, this derivation does not apply and solving a multitask version of Grace is not straightforward any more.

Li and Li [140] proposed to use a singular value decomposition to obtain $\boldsymbol{S}$, but the Lemma also holds if $\boldsymbol{S}$ is replaced by the *incidence matrix* of the network. As it can be constructed in linear time in the number of vertices and edges, this makes for a much faster implementation of both Grace and Multi-Grace. We used this implementation in our experiments.

3.3 Multi-SConES

Multi-SConES is a generalized form of SConES, which achieves feature selection for multiple tasks simultaneously. In what follows, we assume the set of m features $\mathcal{V}$ is shared across all T tasks. For each task t, we assume a task-specific network $\mathcal{G}^t = (\mathcal{V}, \mathcal{E}^t)$ of adjacency matrix $\boldsymbol{W}^t$, and a task-specific relevance function $R^t : 2^{\mathcal{V}} \to \mathbb{R}$. Multi-SConES is then formulated as:

$$\underset{\mathcal{S}^1,\mathcal{S}^2,\dots,\mathcal{S}^T\subseteq\mathcal{V}}{\arg\max} \sum_{t=1}^{T} \left(\sum_{p\in\mathcal{S}^t} R^t(\{p\}) - \eta|\mathcal{S}^t| - \lambda \sum_{p\in\mathcal{S}^t} \sum_{q\in\mathcal{V}\setminus\mathcal{S}^t} W_{pq}^t \right) - \mu \sum_{t<u} |\mathcal{S}^t \triangle \mathcal{S}^u|, \tag{3.4}$$

where $\triangle$ denotes the symmetric difference between two sets, that is

$$\mathcal{S}\triangle\mathcal{S}' = (\mathcal{S}\cup\mathcal{S}') \setminus (\mathcal{S}\cap\mathcal{S}'). \tag{3.5}$$

Each term $\left(\sum_{p\in\mathcal{S}^t} R^t(\{p\}) - \eta|\mathcal{S}^t| - \lambda \sum_{p\in\mathcal{S}^t} \sum_{q\in\mathcal{V}\setminus\mathcal{S}^t} W_{pq}^t \right)$ corresponds to SConES formulated independently on each task (see Eq. (2.4)).
The additional penalty term $\mu \sum_{t<u} |\mathcal{S}^t \triangle \mathcal{S}^u|$ represents our belief that similar sets of features should be selected for all tasks. The larger μ, the more we enforce this belief. By contrast, if $\mu = 0$, solving Eq. (3.4) is equivalent to solving SConES on each task independently.

Eq. (3.4) can be reduced to a problem similar to Eq. (2.4), and therefore cast as a graph-cut problem and solved efficiently using a maximum flow algorithm. To do so, we create a *unified network* by replicating the vertices of each network G^t, thus obtaining $m \times T$ vertices and $\sum_{t=1}^{T} |\mathcal{E}^t|$ edges. In addition, we connect each pair of replicated vertices with an edge weight of μ/λ, thus obtaining an additional $m \times T(T-1)/2$ edges.

Figure 3.1 shows an example of unified network for $T = 2$.

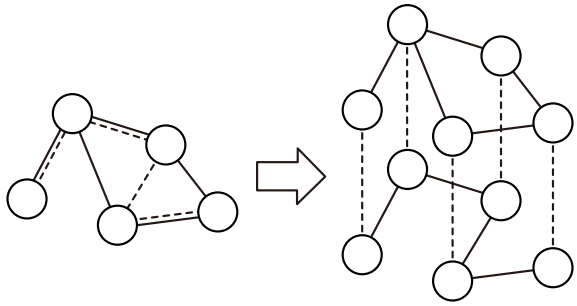


Figure 3.1: Example of two networks (left), which share vertices and have different edges (solid lines for the first and dotted lines for the second), and of the unified network (right), where vertices are duplicated and new edges (dotted lines) are added to connect duplicated vertices.

Let us call $\boldsymbol{W}$ the adjacency matrix of the unified network $(\mathcal{V}', \mathcal{E}')$. $\mathcal{V}'$ has size $m \times T$ and the relevance of its p-th element is given by the relevance of feature $(p \bmod m)$ in task $\lceil \frac{p}{m} \rceil$: $R(p) = R^{\lceil \frac{p}{m} \rceil}(\{p \bmod m\})$. Eq. (3.4) is then equivalent to

$$\underset{\mathcal{S} \subseteq \mathcal{V}'}{\arg\max} \sum_{p \in \mathcal{S}} R(p) - \eta|\mathcal{S}| - \lambda \sum_{p \in \mathcal{S}} \sum_{q \notin \mathcal{S}} W_{pq}, \tag{3.6}$$

and for each task t, the set of selected features is given by

$$\mathcal{S}^t = \{p \bmod m \,|\, p \in \mathcal{S}, tm \leq p < (t+1)m\}\,. \tag{3.7}$$

Eq. (3.6) can be solved exactly as Eq. (2.4). One limitation, however, is that the unified network grows in size with the number of tasks (linearly for the number of vertices, and quadratically for the number of edges). If the original number of features m is already quite large (of the order of 10^5 or more), this formulation becomes intractable for more than a few tasks ($T > 10$). Another important factor to note is that there are now *three* regularization parameters to select, η, λ, and μ, which also increases the computational complexity of running Multi-SConES in practice.

Multi-SConES can also be used when some features are missing for some tasks. In this case, the corresponding replicated vertex is ommitted from the network, as well as the edges connecting it either to other vertices in this task or to replications of the same feature in other tasks.

3.4 Experimental results

3.4.1 *Runtime*

Our empirical results on simulated data show that, under fixed regularization parameters and a fixed network architecture shared by all tasks, the runtime of Multi-SConES increases

cubically with the number of tasks. Nevertheless, Multi-SconES is still efficient enough to run over hundreds of thousands of features for a dozen of tasks, which matches many biomarker discovery setups, on a 2 $\times$ 3 GHz Quad-Core Intel Xeon CPU with 16 GB of RAM. As relevances are computed ahead of time, the number of samples does not matter here.

3.4.2 *Parameter sensitivity*

Our experimental results on simulated data also show that Multi-SConES is sensitive to η, which controls the sparsity of the solution, and more robust to both λ, controlling the connectivity of the network, and μ, which controls how similar the solutions should be between tasks. More specifically, if the solution is truly a set of connected subnetworks of the provided networks, then once λ is set high enough to get to select the correct solution, the corresponding penalty term becomes negligible.

As λ and μ do not need to be carefully tuned, and as the entire regularization path with respect to η can be obtained without increasing time complexity, this suggests that finding optimal hyperparameters for Multi-SConES is rather inexpensive. However, there are several limitations in practice. First of all, on real-world data, the solution of Multi-SConES is much more sensible to λ than on well-behaved simultated data. In addition, the regularization path with respect to η typically turns out to have one value per possible number of selected features, which is m, and can therefore be not so easy to manipulate in practice.

3.4.3 *Ability to recover causal features*

Our experiments on simulated data show that Multi-SConES outperforms both SConES performed on separate tasks independently, and state-of-the-art comparison partners Multi-Grace and multitask lasso on a variety of settings. The more features are shared between tasks, and the better Multi-SConES is at recovering causal features. This holds for all multitask approaches.

Our results on the same *Arabidopsis thaliana* flowering time GWAS data used for Chapter 2 show that combining several related phenotypes helps recovering proportionally more known or suspected flowering time genes, and that Multi-SConES retrieves more of these genes than its comparison partners. Altogether, our results suggest that Multi-SConES can be effectively employed for multilocus, multiphenotype biomarker discovery.

3.5 Conclusion and perspectives

As we have shown, the penalized relevance framework can be used to propose an efficient and effective algorithm for network-guided, multitask biomarker discovery. Compared to the classic regularized linear regressions with network regularizers, Multi-SConES shows better ability to discover relevant features in both simulated and real-world experiments.

As SConES, Multi-SConES directly optimizes feature relevance scores, rather than minimizing a squared error, and an ℓ_0 constraint that does not need to be relaxed to its ℓ_1 counterpart. The resulting optimization problem can be cast as a minimum cut problem and solved exactly and efficiently, even for hundreds of thousands of features, using maximum flow algorithms. Finally, it can easily incorporate cardinality constraints on the size of the selected features set.

Unlike the other existing methods, Multi-SConES can easily accomodate missing features as well as different networks for different tasks.

Currently, Multi-SConES models the relationship between tasks using a single parameter μ, which controls how coupled the sets of selected features should be. However, it is often possible to define similarities between tasks; for example, in the case of a pharmacogenomics screen, the similarity between phenotypes can be directly derived from the similarity between the drugs that were used. As an additional example, for biomarker discovery in cancer, integrative analyses have shown interesting similarities between cancer types [258].

While several existing structured regularized linear regression approaches make use of correlations between tasks [40, 139], Multi-SConES does not currently do so.

The extension is however possible. If one denotes by $\boldsymbol{\Sigma} \in \mathbb{R}^{T \times T}$ a matrix of similarities between tasks, and $\boldsymbol{E} \in \{0,1\}^{m \times T}$ the matrix of indicator vectors $\boldsymbol{e}^t$ for each task t (defined by $e^t_p = 1$ if $p \in \mathcal{S}^t$ and 0 otherwise), $\operatorname{tr}(\boldsymbol{E}\boldsymbol{\Sigma}^{-1}\boldsymbol{E}^\top)$ is a regularizer that enforces that the more similar the tasks, and the more similar their selected feature sets are. Eq. (3.4) then becomes

$$\underset{\mathcal{S}^1,\mathcal{S}^2,\ldots,\mathcal{S}^T \subseteq \mathcal{V}}{\arg\max} \sum_{t=1}^{T} \left(\sum_{p \in \mathcal{S}^t} R^t(\{p\}) - \eta|\mathcal{S}^t| - \lambda \sum_{p \in \mathcal{S}^t} \sum_{q \in \mathcal{V} \setminus \mathcal{S}^t} W^t_{pq} \right) - \mu \operatorname{tr}(\boldsymbol{E}\boldsymbol{\Sigma}^{-1}\boldsymbol{E}^\top). \quad (3.8)$$

If $\boldsymbol{\Sigma}$ is set to a matrix of all ones, therefore containing no information about task similarity, this formulation is equivalent to Eq. (3.4).

In the following chapter, I will describe how to construct task descriptors or task similarities when tasks are described by chemical compounds, as well as several additional contributions to the field of multitask biomarker discovery.

4

MULTITASK LEARNING WITH TASK DESCRIPTORS

Many multitask learning or feature selection approaches assume that the same features should be selected across all tasks. While the Multi-SConES approach I described in Chapter 3 allows some flexibility with respect to how coupled the selected feature sets should be, it does not take into account the degree of similarity between tasks. Intuitively, one would however like that more similar tasks share more features.

An additional limitation of existing multitask approaches is that they typically cannot be applied to make predictions for new tasks for which no training data is available. While it may seem preposterous to wish for such an ability, it could be useful, for example, to predict the cytotoxicity of a new drug on cells or patients.

To address these limitations, it is necessary to be able to define an explicit representation of the tasks. In many biomedical applications, these tasks can be linked to a molecular compound: in precision medicine, when a task corresponds to predicting the response of a patient to a drug; in chemogenomics, when a task corresponds to predicting which proteins are binding a given molecule; or in pharmacogenomics, when a task corresponds to predicting the toxicity of a compound for various cell lines.

In this chapter, I will therefore describe in Section 4.1 how to construct vectorial representations of small molecules. I will then illustrate how these representations can be used to perform lasso-like multitask feature selection with task descriptors (Section 4.2). These ideas can also be applied, without feature selection, to conduct efficient multitask drug-ligand binding prediction, as I describe in Appendix B.

The contents of this chapter are based on work published as

- Chloé-Agathe Azencott, Alexandre Ksikes, S. Joshua Swamidass, Jonathan H. Chen, Liva Ralaivola, and Pierre Baldi. One- to four-dimensional kernels for virtual screening and the prediction of physical, chemical and biological properties. *Journal of Chemical Information and Modeling*, 47(3):965–974, 2007.
- Víctor Bellón, Véronique Stoven, and Chloé-Agathe Azencott. Multitask feature selection with task descriptors. In *Pacific Symposium on Biocomputing*, volume 21, pages 261–272, 2016.

4.1 Vectorial representations of molecules

In many cases, a drug is an organic chemical compounds of low molecular weight which acts by binding a target protein and thus either inhibiting or enhencing its activity. For example, nonsteroidal anti-inflammatory drugs such as aspirin or ibuprofen inhibit cyclooxygenases COX-1 and COX-2, which are essential to the inflammatory response. Antibiotics from the pencillin family bind a group of proteins unsurprisingly called penicillin-binding proteins, which are essential to the synthesis of bacterial cell walls.

The development of rich and informative representations of organic chemical compounds of low molecular weight has therefore been a central question in *chemoinformatics*, or the application of computer science tools to questions from the field of chemistry, for many decades. The first step of my doctoral research has been to develop and study descriptors for such small molecules, systematically derived from structural representations of varying complexity. I have used these different representations to create kernels which I applied, in combination with support vector machines, to several classification and regression problems on chemical compounds. My findings indicate that two-dimensional representations derived from molecular graphs tend to be the most informative [12], and to this date most vectorial representations of small molecules are derived from their molecular graphs.

4.1.1 *Molecular graphs*

Small molecules are most commonly represented as labeled graphs of bonds. The vertices represent the atoms, and the edges represent the bonds. Edges are labeled by the bond type (e.g. single, double) they correspond to. Labels on the vertices correspond to the element (e.g. C, N, O) of the atom they correspond to, and can be expanded to include more information about the local chemical environment of the atom [12, 261].

For small molecules, these molecular graphs are fairly small, both in terms of the number of vertices and the number of edges. Indeed, valence rules constrain the average degree to be typically less than three.

4.1.2 *Path- and tree-based molecular fingerprints*

In chemoinformatics, molecules are traditionally represented using so-called fingerprints. A fingerprint is a binary vector in which each bit corresponds to a particular molecular feature, designed to be chemically relevant, and is turned to 1 if the molecule exhibits that feature and 0 otherwise. Whereas most sets of fingerprints or descriptors (from DRAGON descriptors [249] to MACCS keys [60]) are rather heterogeneous collections of all sorts of computable molecular properties that heavily rely on expert knowledge, substructure-based representations are derived in a more principled and automated way.

Several kinds of molecular graph substructures can be considered. The most commonly used are labeled paths of length up to d, starting at any vertex of the graph, and labeled trees of depth at most d, rooted at any vertex of the graph. The latter are usually known as circular fingerprints or extended-connectivity fingerprints (ECFP) [209]. The parameter d is usually set to $d = 6$ or $d = 8$ for paths, and $d = 2$ or $d = 3$ for trees. These values are chosen so that the resulting substructures can be discriminative when considered together, without being unique to a molecule of the data set.

Molecular fingerprints are long and sparse binary vectors: there are a lot of potential substructures of a given size containing all atoms appearing in organic compounds, but any given molecule only possesses a small fraction of them.

These binary fingerprints can be extended to count fingerprints, in which instead of the presence/absence of a particular substructure, each entry of the fingerprint records the number of occurrences of this substructure.

4.2 Multitask feature selection with task descriptors

To alleviate the statistical challenges due to data scarcity, multitask learning proposes to learn different but related tasks joinlty, rather than independently, by sharing information between these tasks. Within this framework, the joint regularization of model parameters proposed by Obozinski, Taskar, and Jordan [184] results in models with few non-zero coefficients and that share similar sparsity patterns.

In Bellon, Stoven, and Azencott [23], we proposed a new regularized multitask approach that incorporates task descriptors so as to modulate the amount of information shared between tasks according to their similarity. We showed on simulated data that this method outperforms other multitask feature selection approaches, particularly in the case of scarce data. In addition, we demonstrated on peptide MHC-I binding data the ability of the proposed approach to make predictions for new tasks for which no training data is available.

4.2.1 *Formulation*

Given T data sets, corresponding to T tasks, described as $(\boldsymbol{X}^t, \boldsymbol{y}^t) \in \mathbb{R}^{n_t \times m} \times \mathbb{R}^{n_t}$, representing m features for n_t samples and the associated outputs, the multitask lasso [184] solves Eq. (3.1), reproduced here:

$$\underset{\boldsymbol{B} \in \mathbb{R}^{m \times T}}{\arg\min} \sum_{t=1}^{T} \frac{1}{n_t} \sum_{i=1}^{n_t} \left(y_i^t - \sum_{p=1}^{m} \beta_{pt} x_{ip}^t \right)^2 + \lambda \sum_{p=1}^{m} ||\boldsymbol{\beta}_p||_2, \tag{4.1}$$

where the p-th column of $\boldsymbol{B}$ is the parameter vector $\boldsymbol{\beta}_p$ of regression coefficients for feature p across all T tasks. This sparse optimization problem can be solved using proximal optimization [179].

Multi-level multitask lasso To allow for more flexibility in the sparsity patterns of the different tasks, Swirszcz and Lozano [239] propose to decompose the regression parameter $\boldsymbol{B}$ into a product of two components $\boldsymbol{C} \in \mathbb{R}^{T \times m}$ and $\boldsymbol{\theta} \in \mathbb{R}^m$. The intuition here is to capture the global effect of the features across all the tasks with $\boldsymbol{\theta}$, while $\boldsymbol{C}$ provides a task-specific modulation. This results in the following optimization problem, known as the *multi-level multitask lasso*:

$$\underset{\boldsymbol{\theta} \in \mathbb{R}^m, \boldsymbol{C} \in \mathbb{R}^{m \times T}}{\arg\min} \sum_{t=1}^{T} \frac{1}{n_t} \sum_{i=1}^{n_t} \left(y_i^t - \sum_{p=1}^{m} \theta_p C_{pt} x_{ip}^t \right)^2 + \lambda_1 ||\boldsymbol{\theta}||_1 + \lambda_2 \sum_{t=1}^{T} \sum_{p=1}^{m} |C_{pt}|, \tag{4.2}$$

with the constraint that all $\theta_p \geq 0$. Here the mutltitask aspect is explicitely enforced via the $\boldsymbol{\theta}$ parameter, rather than implicitely enforced by a regularization term.

This model gives sparser representations than the multitask lasso, and has the advantage not to impose to select the exact same features across all tasks. The optimization of the parameters is a non-convex problem that can be decomposed in two alternate convex optimizations. This optimization, however, is much slower than that of the multitask lasso.

Multiplicative multitask lasso with task descriptors We now suppose that a matrix $\boldsymbol{Z} \in \mathbb{R}^{T\times d}$ describes each of the T tasks with a d-dimensional vector. When each task corresponds to a molecule, these representations can be given by the fingerprints described in Section 4.1. Inspired by kernel approaches, where task similarities are encoded in the model [26, 68], we introduced the *multiplicative multitask lasso with task descriptors* (MMLD), where the task descriptors are used to explain the specific effect modulating each feature for each task.

More specifically, we follow the multi-level multitask lasso idea and replace in Eq. (4.2) the term C_{pk} with a linear combination of the task descriptors. Hence we formulate the following optimization problem:

$$\underset{\boldsymbol{\theta}\in\mathbb{R}^m,\boldsymbol{A}\in\mathbb{R}^{m\times d}}{\arg\min} \sum_{t=1}^{T}\frac{1}{n_t}\sum_{i=1}^{n_t}\left(y_i^t-\sum_{p=1}^{m}\theta_p\left(\sum_{j=1}^{d}A_{pj}Z_{tj}\right)x_{ip}^t\right)^2+\lambda_1||\boldsymbol{\theta}||_1+\lambda_2\sum_{p=1}^{m}\sum_{j=1}^{d}|A_{pj}|, \quad (4.3)$$

again with the constraint that all $\theta_p \geq 0$. Now A_{pj} indicates the importance of descriptor j for feature p, and controls the specificity of each task. $\lambda_1 > 0$ and $\lambda_2 > 0$ are the regularization parameters for each component of $\boldsymbol{B}$.

An important feature of our proposition is that, because predictions for a new data point $\boldsymbol{x}$ are made as $\sum_{p=1}^{m}\theta_p\left(\sum_{j=1}^{d}A_{pj}Z_{tj}\right)x_p$, we can make predictions for tasks for which no training data is available: the only task-dependent parameters are the descriptors Z_{tj}. This ability to extrapolate to new tasks is not shared by the existing multitask Lasso methods.

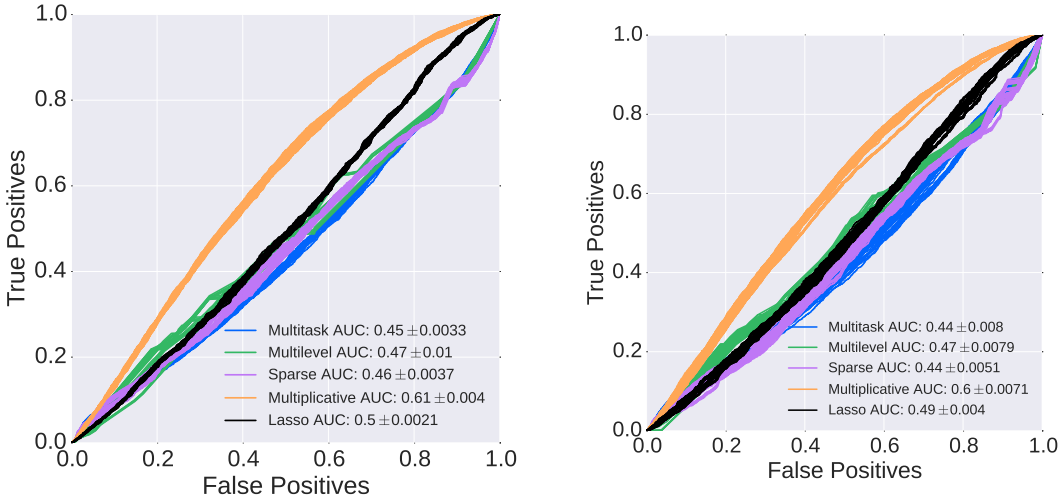
The solution to Eq. (4.3) can be obtained via alternate convex optimization steps. We provide a Python implementation at
`https://github.com/vmolina/MultitaskDescriptor`.

4.2.2 *Experimental results*

Our experiments on simulated data sets show that the MMLD is much more stable than the single-task lasso or the multi-level multiplicative lasso, and presents a much lower variation in the number of selected features than all other methods. The MMLD's ability to recover causal features is also superior to that of its comparison partners. Using task descriptors hence seems to increase the robustness of the feature selection procedure. Finally, the cross-validated root mean squared error of our method is significantly lower than that of state-of-the-art approaches, particulary when the number of available samples is low.

In addition, our experiments on the prediction of whether a peptide can bind to a given MHC-I (major histocompatibility complex class I) protein, based on pairs of binding or non-binding peptide sequences and MHC-I alleles, show that the MMLD performs comparably to the multitask lasso. Here each tasks corresponds to one MHC-I allele.

More interestingly, these experiments illustrate the ability of the MMLD to make predictions on tasks for which no training data is available. Figure 4.1 shows the performance of models previously trained on each of the two data sets provided by Heckerman, Kadie, and Listgarten [100] on the data set provided by Peters et al. [194]. When predicting for a new task with methods others than the MMLD, we use the mean of the predictions made by all models trained on the other tasks. Although its performance is poor, MMLD is the only approach that outperforms the trivial baseline (ROC-AUC=0.5).



(a) Models trained on the first data set in Heckerman, Kadie, and Listgarten [100], evaluated on the dataset in Peters et al. [194]

(b) Models trained on the second data set in Heckerman, Kadie, and Listgarten [100], evaluated on the dataset in Peters et al. [194]

Figure 4.1: ROC curves for the prediction of MHC-I binding, cross-dataset.

4.3 Conclusion and perspectives

In this chapter, I have shown how using task descriptors to guide how much information two tasks should share can noticeably improve the performance of multitask approaches. In particular, such approaches make it possible to make predictions for tasks for which no training data is available. This setting is often encountered in biological applications.

Unfortunately, I have not yet been able to apply these ideas to genome-wide association studies, for lack of appropriate data sets. The work in Section 4.2 was initially motivated by the Rheumatoid Arthritis DREAM Challenge [223], in which we attempted to predict response to several rheumatoid arthritist treatments using both chemical descriptors of the treatment, and SNP data. Unfortunately, one of the conclusions of this challenge was that there was not enough information within the SNPs – indeed gene expression data may have been more appropriate for a first study – to build computational models from this data [223].

5

GENERAL-PURPOSE COMPUTING ON GRAPHICS PROCESSING UNITS

The methods I have presented so far consider either independent or additive effects of molecular features on a phenotype. It is widely accepted, however, that one step towards unveiling the missing heritability is to consider *interactive*, or *synergistic* effects, a phenomenon called *epistasis*, across the whole genome [158, 290].

However, the community is still building the body evidence to support this component of the genetic architecture of complex human traits. Sophisticated modeling approaches and robust computational techniques are an essential part of these ongoing efforts [206].

Unfortunately, the detection of two-loci interactions in GWAS requires a massive amount of computation, as the number of pairs of SNPs that need to be examined can be in the order of $10^{10} - 10^{14}$. In this chapter, I will present how one can use *general-purpose graphics processing unit computing* (*GPGPU computing*) to address computational limitations. While GPGPU computing is now faciliated by numerous linear algebra and machine learning libraries, such was not the case in 2011, when we proposed GLIDE [117].

GPGPU approaches do not solve the statistical problems arising from data scarcity and multiple hypothesis testing, which are similarly exacerbated by the increase in the number of statistical tests to perform. I will present in Chapter 6 a *targeted epistasis* approach that searches for interactions between a specific SNP and the rest of the genome. In addition to limiting the number of tests to perform, our method gains power from avoiding the evaluation of main effects, and accounts for linkage disequilibrium (see Section 1.1.2).

The methods presented in these two chapters model epistasis as a multiplicative effect between two SNPs on the phenotype. In addition, kernels allow us to model a greater variety of nonlinear dependencies, whether between a single genomic feature and the phenotype, or between an arbitrary number of features within a predefined set. This will be the topic of Chapter 7.

In this chapter, after having presented epistasis in Section 5.1 and earlier approaches to use GPUs for epistasis detection(Section 5.2), I will describe in Section 5.3 how we implemented a simple yet efficient linear regression model for the detection of two-locus epistasis in CUDA. In Section 5.4, I will briefly explain how we were able to analyse a hippocampal volume GWAS data set of 567 subjects and over a million SNPs in six hours, where equivalent CPU-based approaches would haave required more than a year's time to complete the same task.

This work was publisehd as

> Tony Kam-Thong, Chloé-Agathe Azencott, Lawrence Cayton, Benno Pütz, André Altmann, Nazanin Karbalai, Philipp G. Sämann, Bernhard Schölkopf, Betram Müller-Myhsok, and Karsten M. Borgwardt. GLIDE: GPU-based linear regression for the detection of epistasis. *Human Heredity*, 73:220–236, 2012.

5.1 Epistasis

Constructing additive models of significant SNPs only explains a small fraction of the heritability of phenotypes [290]. For human height, an extensively-studied trait, this proportion is only 5% [90]; using a linear model from the start raises this proportion to 45% [278], still far from an heritability that has been estimated at 80% [224]. By revealing genetic interactions, epistasis can give an insight into the complex mapping between genotype and phenotype that cannot be extracted from marginal association testing.

Examples of phenotypes for which synergistic effects between gene loci have indeed proven a reliable predictor variable of the phenotypic outcome include diseases such as type 1 and type 2 diabetes [52, 55], inflammatory bowel disease [41] and hypertension [266]. Several examples detailing the different nature of genetic interactions enhancing or suppressing cancer mutations are listed in Ashworth, Lord, and Reis-Filho [6], and new therapeutic treatments have been proposed to target these interactions. In addition, epistatic effects have also been observed in intermediate phenotypes gained by neuroimaging such as working memory-related brain activation [241], and several epistatic mechanisms have been highlighted in the onset of Alzheimer disease [49]. Most notably, the interaction between the two genes BACE1 and APOE4 was found to be significant on four distinct datasets.

The definition of statistical epistasis dates back to Fisher [73], who characterizes it as the departure from additivity in a mathematical model relating multilocus genotypes to phenotypic variation. A number of strategies deployed in the context of statistical epistasis are reviewed in Cordell [51] and Niel et al. [181]. Among those, this chapter will focus on exhaustive search strategies, which systematically evaluate all possible pairwise interactions between two SNPs.

5.2 Epistasis detection using GPGPU

Several software tools designed to perform epistasis searches on GPUs, such as SHEsisEpi [105], EPIBLASTER [116], EPIGPUHSIC [115], GBOOST [280], epiGPU [102], or GWIS [82] have been proposed and demonstrated substantial advantages of the use of GPU in this application. However, they are either restricted to binary or discrete phenotypes, which limits the scope of data sets they can analyze, or neglect main effects, which hinders the overall interpretation of their results.

The more popular of these tools, meant for case-control studies, is GBOOST [280], a GPGPU variant of BOOST [257]. At its core, BOOST computes a likelihood ratio test between two logistic regression models: a main-effect logistic regression over the two SNPs of the pairs under scrutinity, and a full logistic regression over these two SNPs and their product. In addition, BOOST uses a Boolean representation of the genotypes, allowing for quick Boolean operations. A screening step based on the Kirkwood superposition approximation of the main-effect model further speed ups computations.

By contrast, the method we proposed in Kam-Thong et al. [117] aims to be general enough to be applicable to pairwise epistasis studies of various real or continuous value predictor inputs (genetic and environmental factors) related to the phenotypic output.

5.3 GLIDE: GPU-based linear regression for the detection of epistasis

GLIDE (GPU-based LInear regression for the Detection of Epistasis) is a high-performance GPU-based implementation of a systematic two-locus epistasis search. In essence, GLIDE models epistasis as the presence of a non-zero interaction term in a linear model explaining the phenotype as a linear combnation of two SNPs and their products, and computes these terms and their statistical significance very efficiently for all possible pairs of SNPs in a data set.

5.3.1 *Linear regression model*

Let us consider a data set $(\boldsymbol{X}, \boldsymbol{y}) \in \mathbb{R}^{n \times m} \times \mathbb{R}^n$ of m SNPs measured over n individuals together with their real-valued phenotypes. For each SNP pair (p, q) with $1 \leq p, q \leq m$, we consider that the n samples are n independent and identically distributed observations drawn from an m-dimensional variable X and a response variable Y. We denote by X_p the p-th dimension of X.

We fit the following linear regression model:

$$Y \sim \alpha_0^{pq} + \alpha_1^{pq} X_p + \alpha_2^{pq} X_q + \alpha_{12}^{pq} X_p X_q, \tag{5.1}$$

using a standard least squares model (see Section 1.2.4 and Eq. (1.2)):

$$\widehat{\boldsymbol{\alpha}^{pq}} = (\boldsymbol{X}_{pq}^\top \boldsymbol{X}_{pq})^{-1} \boldsymbol{X}_{pq}^\top \boldsymbol{y}, \tag{5.2}$$

where $\boldsymbol{X}_{pq} \in \mathbb{R}^{n \times 4}$ is built by concatenating a column of all ones, the p-th column of $\boldsymbol{X}$, its q-th column, and their product. $\widehat{\boldsymbol{\alpha}^{pq}}$ is a 4-dimensional vector.

Under this model, we will say that there is an epistatic effect of SNPs p and q on the phenotype if the estimated regression coefficient $\widehat{\alpha_{12}^{pq}}$ is significantly different from 0. We determine this using a t-test with $(n - 4)$ degrees of freedom. The t-score is given by:

$$t_{pq} = \frac{\widehat{\alpha_{12}^{pq}} \sqrt{n-4}}{\sqrt{r_{pq} \cdot \left[(\boldsymbol{X}_{pq}^\top \boldsymbol{X}_{pq})^{-1}\right]_{4,4}}}, \tag{5.3}$$

where the residual sum of squared errors r_{pq} is given by

$$r_{pq} = \sum_{i=1}^{n} \left(y_i - \left(\widehat{\alpha_0^{pq}} + \widehat{\alpha_1^{pq}} x_{ip} + \widehat{\alpha_2^{pq}} x_{iq} + \widehat{\alpha_{12}^{pq}} x_{ip} x_{iq} \right) \right)^2. \tag{5.4}$$

In the following section, I will show how t_{pq} can be computed efficiently for millions of values of p and q on GPUs.

5.3.2 *GPU implementation*

GPUs are composed of several hundred lightweight processing units, and are only effective for tasks that decompose into many subproblems that can be solved in parallel. As our problem is composed of many independent regression tasks, it fits easily onto the GPU architecture.

In the GPU programming model, each processing unit executes a thread. These threads are grouped in blocks of size B; within each block, threads can cooperate through execution

synchronization and efficient low-latency memory sharing. Leveraging this block structure to reduce memory accesses is crucial for performance. GLIDE associates each thread with a single regression problem. These threads are then collected into blocks such that threads within a block can share access to a subset of matrix $\boldsymbol{X}$. In particular, each two-dimensional block of size $B \times B$ loads $2 \times B$ columns of $\boldsymbol{X}$ and solves all pairwise linear regression problems on the corresponding $B \times B$ pairs of SNPs, by performing the following operations:

1. Compute matrix $\boldsymbol{X}_{pq}$;
2. Estimate the regression coefficients $\widehat{\boldsymbol{\alpha}^{pq}}$ by solving Eq. (5.2) analytically – which is possible as the inversion it requires is that of a 4×4 matrix;
3. Estimate the corresponding residual sum of squared errors r_{pq} (Eq. (5.4));
4. Estimat the t-score t_{pq} (Eq. (5.3)).

GLIDE is written in the C programming language using NVIDIA's CUDA extension and made available online at `https://github.com/BorgwardtLab/Epistasis-GLIDE`.

5.4 Experimental results

We conducted this study using 12 NVIDIA GTX 580 GPUs. These cards have 16 streaming multiprocessors, each holding 32 processors, yielding a total of 512 GPU cores. They support double-precision floating-point calculations. The host machine was running on an Intel Core i7 920 with a 2.66-GHz CPU host using 12 GB of DDR3 RAM.

5.4.1 *Runtime*

Using a synthetic data set of 1 000 individuals and about 5 000 SNPs (corresponding to 25 million SNP pairs to test), we compared the speed of GLIDE with that of the state-of-the-art CPU methods PLINK [200] and FastEpistasis [219]. PLINK performs a likelihood ratio test comparing the regression model in Eq. (5.1) with a similar model without the $\alpha_{12}^{pq} X_p X_q$ interaction term. FastEpistasis performs exactly the same computations, but distributes the work over a multi-CPU environment. Its speed scales up linearly with the number of CPU cores used.

The correlation coefficient between the p-values produced by GLIDE and those returned by PLINK is exactly 1, therefore satisfyingly validating the correctness of our implementation.

The runtime of all three methods scales up linearly with respect to the number of pairwise SNP interactions. Figure 5.1 illustrates the advantage of porting the code onto GPUs. Although the performance depends on technical specifications, GLIDE runs at least 2 000 times faster than PLINK; and to reach reach GLIDE's speed on a single GPU with FastEpistasis would require a cluster of 250 nodes.

5.4.2 *Hippocampal volume association study*

We conducted an exhaustive search for epistatic interactions associated with hippocampal volume. The hippocampus is a small but complex bilateral brain structure involved in many cognitive processes, particularly the formation of new memories. An extreme reduction of its volume is a hallmark of Alzheimer's disease, but mild forms of hippocampal volume reductions are also found in patients with schizophrenia or recurrent depression [77]. The

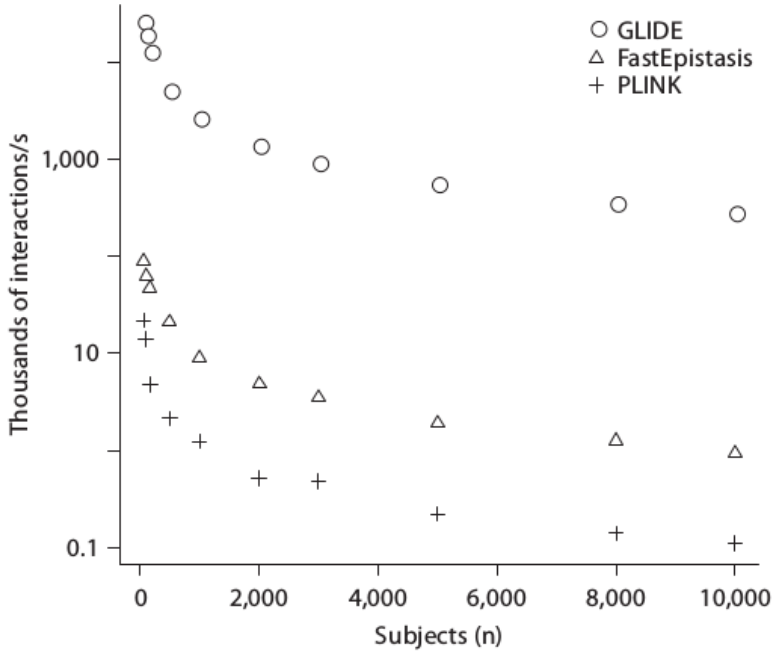


Figure 5.1: Runtime (in thousands of interactions per second) of GLIDE, FastEpistasis (on a single CPU core), and PLINK, as a function of the number of subjects, displayed on a logarithmic scale.

hippocampal volume is heritable to some degree, with the heritability estimated from twin studies to be between 40 and 69% [192], and is therefore a good candidate for explicit genetic studies.

We used data for 567 subjects from the Max Planck Institute for Psychiatry, for which hippocampal volume was determined from high-resolution MRI images as in Kohli et al. [124]. After quality control, a total of 1 075 163 SNPs were available.

We first conducted a standard single-SNP association study on this data set. No SNP was significant at $p < 0.05$ after Bonferroni correction. We then performed an exhaustive pairwise test using GLIDE. The strongest p-value was 2.6×10^{-13}, larger than a Bonferroni-corrected threshold of 0.05 for $\frac{1}{2}1,075,163^2$ tests (8×10^{-14}). However, genes linked to hippocampus-dependent tasks in animal models (such as ICOS and CTLA-4), as well as hippocampal development (ZEB2), cerebral cavernous malformations (ZLPD1) or a cation channel expressed in the brain (TRPM6) were tagged by SNPs involved in the top 20 pairs according to our ranking.

None of the highest-ranked univariate SNPs are involved in the top 20 highest-ranking interaction pairs. In other words, the best-ranked pairs would not have been detected if we had first pruned the SNP space based on the univariate tests.

The 20 highest-ranking SNPs in the single-locus study explain 18% of the phenotypic variance, while the 20 highest-ranking pairs explain 40% of this variance, suggesting that GLIDE detects informative features in the data.

5.5 Conclusion and perspectives

One step towards revealing the missing heritability in complex traits is to search for epistatic effects by mapping phenotypic variation to pairs of genetic loci. We implemented a fast two-locus genome-wide interaction detection algorithm, which performs an exhaustive SNP-SNP interaction search on typically sized GWAS data sets in six hours on relatively inexpensive GPUs. Although recent developments in both GPGPU librairies and computing hardware make our work somewhat outdated, GLIDE is one of the fastest tools for the exhaustive testing of pairs of SNPs in a linear model [38].

That GLIDE is based on a linear model makes it possible to analyze real-valued genotypes, as obtained from imputation methods, as well as to incorporate confounding factors as additional covariates, or environmental factors for gene-environment studies.

Several authors [101, 151] have argued that linear models can also be used for case-control studies, where one would typically chose a logistic regression model. We have run some experiments on the Wellcome Trust Case Control Consortium data sets [244] and our results match those previously reported by Wan et al. [257]. However, for case-control phenotypes and discrete genotype encodings, GBOOST [280] is approximately 75 times faster than GLIDE.

While GLIDE addresses the *computational* issues associated with the detection of two-locus epistasis, it does not address the *statistical* limitations of running more than 10^{10} statistical tests on thousands of samples only. I will present in the following chapter an approach that alleviates this burden by first focusing on synergistic effects with a predetermined locus, and second uses a modified oucome approach to avoid having to evaluate main effects in the search for epistasis.

6

TARGETED CASE-CONTROL EPISTASIS WITH MODIFIED OUTCOME REGRESSION

I have shown in the previous chapter that it is possible to test SNP-SNP interactions exhaustively at a genome-wide scale. Several packages in addition to GLIDE are available for this purpose, corresponding to several definitions of epistasis and applicable to diverse settings [35, 181]. However, exhaustive testing must be followed by multiple hypotheses testing procedures, reducing the statistical power of the studies [176].

In this chapter, I will introduce *targeted* epistasis, which aims at identifying epistasis between a specific locus and the rest of the genome (Section 6.1). I will then present epiGWAS, a method for case-control targeted epistasis (Section 6.2). EpiGWAS models linkage disequilibrium (see Section 1.1.2), and gains power from avoiding the need to evaluate main effects thanks to a modified outcome reformulation.

The work presented in this chapter is available as a preprint:

> Lotfi Slim, Clément Chatelain, Chloé-Agathe Azencott, and Jean-Philippe Vert. Novel methods for epistasis detection in genome-wide association studies. *bioRxiv:10.1101/442749*, 2018.

6.1 Targeted epistasis

Exhaustive genome-scale models with all pairwise terms are computationally intensive and suffer from low statistical power. Instead of constructing exhaustive models, we therefore focus on expanding knowledge around predetermined loci, which we refer to as *targets*. Such targets can be drawn from the literature, or be top hits in previous GWAS. This leads to a drastic knowledge-driven reduction of the number of interactions to study.

Limiting the scope of the interactions of interest allows us more flexibility in their modeling: rather than limiting ourselves to pairwise effects studied independently from each other, as with GLIDE or other exhaustive epistasis search tools, we propose a model that captures interactions between the target and all other SNPs in the genome at once. In addition, we wish our model to account for the main effects of all the SNPs involved.

Such models are not frequent in the epistasis literature. In the clinical trial literature, however, similar problems appear, where the goal is to infer the variation in treatment response that is due to the interaction between the treatment assignment and the clinical covariates. In particular, propensity score [210] techniques have been developed specifically for this purpose. We therefore draw from this literature to propose a model selection method that robustly infer second-order interactions with a fixed SNP, through the formulation of different ℓ_1-penalized regression problems.

6.2 Modified outcome regression

6.2.1 *Mathematical model of epistasis*

We model the data with three random variables: X, which is m-dimensional and represents the genotype, with values in $\{0,1,2\}^m$; $A \in \{-1,+1\}$, which is the target SNP; and $Y \in \{0,1\}$, which represents the phenotype. We restrict ourselves to a binary encoding of the target SNP.

A being binary, it is always possible to write Y as

$$Y = \mu(X) + \delta(X) \cdot A + \epsilon, \tag{6.1}$$

where ϵ is a zero mean random variable and

$$\begin{cases} \mu(X) = \dfrac{1}{2}\left[\mathbb{E}(Y|A=+1,X) + \mathbb{E}(Y|A=-1,X)\right], \\ \delta(X) = \dfrac{1}{2}\left[\mathbb{E}(Y|A=+1,X) - \mathbb{E}(Y|A=-1,X)\right]. \end{cases} \tag{6.2}$$

The term $\mu(X)$ represents the main effects of the genotype on the phenotype. The term $\delta(X) \cdot A$ represents the synergistic effects between A and all SNPs in X. In the context of genomic data, we can interpret these synergies as pure epistatic effects. Furthermore, if $\delta(X)$ is sparse in the sense that it only depends on a subset of elements of X (which we call the *support* of δ), then the SNPs in the support of δ are the ones interacting with A. In other words, searching for epistasis between A and SNPs in X amounts to searching for the support of δ.

6.2.2 *Modified outcome regression*

Because, for any given sample, only one of the two potential outcomes $A=+1$ or $A=-1$ is observed, directly estimating $\delta(X)$ Eq. (6.2) is difficult. One could estimate the first term of $\delta(X)$ on the samples for which $A=+1$ and the second term on those where $A=-1$, but then the causal effect of treatment (here, of SNP A) is estimated across two different groups of samples which my differ in other ways than treatment assignment. Following Tian et al. [246], we consider $\widetilde{A} = (A+1)/2 \in \{0,1\}$, and rewrite Eq. (6.2) as:

$$\delta(X) = \frac{1}{2}\,\mathbb{E}\left[Y\left(\frac{\widetilde{A}}{\pi(\widetilde{A}=1|X)} - \frac{1-\widetilde{A}}{\pi(\widetilde{A}=0|X)}\right)\middle| X\right],$$

where $\pi(\widetilde{A}|X)$, the conditional probability of A given X, is called the *propensity score*. The propensity score models the linkage disequilibrium (see Section 1.1.2) between the target SNP and the rest of the genotype.

Given an estimate of $\pi(\widetilde{A}|X)$, we define the *modified outcome* $\widetilde{Y}$ of a triplet (X,A,Y) as:

$$\widetilde{Y} = Y\left(\frac{\widetilde{A}}{\pi(\widetilde{A}=1|X)} - \frac{1-\widetilde{A}}{\pi(\widetilde{A}=0|X)}\right), \tag{6.3}$$

and re-express simply

$$\delta(X) = \frac{1}{2}\,\mathbb{E}\left[\widetilde{Y}|X\right]. \tag{6.4}$$

Our definition of modified outcome (Eq. (6.3)) generalizes that of Tian et al. [246], which considers the specific case where A and X are independent.

According to our definition, the SNPs in epistasis with A are those contributing to $\delta(X)$. We now make the assumption that $\widetilde{Y}$ depends linearly on X, and recover the SNPs in epistasis as the support of a linear regression between $\widetilde{Y}$ and X. Importantly, we do not need to estimate μ at all to estimate the support of δ.

6.2.3 *Support estimation*

We estimate the support of δ by combining an elastic-net model (see Section 1.2.5) with the stability selection approach proposed by Meinshausen and Bühlmann [164]. The general idea of stability selection is to repeat the feature selection procedure several times and consider as selected only the features that appear in many of the obtained solutions. We use the modification proposed by Haury et al. [98], and select features according to the area under the stability path, so as to capture variables that enter the regularization path early.

6.2.4 *Propensity scores estimation*

For the estimation of the propensity scores $\pi(\widetilde{A}|X)$, that is to say, the dependence between SNPs, we use a Hidden Markov Model as proposed by Scheet and Stephens [214] for imputation. This model is a popular one for the representation of genetic architectures [120, 204, 235]. The hidden states represent contiguous clusters of phased haplotypes. The emission states correspond to SNPs.

6.2.5 *Correction for numerical instability and large-sample variance*

In practice, given n samples $(\boldsymbol{X}, \boldsymbol{a}, \boldsymbol{y}) \in \{0,1,2\}^{n\times m} \times \{-1,1\}^n \times \{0,1\}^n$, computing the n modified outcomes $\widetilde{\boldsymbol{y}}$ according to Eq. (6.3) requires using the inverses of the propensity scores, which tend to be close to 0. To avoid the resulting numerical instability and variance, we follow Lunceford and Davidian [155] to propose a *robust modified outcome* approach.

The resulting methods for the detection of targeted epistasis in case-control studies, with LD modeling and no need to evaluate main effects, is one of several we proposed in an R package called epiGWAS, and available on CRAN package [227].

6.3 Experimental results

Our experiments on simulated data show that the robust modified outcome approach in epiGWAS has power and false discovery rate comparable to those of state-of-the-art approach GBOOST (see Section 5.2), with a slight advantage in low sample sizes.

To illustrate the scalability of our methods to real datasets in the case of targeted epistasis, we applied epiGWAS to the Type II Diabetes data set of the Wellcome Trust data set [244]. To the best of our knowledge, there are no confirmed epistatic interactions for this disease. We studied the synergies with a particuler target, SNP rs41475248 on chromosome 8, which we chose because it is involved in 3 epistatic interactions according to GBOOST. We found that the correlation between the findings of GBOOST and those of epiGWAS are rather low. This can be explained by the fact that both tools use different mathematical

models of epistasis, and suggests that the two approaches could be use in a complementary fashion.

6.4 Conclusion and perspectives

In this chapter I have presented epiGWAS, a method for the detection of targeted epistasis in case-control studies, which uses a modified outcome approach to avoid having to evaluate main effects, further improving statistical power. Our first experimental results show good power with respect to comparable approaches, and we are currently investigating in more details the application of epiGWAS to multiple sclerosis.

Our approach was inspired by the clinical trials literature. The rich literature in this field opens the door to a much broader panel of methods. Future directions could include conditioning for multiple covariates to account for, among other things, higher-order interactions and population stratification.

epiGWAS accounts for linkage disequilibrium through the hidden Markov modeling of propensity scores. The use of a robust modified outcome estimator compensates possible propensity score misspecifications. A possible area of improvement would be to instead follow Athey, Imbens, and Wager [8], who completely forgoe propensity scores for the estimate of average treatment effects.

The synergies identified by our approach are often complementaries to that of GBOOST, one of the most popular tools for identifying epistasis in case-control studies; this is not unexpected, as the underlying statistical models of epistasis are different. This suggest that a consensus method combining GBOOST, epiGWAS, and even additional tools, could improve the recovery of actual epistatic effects.

Given the computational resources, epiGWAS could conceptually be applied at a genome-wide scale, considering each SNP in turn to be a target. However, this would be at the expense of the statistical power gained from reducing the number of tests. Runtime could be improved using recent fast lasso solvers for large scale problems [136, 161].

epiGWAS is limited to modeling quadratic effects between one target SNP and a linear combination of other SNPs. In the following chapter, I will show how to use kernels to model a greater variety of nonlinear dependencies.

7

NONLINEAR FEATURE SELECTION WITH KERNELS

In the previous chapters, I have outlined strategies to identify quadratic interactions between genomic features and a phenotype, either pairwise (as in Chapter 5) or as a linear combination of products between a target SNP and the rest of the genome (Chapter 6). However, one can envision more complex types of interactions; and *kernels* appear as a natural candidate to model those (see Section 1.2.6).

In addition to modeling synergetic effects that several features can have on a phenotype, kernels can also be used to measure a nonlinear association between a single feature and a phenotype. In particular, the Hilbert-Schmidt Independence Criterion (see Section 1.2) uses kernels to measure the dependency between two variables in a nonlinear way. In Section 7.1, I will show how this idea can be applied to perform non-redundant feature selection in very high dimensional settings.

Kernels are already well used in the GWAS community, in particular through the Sequence Kernel Association Test (SKAT) (see Section 1.2.6), which is used to assess the association between a predefined group of SNPs and the phenotype. In Section 7.2, I will describe a generalization of the SKAT statistics to a broader family of association scores that enjoy greater statistical power in practice.

The work presented in this chapter was published as

- Héctor Climente-González, Chloé-Agathe Azencott, Samuel Kaski, and Makoto Yamada. Block HSIC Lasso: model-free biomarker detection for ultra-high dimensional data . *Bioinformatics*, 35(14), 2019. Proceedings of the 27st Annual International Conference on Intelligent Systems for Molecular Biology (ISMB 2019).
- Lotfi Slim, Clément Chatelain, Chloé-Agathe Azencott, and Jean-Philippe Vert. kernelPSI: a post-selection inference framework for nonlinear variable selection. In *Proceedings of the Thirty-Sixth International Conference on Machine Learning (ICML)*, volume 97, pages 5857–5865, 2019.

7.1 Non-redundant biomarker discovery with block HSIC lasso

Nonlinear association measures, such as mutual information [53] or the Hilbert-Schmidt Independence Criterion (HSIC) [84], select the features with the strongest dependence with the phenotype. However, these methods do not account for the redundancy between features, which is frequent in biological data sets. Hence, many redundant features are typically selected, hindering interpretability.

While in GWAS this redundancy is often exploited, in particular when rare variants are being measured, so as to increase statistical power, we here suppose that one wishes to find non-redundant biomarkers, each pointing at a different region of the genome rather than accumulating evidence for a particular region.

7.1.1 *mRMR*

The nonlinear selection of non-redundant features can be achieved with the minimum redundancy maximum relevance (mRMR) algorithm [58, 191]. mRMR selects a set of non-redundant features that have high association with the phenotype, while penalizing the selection of mutually dependent features by a mutual information term. Supposing our data, containing n samples and m features, is represented by $\boldsymbol{X} \in \mathbb{R}^{n \times m}$, and the phenotype by $\boldsymbol{y} \times \mathbb{R}^n$, we call $\boldsymbol{g}_p \in \mathbb{R}^n$ the vector containing n observations for feature p and corresponding to a column of $\boldsymbol{X}$. The mRMR score of a set of features $\mathcal{S}$ is defined as

$$\text{mRMR}(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{p \in \mathcal{S}} \widehat{\text{MI}}(\boldsymbol{g}_p, \boldsymbol{y}) - \frac{1}{|\mathcal{S}|^2} \sum_{p \neq q \in \mathcal{S}} \widehat{\text{MI}}(\boldsymbol{g}_p, \boldsymbol{g}_q), \tag{7.1}$$

where $\widehat{\text{MI}}$ is an empirical estimate of mutual information [191].

Finding the set of features $\mathcal{S}$ that maximizes the mRMR score is a nonconvex optimization problem. mRMR implementations therefore rely on a greedy approach, with no guarantee on finding the optimal set. In addition, the estimation of mutual information is difficult [256], which further limits mRMR.

7.1.2 *HSIC lasso*

Yamada et al. [273] proposed to apply a similar idea, using HSIC instead of mutual information to measure dependency between variables. Their HSIC lasso uses an ℓ_1 penalty term to select a small number of features. This results in a convex optimization problem, for which one can therefore find a globally optimal solution.

If we denote by $\widehat{\text{HSIC}}$ the estimator of HSIC proposed by Gretton et al. [86], HSIC lasso solves the following optimization problem:

$$\max_{\boldsymbol{\alpha} \geq 0} \sum_{j=1}^{m} \alpha_p \, \widehat{\text{HSIC}}(\boldsymbol{g}_p, \boldsymbol{y}) - \frac{1}{2} \sum_{p,q=1}^{m} \alpha_p \, \alpha_q \, \widehat{\text{HSIC}}(\boldsymbol{g}_p, \boldsymbol{g}_q) - \lambda \, ||\boldsymbol{\alpha}||_1. \tag{7.2}$$

The first term enforces selected features that are highly dependent on the phenotype; the second term penalizes selecting mutually dependent features; and the third term enforces selecting a small number of features. As with the lasso, the selected features are those that have a non-zero coefficient α_p and $\lambda > 0$ is a regularization parameter that controls the sparsity of the solution.

HSIC lasso can be rewritten as a regular lasso problem using a vectorized version of the kernel matrices involved in the computation of $\widehat{\text{HSIC}}$, and performs well for high-dimensional data. However, it requires a large amount of memory, in the order of $O(mn^2)$. Several approximations have been proposed. One can for example use a memory lookup to reduce memory space, which is in exchange computationally expensive [273]. Another possibility is to rewrite the problem using the Nyström approximation [217], which makes it non-convex, but amenable to a MapReduce implementation [275].

7.1.3 *Block HSIC lasso*

In Climente-González et al. [46], we proposed another approximation, which relies on the block HSIC estimator [283] to estimate the HSIC terms in Eq. (7.2). By splitting the data in

blocks of size $B \ll n$, the memory complexity of HSIC lasso goes from $O(mn^2)$ down to $O(mnB)$, and the optimization problem of the block HSIC lasso remains convex.

The block HSIC estimator [283] is computed by first partitioning the dataset into $\frac{n}{B}$ partitions $\{\{(\boldsymbol{x}_i^{(k)}, y_i^{(k)})\}_{i=1}^{B}\}_{k=1}^{n/B}$, where B is the number of samples in each block. The block size B is set to a relatively small number such as 10 or 20 ($B \ll n$). Then, the block HSIC estimator can be written as

$$\widehat{\mathrm{HSIC}}_b(\boldsymbol{g}_p, \boldsymbol{y}) = \frac{B}{n} \sum_{k=1}^{n/B} \widehat{\mathrm{HSIC}}(\boldsymbol{g}_p^{(k)}, \boldsymbol{y}^{(k)}), \tag{7.3}$$

where $\boldsymbol{g}_p^{(k)} \in \mathbb{R}^B$ represents the p-th feature vector of the k-th partition.

The HSIC estimator in Eq. (7.2) can easily be replaced by the block HSIC estimator, leading to a method we call *block HSIC lasso*. Because the problem can be reformulated as a regular lasso problem, it can be solved using LARS [64], and λ can be chosen to yield a predefined number of selected features. We made both HSIC lasso and block HSIC lasso available in the Python package `pyHSICLasso`[1].

7.1.4 *Relationship with SConES and Grace*

The objective of SConES, presented in Section 2.3, is given by Eq. (2.5):

$$\underset{\boldsymbol{f} \in \{0,1\}^m}{\arg\max}\, \boldsymbol{r}^\top \boldsymbol{f} - \eta ||\boldsymbol{f}||_0 - \lambda \boldsymbol{f}^\top \boldsymbol{L} \boldsymbol{f}.$$

If we set $r_p = \widehat{\mathrm{HSIC}}(\boldsymbol{g}_p, \boldsymbol{y})$ and $\boldsymbol{L}_{j,k} = \widehat{\mathrm{HSIC}}(\boldsymbol{g}_p, \boldsymbol{g}_q)$, and relax $\boldsymbol{f}$ to be real-valued and the ℓ_0 norm to its ℓ_1 surrogate, we obtain a problem equivalent to that of Eq. (7.2). Hence, HSIC lasso can be seen as a special case of the relaxation of SConES, with a network defined from the data and describing dependence between features, itself using the same regularization term as Grace (Eq. (2.1)).

7.1.5 *Experimental results*

Our experimental results on synthetic data sets show that block HSIC lasso is computationally efficient and performs comparably to the state of the art. On gene expression microarray and single-cell RNA sequencing data sets, we found that the biomarkers selected by block HSIC lasso achieve state-of-the-art classification performance. In addition, those biomarkers are quite different from those selected by mRMR, suggesting a complementarity of the two approaches.

We also applied HSIC lasso, using normalized Dirac kernels as kernels, to the WTCCC1 datasets [244] for rheumatoid arthritis (RA), type 1 diabetes (T1D) and type 2 diabetes (T2D). Using nonlinear models such as block HSIC lasso to explore the relationship between SNPs and phenotype does not require to make an assumption on the genetic architecture of the trait. In addition, by penalizing the selection of redundant features, block HSIC lasso avoids selecting multiple SNPs in high linkage disequilibrium. We illustrate this on Figure 7.1, where we highlight, on classic GWAS Manhattan plots, for each of the three phenotypes, the 10 SNPs selected with block HSIC lasso.

1 `https://pypi.org/project/pyHSICLasso`

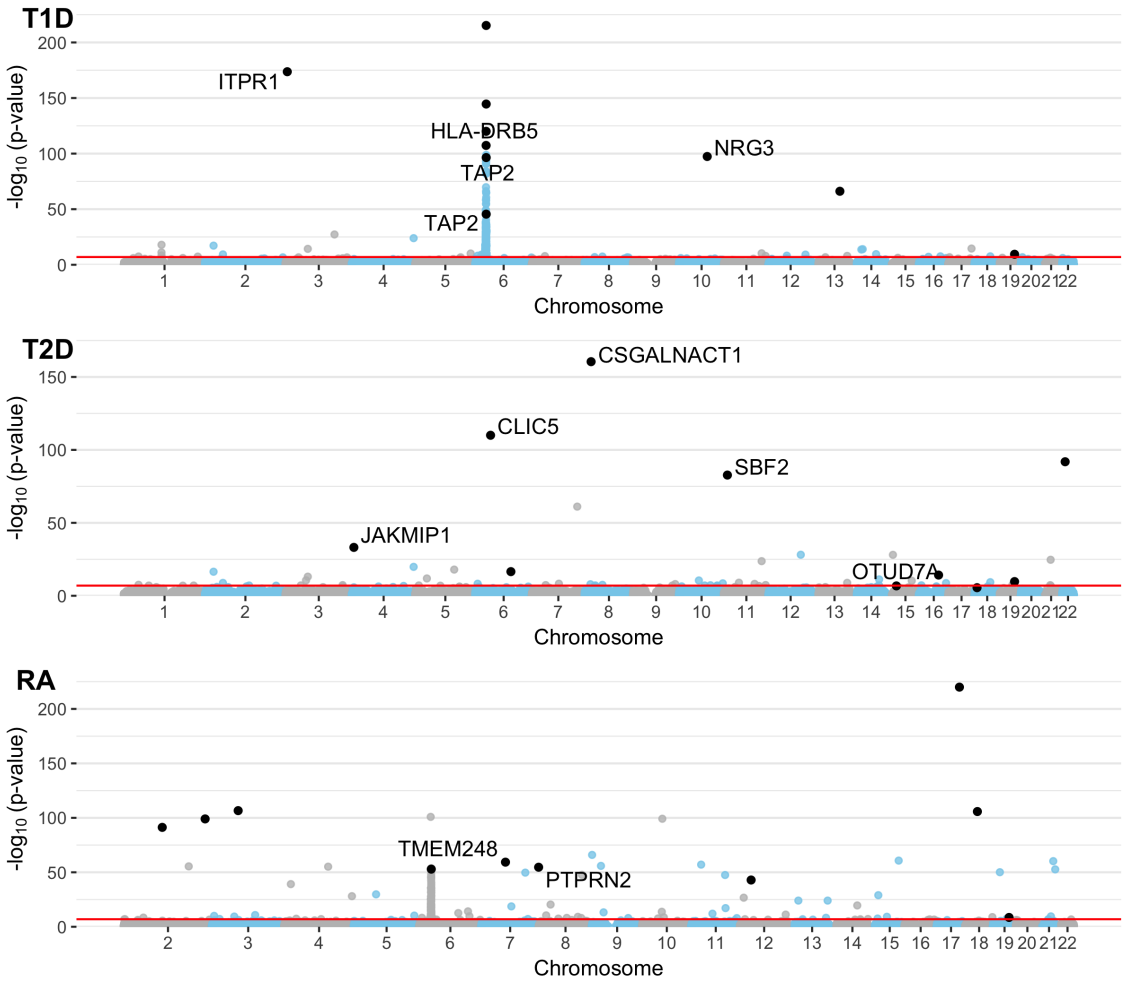


Figure 7.1: Manhattan plot for the three WTCCC1 GWAS, using p-values from the genotypic test. A constant of 10^{-220} was added to all p-values to allow plotting p-values of 0. SNPs in black are the SNPs selected by block HSIC lasso ($B = 20$), 10 per phenotype. When SNPs are located within the boundaries of a gene (± 50 kb), the gene name is indicated. The red line represents the Bonferroni threshold with $\alpha = 0.05$.

Block HSIC lasso selects SNPs among those with the most extreme p-values. In addition, not being constrained by a conservative p-value threshold, block HSIC lasso selects two SNPs in type 2 diabetes with low, albeit non-Bonferroni significant, p-values when they improve classification accuracy. Moreover, the selected SNPs are scattered all across the genome, displaying the lack of redundancy between them. This strategy gives a more diverse set of SNPs than classic GWAS approaches.

All our experimental results are available in Climente-González et al. [46] and can be reproduced using the code we made public at `https://github.com/hclimente/nori`.

7.2 Post-selection inference with kernels

With block HSIC lasso, we have used kernels to model nonlinear effects of a single feature on the phenotype. However, kernels can also be used to create nonlinear models of multiple features, for example SNPs belonging to the same genomic region, acting together on

the phenotype. We propose in Slim et al. [229] to use *quadratic kernel association scores*, a definition under which a number of existing scores such as HSIC estimators fall, to select the kernels most associated with a phenotype. If those kernels are defined on single features, or groups of them, selecting kernels leads to feature selection (individually or by groups). In addition, the *post-selection inference* framework allows us to account for the selection event when performing inference, and in particular to provide p-values for the association of a selected kernel with the phenotype.

7.2.1 *Quadratic kernel association scores*

Let $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ be a kernel matrix over n samples, $\boldsymbol{y} \in \mathbb{R}^n$ a corresponding phenotype or outcome, and $q : \mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n}$ a function. We define a *quadratic kernel association score* (QKAS) as a function

$$\begin{aligned} s : \mathbb{R}^{n \times n} \times \mathbb{R}^n &\to \mathbb{R} \\ (\boldsymbol{K}, \boldsymbol{y}) &\mapsto \boldsymbol{y}^\top q(\boldsymbol{K})\, \boldsymbol{y}. \end{aligned} \tag{7.4}$$

If q is positive semi-definite, there exists a function $h : \mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n}$ such that $h(\boldsymbol{K}) = q^{1/2}(\boldsymbol{K})$; we'll refer to that function as the hat function of s. s is a positive definite form in $\boldsymbol{y}$, and can be rewritten as:

$$s(\boldsymbol{K}, \boldsymbol{y}) = ||\widehat{\boldsymbol{y}_{\boldsymbol{K}}}||_2^2, \tag{7.5}$$

where $\widehat{\boldsymbol{y}_{\boldsymbol{K}}} = h(\boldsymbol{K})\boldsymbol{y}$. Following Reid, Taylor, and Tibshirani [205], who use a similar concept to design statistical tests of linear association between $\boldsymbol{y}$ and a group of features, we call $\widehat{\boldsymbol{y}_{\boldsymbol{K}}}$ a *prototype* for kernel $\boldsymbol{K}$.

This definition encompasses several ways to define the association between a kernel and a phenotype. For example, it generalizes two recently proposed scores for groups of features [152, 205]. In addition, both the biased estimator of HSIC proposed by Gretton et al. [86] and its unbiased estimator proposed by Song et al. [232] can be used to define the QKAS $s(\boldsymbol{K}, \boldsymbol{y}) = \widehat{\mathrm{HSIC}}(\boldsymbol{K}, \boldsymbol{y}\boldsymbol{y}^\top)$. Both are positive quadratic forms in $\boldsymbol{y}$.

7.2.2 *Kernel selection*

Given any QKAS s and a set $\mathcal{S} = \{\boldsymbol{K}_1, \boldsymbol{K}_2, \ldots, \boldsymbol{K}_S\}$ of S kernels, we consider three standard strategies to select a number $S' \leq S$ of kernels from $\mathcal{S}$:

- *Filtering*: select the top S' kernels with highest score $s(\boldsymbol{K}, \boldsymbol{y})$.
- *Forward stepwise selection*: start from an empty list of kernels, and iteratively add new kernels one by one in the list by picking the one that leads to the largest increase in association score when additively combined with the kernels already in the list.
- *Backward stepwise selection*: conversely, start from the full list of kernels, and iteratively remove the one that leads to the smallest decrease in association score.

We can also consider *adaptive* variants of these selection methods, where the number S' of selected kernels is not fixed beforehand but automatically selected in a data-driven way. In adaptive estimation of S', we maximize over S' the association score computed at each step, potentially regularized by a penalty function that does not depend on $\boldsymbol{y}$.

7.2.3 *Post-selection inference*

One way to reduce the statistical burden of multiple hypotheses testing is to start by selecting only relevant features and limiting statistical tests of associations to those. However, if the phenotype is used to perform this selection (as is the case when a regularized linear regression is used, or in the kernel selection schemes previously described), the features that are tested are more likely to exhibit a strong association with the phenotype, as they were specifically selected for that purpose. Therefore, standard statistical tests must be adapted to correct for this bias, a setting refered to as *post-selection inference* [137].

Post-selection inference is possible in the context of kernel selection as we have described it because the selection event can be described as a conjunction of quadratic constraints, which allows us to leverage techniques explored by Loftus and Taylor [152] and Yang et al. [277] for group feature selections.

More specifically, given a set of kernels $\mathcal{S} = \{\boldsymbol{K}_1, \boldsymbol{K}_2, \ldots, \boldsymbol{K}_S\}$, a quadratic kernel association score s and its hat function h, and one of the aforementioned kernel selection strategies, let $m(\boldsymbol{y}) \subseteq \mathcal{S}$ denote the subset of kernels selected based on the phenotype vector $\boldsymbol{y}$. For any $\mathcal{M} \subseteq \mathcal{S}$, there exists $i_{\mathcal{M}} \in \mathbb{N}$, and $(q_{\mathcal{M},1}, b_{\mathcal{M},1}), \ldots, (q_{\mathcal{M},i_{\mathcal{M}}}, b_{\mathcal{M},i_{\mathcal{M}}}) \in \mathbb{R}^{n\times n} \times \mathbb{R}$ such that

$$\{\boldsymbol{y} : m(\boldsymbol{y}) = \mathcal{M}\} = \bigcap_{i=1}^{i_{\mathcal{M}}} \{\boldsymbol{y} : \boldsymbol{y}^\top q_{\mathcal{M},i} \boldsymbol{y} + b_{\mathcal{M},i} \geq 0\}. \tag{7.6}$$

Let us consider the general model

$$Y = \mu + \sigma^2 \epsilon\,, \tag{7.7}$$

where $\epsilon \sim \mathcal{N}(0, I_n)$ and $\mu \in \mathbb{R}^n$. Characterizing the set $\mathcal{E} = \{\boldsymbol{y} : m(Y) = \mathcal{M}\}$ allows us to answer a variety of statistical inference questions about the true signal μ and its association with the different kernels, conditional to the fact that a given set of kernels $\mathcal{M}$ has been selected.

For example, testing whether $s(\boldsymbol{K}, \mu) = 0$ for a given kernel $\boldsymbol{K} \in \mathcal{M}$, or for the sum of kernels $\boldsymbol{K} = \sum_{\boldsymbol{K}' \in \mathcal{M}} \boldsymbol{K}'$, is a way to assess whether $\boldsymbol{K}$ captures information about μ. This is the test carried out by Yamada et al. [274] to test each individual kernel after selection by marginal HSIC screening.

Alternatively, to test whether a given kernel $\boldsymbol{K} \in \mathcal{M}$ has information about μ not redundant with the other selected kernels in $\mathcal{M} \setminus \{\boldsymbol{K}\}$, one may test whether the prototype of μ built from all kernels in $\mathcal{M}$ is significantly better than the prototype built from $\mathcal{M} \setminus \{\boldsymbol{K}\}$. This can translate into testing whether

$$s\left(\sum_{\boldsymbol{K}' \in \mathcal{M}} \boldsymbol{K}', \mu\right) = s\left(\sum_{\boldsymbol{K}' \in \mathcal{M}, \boldsymbol{K}' \neq \boldsymbol{K}} \boldsymbol{K}', \mu\right).$$

Such a test is performed by Loftus and Taylor [152] and Yang et al. [277] to assess the significance of groups of features in the linear setting, using the projection prototype.

In general, testing a null hypothesis of the form $s(\boldsymbol{K}, \mu) = 0$ for a positive quadratic form s can be done by forming the test statistic $V = ||h(\boldsymbol{K})\boldsymbol{y}||_2^2$, and studying its distribution conditionally on the event $\boldsymbol{y} \in \mathcal{E}$. That $\mathcal{E}$ is an intersection of subsets defined by quadratic

constraints (Eq. (7.6)) can be exploited to derive computationally efficient procedures to estimate p-values and confidence intervals.

The techniques of Loftus and Taylor [152] and Yang et al. [277] can be directly applied to our setting when $h(\boldsymbol{K})$ is a projection matrix, as with the KPCR prototype. For more general $h(\boldsymbol{K})$ matrices, these techniques need to be adapted; one way to proceed is to estimate the distribution of V by Monte-Carlo sampling. We proposed in Slim et al. [229] a constrained Monte-Carlo hit-and-run sampler [22] based on the hypersphere directions algorithm [24].

Alternatively, Reid, Taylor, and Tibshirani [205] propose to test the significance of groups of features through prototypes, which they argue uses fewer degrees of freedom than statistics based on the norms of prototypes, which can increase statistical power. We have also adapted this idea to the case of kernels.

7.2.4 *kernelPSI*

We refer to the kernel selection and inference procedure descibed above as *kernelPSI*. kernelPSI is available as an R package from CRAN [226].

In Slim et al. [229], we limited ourselves to kernels corresponding to predefined groups of features, the QKAS defined by the biased HSIC estimator of Gretton et al. [86], and testing the association of the sum of selected kernels with the phenotype.

Our experiments on synthetic data show the statistical validity of kernelPSI. In addition, we observe that both kernelPSI and SKAT [268], which are both kernel-based procedures, are superior to linear alternatives. Moreover, methods selecting fewer kernels enjoy greater statistical power, and adaptive methods tend to select too many kernels. A representative example of these experiments is presented on Figure 7.2, in which we plot the evolution of the statistical power as a function of the effect size. Here the effect size θ is the magnitude of the coefficient applied to the kernels we chose as causal in our simulations. The statistical power is measured as the recall of the experiment, that is to say, the proportion of causal kernels properly identified as such.

Finally, preliminary applications to GWAS on *Arabidopsis thaliana*, using the Identical By State kernel (see Section 1.2.6)over clusters of SNPs considered to be in linkage disequilibrium, indicate that kernelPSI has the power to detect relevant genes in GWAS and are complementary to existing approaches. We are currently conducting a more in-depth study of the application of this methodological contribution to actual GWAS data sets.

7.3 Conclusion and perspectives

In this chapter, I have shown how kernels can be used to perform nonlinear feature selection in high-dimensional data. The first approach I presented, HSIC lasso, can be seen as a specific case of the relaxed version of SConES, where the score of association between a feature and the phenotype is computed using an HSIC estimator, and the network constraint is given not by an a priori biological network but by the data-driven HSIC between features. HSIC lasso is therefore used to select nonredundant informative features.

The second approach, kernelPSI, also uses HSIC (or more generally a quadratic kernel association score), but to determine the association between a group of features and the

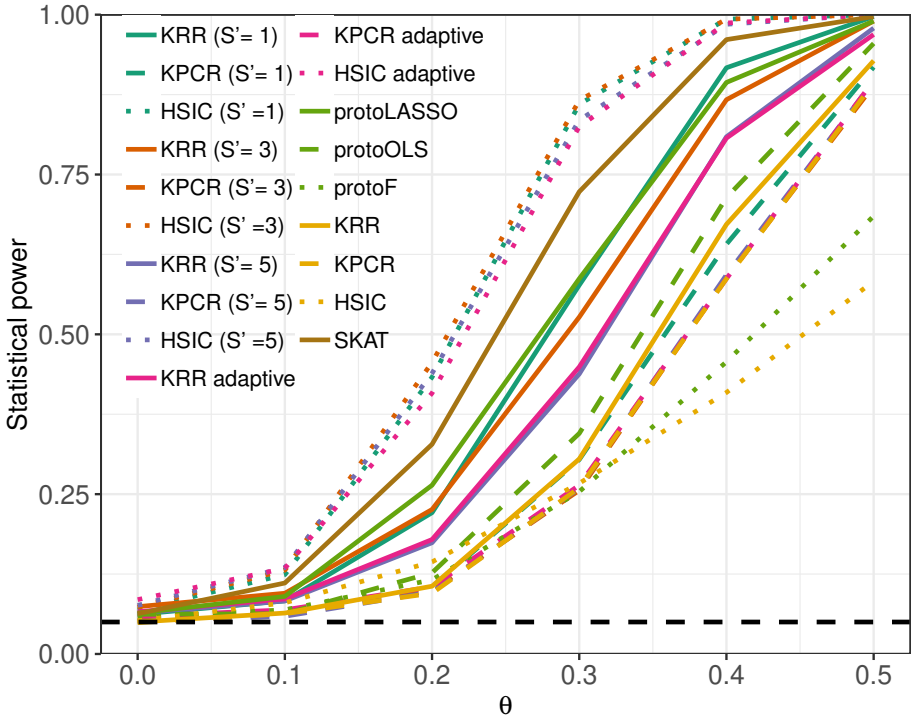


Figure 7.2: Evolution of the statistical power of kernelPSI variants and benchmark methods, using Gaussian kernels for simulated Gaussian data, as a function of the effect size θ. The kernelPSI method with HSIC kernel and set numbers of kernels to select ("HSIC ($S' = 1$)", "HSIC ($S' = 3$)", "HSIC ($S' = 5$)") enjoy slightly better power than the adaptive version ("HSIC adaptive"). "HSIC" is the independence test proposed by Gretton et al. [85]. The KRR and KPCR are variants of kernelPSI using other quadratic kernel association scores than HSIC. If no indication appears in parentheses after their names, no feature selection is performed and the statistical significance of a prototype built using all features is used. Finally, "protoLasso" is the method proposed in Reid, Taylor, and Tibshirani [205] and can be considered a linear alternative to the KRR. "protoOLS" and "protoF" are two variants that do not perform feature selection but simply evaluate the statistical significance of prototypes built from the full model.

phenotype. There is no notion of redundancy here. A main advantage of kernelPSI is to make use of the post-selection inference framework to provide valid inferences and hence p-values for the selected features.

While our kernelPSI contribution was very methodological in nature, we are currently exploring its application to GWAS data sets.

One of the limitations of such approaches is their somewhat black-box nature: while we can use them to find which features act nonlinearly on the phenotype, the exact nature of this action is not accessible. In addition, they suffer from computational limitations, in particular as the number of samples grows, something we now see happening for certain phenotypes such as height or body-mass index. The block HSIC estimator [283] alleviates this issue, as we demonstrated with the block HSIC lasso. In addition, we have started experimenting with GPU implementations of HSIC computations.

8

LESSONS LEARNED

To conclude the exposition of the research I have been conducting, I will summarize in Section 8.1 my methodological contributions to the fields of biomarker discovery and precision medicine. Most of these contributions can be framed in the context of feature selection in high-dimensional settings.

In addition, I would like to share some lessons I have learned – and keep learning – about the application of machine learning to bioinformatics problems. These lessons can be summarized as three messages: (1) the evaluation of machine learning models must match the problem at hand (Section 8.2); (2) more complex machine learning algorithms do not necessarily yield better models (Section 8.3); (3) the hardest part is often to build the data set (Section 8.4).

To the machine learning practicioner, these three statements may appear rather trivial. They are, after all, among the key points I try to convey in any of my machine learning lectures. However, their execution in bioinformatics research is not always so straightforward.

The insights I present in this chapter are born from the work I have presented so far, as well as additional published work that did not make it to the bulk of this document:

- S. Joshua Swamidass, Chloé-Agathe Azencott, Ting-Wan Lin, Hugo Gramajo, Sheryl Tsai, and Pierre Baldi. The Influence Relevance Voter: an accurate and interpretable virtual high throughput screening method. *Journal of Chemical Information and Modeling*, 49(4):756–766, 2009.
- S. Joshua Swamidass, Chloé-Agathe Azencott, Kenny Daily, and Pierre Baldi. A CROC stronger than ROC: measuring, visualizing and optimizing early retrieval. 26(10):1348–1356, 2010.
- Matthew A. Kayala, Chloé-Agathe Azencott, Jonathan H. Chen, and Pierre Baldi. Learning to predict chemical reactions. *Journal of Chemical Information and Modeling*, 51(9):2209–2222, 2011.
- Dominik Grimm, Chloé-Agathe Azencott, Fabian Aicheler, Udo Gieraths, Daniel MacArthur, Kaitlin Samocha, David Cooper, Peter Stentson, Mark Daly, Jordan Smoller, Laramie Duncan, and Karsten Borgwardt. The evaluation of tools used to predict the impact of missense variants is hindered by two types of circularity. *Human Mutation*, 36(5):513–523, 2015.
- Chloé-Agathe Azencott, Tero Aittokallio, Sushmita Roy, et al. The inconvenience of data of convenience: computational research beyond post-mortem analyses. *Nature methods*, 14(10):937, 2017.

8.1 Summary of research work

Because differences between patients in diagnosis, prognosis and response to treatment are partially due to genetic causes, enabling precision medicine requires the development of methods that can identify biomarkers that explain a phenotype of interest. Biomarker discovery can often be cast as a problem of feature selection in high-dimensional omics data sets, where the number of samples is usually small, which poses statistical challenges. To address these challenges, I have worked along three axes:

- The integration of prior biological knowledge, encoded as networks, to machine learning methods for biomarker discovery (Chapter 2);
- The development of multitask algorithms, which alleviate the data scarcity by jointly fitting models for related problems (Chapters 3 and 4);
- The development of methods for nonlinear feature selection, so as to depart from single-feature or additive models (Chapters 5 to 7).

Although feature selection in high dimension is the main focus of my work, I have also recently worked on graph node labeling, for the inference disease genes from gene-gene interaction network (Appendix A), or on supervised multitask learning, for the prediction of drug-protein binding (Appendix B).

8.1.1 *Using biological networks*

Biological systems are complex systems, in which components interact in synergistic, antagonistic, or redundant ways. Genetic alterations responsible for a particular phenotype can hence be expected to affect genes, either locally or remotely, that interact physically or along a molecular pathway.

Biological networks, which capture this complexity, are a rich source of information for mathematical modeling, thanks to their graph structure. In Chapter 2, I have shown how the penalized relevance framework we proposed can be used, with the SConES algorithm, to combine biological networks and classic tests of association to increase our ability to discover SNPs that explain a phenotype. In Appendix A, I have described ongoing work on applying recent developments in graph representation learning to the problem of prioritizing disease gene candidates using multi-layer biological networks.

While my work within the penalized relevance framework has focused on graph regularizers built from biological networks, this framework is amenable to other forms of regularization, and in particular group regularizers built, for example, from pathways.

8.1.2 *Multitask approaches*

While biological networks help us shape the space of features describing our data, increasing statistical power by essentially reducing the dimensionality of the problem, a complementary approach consists in increasing the sample size by means of jointly learning on multiple related but different tasks.

In the context of feature selection, jointly selecting features for all tasks reduces the features-to-sample ratio of the data, while keeping the particularities of each data set. I have described in Chapter 3 how we extended SConES to multitask settings, and in

Section 4.2 the algorithm we proposed to incorporate task descriptors to improve lasso-like multitask feature selection.

I have shown that using task descriptions can improve the quality of predictions in a mutltitask framework, whether based on lasso (Section 4.2) or on SVM (Appendix B). This allows us to make predictions for tasks for which no training data is available, which can prove useful, for example, for new drug candidates for which no experimental data has been collected yet.

8.1.3 *Nonlinearities*

Biological phenomena are often nonlinear, which suggests that one must incorporate interactive effects between genetic features in biomarker discovery models.

Unfortunately, the detection of two-loci interactions in GWAS requires a massive amount of computation, as the number of pairs of SNPs that need to be examined can be in the order of $10^{10} - 10^{14}$. I have presented in Chapter 5 how to use GPGPU to accelerate those computations in the case of linear models.

To address the statistical challenges that are also aggravated by the increased number of statistical tests to perform, we have proposed a targeted epistasis approach (Chapter 6) that searches for interaction between a specific SNP and the rest of the genome. This model is more flexible than pairwise interactions, as it can model interactions between the target SNP and several other SNPs jointly. In addition, our method gains power from not requiring the evaluation of main effects to test interactive ones, and corrects for linkage disequilibrium effects.

Finally, I have shown in Chapter 7 how to use kernels, in particular via the HSIC, first to model nonlinear dependencies between two variables, so as to select informative but non-redundant features (Section 7.1), and second to select groups of features nonlinearly associated with a phenotype (Section 7.2). In this second setting, the post-selection inference framework makes it possible to obtain p-values to quantify these associations.

Finally, with open science and reproducible research in mind, we have released code for each of these methodological contributions, as well as, when possible, data sets and scripts that can be used to reproduce the published results.

8.2 Proper evaluations are not always straightforward

Evaluating a machine learning model requires two things: an evaluation data set, and an evaluation metric. The evaluation data set should be separate from the training set, so as to avoid overfitting. The evaluation metric should be suited to the problem and informative about the model. While these two concepts sound rather obvious, I have often found myself in situations where there are not straightforward to implement, and will detail some of these scenarios below.

8.2.1 *Beware circular reasonings*

Separating training and evaluation data implies ensuring that the evaluation data set has not been used at any point of the training process, including the model selection stage.

This may not be straightforward when the public benchmarks available for the evaluation of predictive tools overlap with the databases used to build these tools. We have encountered this situation when trying to propose – and therefore evaluate – new tools for predicting whether a missense single nucleotide variant is more likely to be pathogenic or neutral [87]. Many tools, such as SIFT [180], PolyPhen-2 [2] or CADD [122], have been developed for that purpose. Nair and Vihinen [175], having noticed that some of them had been developed using databases also typically used for benchmarking, endeavored to build independent benchmark data sets. However, there can still be some overlaps (mainly in neutral variants). Furthermore, not all authors disclose the variants they used to build their tools. It is impossible to guarantee that an evaluation data set does not contain some of these variants, and hence to guarantee fairness when comparing these tools against others.

A more subtle issue arises when the set of examples that are already annotated is not a uniform random sample of the population of interest. Still on the topic of pathogenic variants, we found [87] that many annotations available in data bases are obtained from predictions – either from well-established tools or, more simply, by annotating all variants in a gene with the same label as the only one in this gene to be supported by biological evidence. This means that one can very efficiently leverage the annotation of other SNVs in the same gene to build what will appear to be a very accurate tool; but there is no guarantee that this tool will perform well on new variants.

A similar problem arises in the disease gene prioritization problem I described in Appendix A: the genes that are labeled as disease genes may have been investigated following up on computational predictions, or more simply because they are connected to other disease genes. The distribution of positive labels is therefore likely to be biased by network topology. In these circumstances, one may argue that the performance of network-based methods is partially due to the way genes have been investigated experimentally. These concerns can be related to those related to the limitations of the guilt-by-association hypothesis discussed by Gillis and Pavlidis [78].

8.2.2 *Realistic evaluation data sets*

An additional difficulty I have encountered many times in bioinformatics is making sure that the evaluation data set, in addition to not being biased as described above, actually corresponds to the envisioned application.

This is a point we illustrated in our work on drug-protein binding prediction [196], as exposed in Section B.2 of Chapter 4. When applying drug-protein binding prediction algorirthms at a proteome-wide scale, so as to discover secondary targets of a given molecule, one encounters many orphan proteins, that is, proteins for which no ligand is known. We showed that making good predictions for these proteins is notably harder than for proteins that have several known ligands, and, unsurprisingly, even more so if they are dissimilar to all non-orphan proteins of the training set [196]. For this reason, we had to carefully construct realistic evaluation data sets, that mirror this orphan situation, to avoid evaluating our methods on settings that are “too easy”.

8.2.3 *Appropriate evaluation metrics*

Finally, one must not forget to use appropriate evaluation metrics. Most feature selection methods in machine learning have been developed to reduce the size of the training data

and improve the predictive quality of supervised learning models. However, in biomarker discovery, the ability to interpret the selected features is crucial. This means that the *stability* of feature selection methods must be taken into account, as we did for instance in Azencott et al. [11] or Bellon, Stoven, and Azencott [23]. This aspect is still, to my eyes, underestimated in many published studies.

Another situation in which evaluation metrics must be carefully considered is in virtual screening, that is to say, the exploitation of an exploratory *in vitro* screening to rank unscreened compounds according to their activity towards the same target. The Influence Relevance Voter (IRV) algorithm we developed for that purpose [237] retrieves up to three times as many active compounds among those in the first percent of the ranked list returned. Evaluating IRV led us to studying how to assess an algorithm's ability for *early recognition*, that is to say, its ability to put active compounds at the very top of the ranked list it returns, irrespective of how the inactive compounds are ranked. A tool with good early recognition capacities allows experimentalists to focus their efforts on a small number of compounds. To address this question, we have developed CROC (Concentrated ROC), an extension of ROC curves for the quantification, visualization and optimization of early recognition [236].

8.3 Complex models may not be better

Many machine learning researchers share a tendency to want to build more complex models than those that already exist to address a particular problem, hoping their model will outperform the state of the art. This tendency is both a curse and a blessing: while this is, of course, how we get better solutions for existing problems, it often happens that the more complex model is, in fact, not making any better predictions.

It is along those lines that Haury, Gestraud, and Vert [99] showed that the most satisfactory method to extract molecular signatures from gene expression datasets, among the 32 they evaluated, were not the wrapper or embedded methods, but a simple Student's t-test. It does not mean we should stop investigating complex models, but rather that we should pay careful attention to comparisons with reasonable baselines – such as a univariate statistical test. In Azencott et al. [11], we compared SConES to a univariate test for all the reported metrics, and found that, on the problems we were investigating, our algorithm does indeed, unlike the lasso, outperform this baseline.

8.3.1 *Components of a complex model should be evaluated separately*

An additional – and unpublished – example comes to mind. The work I presented in Chapter A was not my first attempt at addressing the question of prioritizing disease genes using gene-gene networks. In 2011, I looked into a similar question, where the problem was formulated as a link prediction problem on an undirected graph containing two types of vertices: some representing genes, and others representing diseases. Edges between two genes correspond to a protein-protein interaction network. Edges between two diseases correspond to a measure of disease similarity. Finally, diseases are connected to their disease genes, and it is this relationship that one wishes to complete. This setting is that of several publications [67, 173].

I was interested in proposing a kernel-based approach to this problem, and defined the kernel between two (disease, gene) pairs as a function of a kernel between two diseases

and a kernel between two genes. The disease kernel was derived from the disease similarity used by other authors, whereas I had several ideas of kernels between graph nodes that could be used for the gene kernel.

However, I quickly realized that, in practice, the gene kernel had very little importance on the results, which were overwhelmingly dependent on the disease kernel. I then replaced, in previously published methods, the protein-protein interaction network with a trivial network with no edges. This only had a very small effect on the overall performance of these methods. Hence, building more and more complex gene kernels had no use for that problem. Another lesson drawn from this experience is that it is important to evaluate separately the different components of a complex models, again creating meaningful baselines in which all components but one are replaced by trivial baselines.

This point does not invalidate the work I presented in Chapter A: the data sets are very different, and the fact that the protein-protein interaction network brings little information with respect to the disease similarity data does not mean that the gene-gene network is irrelevant when focusing on a single disease (or disease family).

8.3.2 *Deep learning will not cure all that ails you*

Deep learning has recently lead to major breakthroughs in computer vision, natural language processing, or speech recognition, thanks to its ability to learn powerful representations for data that live on a grid-like structure. These inroads have ignited the interest of the community for developing deep learning models for data that live on graphs. Unfortunately, these advances do not always readily translate in progress for all application areas.

The work I presented in Appendix A illustrates this point: we did not find deep learning approaches for node labeling to perform better than the classic label propagation methods for this application. Moreover, the slight gain in performance we observed in some settings was obtained at the expanse of requiring much larger computational resources.

Benoît Playe made similar observations in the context of drug-ligand prediction. In work he conducted following the study I presented in Appendix B, and which is described in his PhD thesis [195], he thoroughly investigated the application of the latest developments in graph representation learning and neural network architectures for chemogenomics. Although neural networks sometimes outperform the kernel-based method we proposed in some settings, tuning their parameters is much more computationally intensive. Overall, his results strongly suggest that further investigating neural network architectures is not a promising direction to improve drug virtual screening in a chemogenomics framework.

8.4 The hardest part may be to build the data set

Most data scientists will tell you that most of their time is spent on data pre-processing. This is indeed very much the case for bioinformatics applications as well, and I have encountered several occasions where building an appropriate data set was the most challenging part of the project.

This issue often ties in with the aforementioned difficulty to build a proper evaluation framework (see Section 8.2). Indeed, in both the variant pathogenicity prediction study [87] or in the drug-ligand prediction project [196] I discussed, figuring out how to separate our

data in subsets appropriate to properly answer the questions we had was a large part of our work.

Furthermore, collecting the data to start with can turn out to be the most difficult – and the most crucial – part of the work of a machine learner in bioinformatics.

8.4.1 *Predicting organic chemistry reactions*

Towards the end of my PhD, I have worked on molecular synthesis, one of the applications being to assist chemists in finding efficient and cost-effective ways of synthesizing drugs or candidate drugs. More specifically, I have studied the prediction of the course taken by organic chemical reactions. Whereas the literature at the time was focused exclusively on expert systems, Matt Kayala and I wanted to take a machine learning approach to address this problem. We decided to model chemical reactions by decomposing them in elementary mechanisms, from which we could derive atom features that we could in turn use to train models to detect active sites, and to rank active sites by reaction favorability.

Our main contribution was to build the first data set of elementary mechanisms, which we then made publicly available [118]. This allowed us to build the first reaction predictor capable to generalize to reaction types not included in its training set [118].

8.4.2 *The inconvenience of data of convenience*

More generally, data collected in biological and medical studies are often generated without the input of those who will later analyze it. Computational analyses are therefore, in the words of statistician Ronald Fisher, mostly performed "post-mortem".

I took part in two DREAM challenges that illustrate the difficulties in this process. In DREAM challenges, computational biologists around the world attempt to solve a biological or medical problem using the provided "data of convenience". In the Rheumatoid Arthritis Responder Challenge, where the goal was to predict drug response from patient genome, using the SNP data did not improve predictions over those obtained using a handful of clinical predictors [223]. In the Toxicogenetics Challenge, SNP data by themselves were not predictive, but the RNA-seq data were. However, RNA-seq data was only available for 38% of the patients [63].

These situations may arise because computational approaches are just not good enough yet for the task. However, that none of several dozen independent expert teams were successful in solving the problems using the same data suggests that, instead, more or different kinds of data may be needed. How can one efficiently determine which data we need to, rather than can, measure to accelerate scientific discovery?

Following our belief that computational biologists can contribute to model-driven experimental research, we have launched in 2016 the Idea DREAM Challege [13]. Participants were asked to propose biomedical research questions for which computational models have exploited available data to the limit, and are ready to guide new data collection efforts to move the field forward. Through peer review and discussions among participants, we selected two winning ideas. We have matched the winning participants with wet-lab researchers to generate the necessary data.

It is too early yet to see where these efforts have led, but we hope that the Idea DREAM Challenge is just the beginning of many more endeavors in which data analysts and computational biologists can be actively engaged in all stages of the scientific process.

9

PERSPECTIVES

To pursue my work in the development and application of machine learning tools for precision medicine, I am planning on following four axes: (1) addressing GWAS-specific questions, such as linkage disequilibrium, population structure, or the construction of SNP-SNP networks, in line with the methods I have already proposed (Section 9.1); (2) focusing on the *stability* of feature selection methods (Section 9.2); (3) integrating multiple types of data, including from electronic health records (Section 9.3); and (4) data privacy (Section 9.4).

9.1 GWAS-specific questions

In the course of my work on GWAS data, several topics have arised that I would like to explore in more depth.

9.1.1 *Linkage disequilibrium*

The first of these topics is *linkage disequilibrium* (see Section 1.1.2). How can we properly account for the non-random correlations between features in our methods? In epiGWAS [228], we used propensity scores to model them explicitly. In work such as SConES [11], linkage disequilibrium is not explicitly accounted for; the sparsity constraints will tend to enforce the selection of a single SNP out of an LD block, while the network constraints will rather favor selecting several SNPs in linkage disequilibrium if they are connected on the underlying biological network and all contribute to the signal. In block HSIC lasso [46], we explicitly chose to select non-redundant features, which results in picking, again, a single SNP per LD block.

Which of these approaches is more desirable from a practical point of view? Should we maybe only consider association at the level of an LD block, rather than at the variant level? This would probably improve the algorithms stability.

Another interesting question is that of the definition of LD blocks. Indeed, beyond LD pruning and LD clumping, the prototypes we proposed in Slim et al. [229] could be used to form LD blocks.

9.1.2 *Population structure*

Population structure is another important concern in GWAS. Indeed, some SNPs may appear to be associated with the phenotype when, in fact, they are associated with a subpopulation of the data in which the trait is more present. In addition, the SNPs associated with a phenotype may be different in different populations. Furthermore, subpopulations may be due to phenotype heterogeneity; whether in cancer or psychiatric disorders, two patients with the same symptoms may in fact suffer from a different disease.

In order to avoid confounding by population structure, several approaches are possible:

- separate the data in homogeneous subpopulations. While this reduces the number of samples, statistical power may actually be improved if population-specific SNPs have strong effects.
- incorporate population structure, typically identified by a principal component analysis, as covariates (usually refered to as axes of genetic variation) in models [190, 197]. This is the approach we have followed in the work on *Arabidopsis thaliana* I have presented here.
- model population structure using linear mixed models [265]. While I find this solution more satisfying, it poses numerous statistical challenges [234] and its application to my work is not straightforward.

I am interested, however, in whether multitask models (see Chapters 3 and 4), where each of the tasks corresponds to a subpopulation (or a subphenotype), can be successfully applied to this question.

9.1.3 *SNP-to-gene mapping*

I have alread mentioned in Section 2.3.7 of Chapter 2 the difficutly of mapping SNPs to a gene, whether for functional interpretation or for building SNP-SNP networks. We are currently investigating this aspect (see Duroux et al. [61]), and comparing empirically on several GWAS data sets the impact and meaning of the various possible mappings.

It is important here to remember that most SNPs discovered by GWAS are in intergenic regions [162]. It is therefore important to find ways to map intergenic SNPs to genes. Both expression quantitative trait loci (eQTL) information, linking SNPs to genes whose expression they partially regulate, and chromatin interaction matrices, which allow to link SNPs to genes they are physically in contact within live cells, can help with this endeavour. While neither type of information was available in data bases at the time we developed SConES, this has now changed and both approaches have been used for the functional interpretation of GWAS results [264].

9.2 Stable nonlinear feature selection

Although I have insisted in several places on the importance of the stability of feature selection methods, none of the approaches I have proposed so far directly enforce this stability. Our experiments show that using biological networks or solving biomarker discovery problems in a multitask fashion does tend to improve stability. I believe this to be a consequence of their improvement of both power and false discovery rate, and would like to achieve stability in a more explicit fashion.

9.2.1 *Stability*

I am therefore planning to further develop the penalized relevance framework so as to propose more stable network-guided feature selection approaches. Combining the strengths of multiple feature selectors created from bootstrapped samples can guarantee more stable sets of features than the individual selectors themselves [164, 220]. We have used these strategies with some success in the context of GWAS studies [23, 228], and expect them to be effective in the regularized relevance framework, which lends itself to explicitly including stability in the objective function.

Given K bootstrapped samples of the data $(\boldsymbol{X}, \boldsymbol{y})$, which can be choosen according to the paired sample paradigm of [220], yielding K different relevance functions $R_1, R_2, \ldots, R_K$, we can look simultaneously for K sets of features $\mathcal{S}_1, \ldots \mathcal{S}_K$ that satisfy the objective in Eq. (2.4) while being similar. This yields Eq. (9.1), which becomes equivalent to the multitask formulation in Eq. (3.4). In practice, solving it will require the creation of a meta-network in which each feature is duplicated as many times as bootstrap iterations.

$$\underset{\mathcal{S}_1,\ldots\mathcal{S}_K \subseteq \mathcal{V}}{\arg\max} \sum_{k=1}^{K} R_k(\mathcal{S}_k) - \lambda\Omega(\mathcal{S}_k) - \mu \sum_{k<k'} |S_k \triangle S_{k'}|. \tag{9.1}$$

One can also directly enforce $\mathcal{S}_1 = \cdots = \mathcal{S}_K$.

An important question I think is that of the level at which we expect stability. Should our methods systematically select the same SNPs? Or SNPs within the same gene? Within the same LD block? Within the same pathway? Boyle, Li, and Pritchard [28] advance the hypothesis of an "omnigenic" model. They propose that most heritability lies within regulatory pathways. While this confirms that it is meaningful to use networks, or pathways, to guide biomarker discovery, it also suggests that a large number of variants influence the phenotype. How can one then define a causal SNP when all variants are related to phenotype? The omnigenic model hypothesis suggests looking for stability at a scale no smaller than that of the regulatory pathway.

9.2.2 *Nonlinear models*

Our work on nonlinear feature selection has only just started, and there are many avenues I would like to explore in the future.

One of them is the application of rule-based models, which have been successfully applied to very high-dimensional microbiome data [59], to the gene expression or SNP data we are encountering in human genomics. The work of Drouin et al. [59] gives strong performance guarantees, and the rule-based models are very selective and highly intepretable. I am curious to see how good these bounds are for human GWAS of complex traits.

Recent work in the field of safe screening rules [178] have allowed to greatly accelerate the lasso and some of its variants, as well as the associated hyperparameter selection process. I am curious to explore whether those ideas can be applied to the penalized relevance framework. Along those lines, Le Morvan and Vert [136] proposes a working set method for applying the lasso with two-way interactions.

We have recently started exploring post-selection inference to provide p-values for nonlinear feature selection methods (Section 7.2). Recent statistical developments, such as knock-off filters [33], or multi-layer p-fitlers [20], are also an interesting avenue to explore.

9.3 Multiview feature selection

The methods I have presented in this document are meant to be applied a single type of molecular data. However, one can often have access to several different types of measures, ranging from molecular data to images and text, for the same samples. These different data types correspond to different *views* of the data. Integrating them in a single analysis could increase our ability to discover relevant biomarkers [79, 157].

9.3.1 *Multi-omics data integration*

In the case of multi-omics data, considering for example SNPs, gene expressions, and methylation patterns for the same samples, integration is facilitated by our ability to link together omics features of different nature. This can be done through position on the genomic sequence, or the SNP-to-gene mapping techniques discussed in Section 9.1.3.

Again, the penalized relevance framework allows for the integration of multiple views of data. If D is the number of available views, and if the same features (e.g. genomic loci) are available across all views, Eq. (2.4) can become

$$\arg\max_{\mathcal{S}\subseteq\mathcal{V}} \sum_{d=1}^{D} R_d(\mathcal{S}) - \lambda\Omega(\mathcal{S}), \tag{9.2}$$

and can again be solved using a maximum flow algorithm. A variant where views can be ignored if they are not relevant is also possible. One can also contemplate formulations that allow for the selection of one set of features per view, with the help of a second regularizer that enforces that related features are selected simultaneously across views.

9.3.2 *Multi-modality*

In addition to genomic data, patients can also be described at the cellular or systematic scale by data of very different nature: time series, images, free texte, etc. The joint analysis of these rich and complementary data can be expected to further our understanding of human diseases and responses to treatment.

I am interested in performing feature selection, separately or jointly, on time series representing patient trajectories [21] or accelerometer data [259]; on free-form medical text, which can describe medical interventions [174] or clinical trials; or lab test results.

I am currently working on several projects in that direction: matching patients with clinical trials; discovering prognostic factors in patients treated with neoadjuvent chemotherapy from lab test results and anatomo-pathology reports; or identifying interactions between comedications and recurrence-free survival in breast cancer from public health data.

While these projects may require new methodological developments, they also fall within my recent efforts to also contribute to translational research. In working closely with clinicians or biologists on specific questions of their devising, I am hoping to both feed my long-term vision about necessary methodological developments, and to maybe see a shorter-term impact of my work on patient health.

9.4 Data privacy

Finally, in a scarce data context, one cannot argue against the need to combine data sets produced within different studies in different labs. Federated learning [30], however, is hindered by, on the one hand, batch effects that may bias analyses if they are not properly accounted for, and, on the other hand, by data privacy concerns. I have recently started studying the state of the art in genomic data privacy [10], and would like in years to come to adapt some of my work to the differential privacy framework.

APPENDIX

A

DISEASE GENE PRIORITIZATION

Chapter 2 described how to use biological networks to help biomarker discovery formulated as a feature selection problem on a set of samples that have been both genotyped and phenotyped. A very different approach consists in using gene-gene interactions to prioritize candidate genes, that is, identifying computationally the genes most likely to be linked to a disease.

The underlying hypothesis is the same: if a disease results from the perturbation of a molecular pathway by a genetic dysfunction, then dysfunctions along that pathway may produce similar phenotypes. In other words, genes responsible for similar diseases are likely to participate in the same interaction networks [75]. This principle is sometimes refered to as guilt-by-association: genes that are interacting are more likely to share function.

In this appendix, I will describe in Section A.1 how the problem of disease gene prioritization can be formulated as a node labeling problem on multilayer graphs, that is, using multiple networks over the same set of nodes. In Section A.2, I will give a brief overview of state-of-the-art label propagation approaches to this problem. I will then describe in Section A.3 recent developments in the field of deep learning for graphs. Finally, I will describe in Section A.4 our first attempts to apply these techniques to address disease gene prioritization.

The work in this appendix, which was jointly conducted with The work I present in this part was conducted jointly with Stefani Dritsa, Thibaud Martinez, Antonio Rausell, and Weiyi Zhang, has not been published yet, but we recently presented a first version at the Jobim conference:

> Stefani Dritsa, Thibaud Martinez, Weiyi Zhang, Chloé-Agathe Azencott, and Antonio Rausell. Prediction of candidate disease genes through deep learning on multiplex biological networks. 20th Open Days in Biology, Computer Science and Mathematics (poster), 2019.

Because both *biological networks* and *artificial neural networks* appear in this appendix, I will here use "graphs" to refer to biological networks and "networks" to refer to artificial neural networks, unless otherwise specified.

A.1 Disease gene prioritization as a node labeling problem

In this appendix, we are considering a large set of genes, as well as a disease, or family of diseases, for which a number of disease genes are known. Our goal is to find which other genes are the most likely to be also associated with this disease. To that end, we want to use multiple biological networks, representing relationships of different natures between these genes.

As all the biological networks share the same set of nodes, we model them as a single multi-layer graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of L layers and n vertices (or nodes). An edge $e \in \mathcal{E}$ is a quadruplet (i, j, l, w) where $i \neq j \in \mathcal{V}$ are the two vertices it connects, $1 \leq l \leq L$ indicates to which layer it belongs, and $w \in \mathbb{R}$ is its weight. Vertices can also have attributes, in which case they are described by $\boldsymbol{X} \in \mathbb{R}^{n \times d}$. Finally, we call $\mathcal{P}$ the vertices labeled positively, that is to say, those corresponding to the genes known to be associated with the disease of interest. Our goal is to find a scoring function $s : \mathcal{V} \to [0, 1]$ that gives the likelihood for a vertex to be positive. Hence, our task is indeed a node labeling task.

Positive-unlabeled learning

This task can be described as a *positive-unlabeled learning* problem: no negative labels are available for training, and our goal is to identify positive nodes in the unlabled set $\mathcal{U} = \mathcal{V} \setminus \mathcal{P}$. To address this issue, we follow a bagging approach similar to that proposed by Mordelet and Vert [172], we repeatedly sample boostrap samples from the unlabeled data, and train a binary classifier to discriminate the bootstrap sample from the positive data instances of the training set. We then aggregate the predictions of the trained classifiers simply by computing the mean of the individual predictions.

Model evaluation

Having labels for positive instances only also affects the way we can evaluate our methods, and there is currently no general performance evaluation strategy to compare gene prioritization methods [89]. Indeed, many authors [7, 123] build ROC curves assuming that all unknown genes from the test set are negatives.

We advocate instead in favor of an evaluation method that takes into account the uncertainty about the label of non-positive data. Such an evaluation is difficult to conduct without making assumptions, such as the proportion of positively labeled examples among the unlabeled ones [43, 110], which do not apply to our setting. We adopt the same evaluation approach as in Mordelet and Vert [173] and Valdeolivas et al. [251], and evaluate for each fold of a leave-one-positive-out cross-validation the rank of the positive sample excluded from the training set among all samples of the test set – the lower the better.

A.2 Propagation-based methods

Numerous gene prioritization methods leverage molecular interaction networks to extract evidence for genes susceptible to be involved in a disease [31]. Among them, propagation methods, or diffusion methods, which propagate information about known disease genes (or "seeds") along the edges of the graph [54], are the more popular.

Most of these methods are based on variants of *random walks:* the probability that a vertex is positive is evaluated as the probability that random walkers released from seed nodes arrive on this vertex [153].

If the graph is fully connected, the random walk will converge to a steady state in which all vertices have the same probability. To avoid this situation, in a random walk with restart, at each step, the random walker is given probability $0 < r < 1$ to start again from a seed. Köhler et al. [123] and PRINCE [253] use random walks with restart for gene prioritization on protein-protein interaction networks.

Finally, Li and Li [144] and Valdeolivas et al. [251] extend the random walk with restart to multi-layer graphs. Here, the random walker can, at each step, jump to the same node in a different layer with probability $0 < 1 < \delta$.

A.3 Disease gene prioritization with deep learning on multi-layer biological networks

The recent enthusiasm for the data representation capabilities of deep learning methods have led to many developments in the domain of deep learning for graphs, with applications ranging from social networks to information retrieval and bioinformatics. Among those methods, two seem particulary relevant to address node labeling problems: relational graph convolutional networks and node embeddings.

Relational graph convolutional networks

Inspired by the success of convolutional neural networks on images, many of the recent developments in the application of deep learning to graphs rely on adapting the concept of convolutions to graphs. Among those, spectral-based methods [32, 56] operate on the spectrum of the graph, while spatial-based methods operate directly on the nodes and the edges of the graph [270]. In this last category, *Graph Convolutional Networks*, or *GCN* [121], are computationally efficient and show robust performance in practice [221]. They were first applied to semi-supervised learning on the nodes of a graph, and therefore seem already well suited to our problem.

The *Relational Graph Convolutional Network*, or *RGN* [216], extends the GCN to relational graphs, which are directed, non-weighted graphs containing multiple types of edges describing different types of relationships between the vertices. Multi-layer biological networks are hence similar in concept. Their undirected edges can be modeled using two directed edges (one in each direction). However, the method cannot be directly applied to weighted edges.

Node embeddings

While GCN and RGCN learn graph representations that are adapted to the downstream classification task, another approach to representation learning in deep learning consists in learning task-independent representations, in an unsupervised way, without optimizing for a specific downstream supervised task. In this spirit, graph embedding methods aim at learning generally useful representations of graphs or graph nodes, preserving the information contained in the graph structure; those embeddings can then be used as input to machine learning tasks [94].

In particular, *node embedding methods* encode nodes as low-dimensional vectors that preserve information about the node position in the graph and its local neighbourhood. The most popular node embedding methods, DeepWalk [193] and its extension node2vec [88], are based on random walks. They build upon previous developments in the field of natural language processing, in particular on the skip-gram (or word2vec) model [168]. The underlying idea of word2vec is that words with a similar meaning should have similar vector representations. DeepWalk and node2vec equivalently maps topologically similar nodes to close embeddings. Node similarity is computed through random walks, and can capture

both homophily (nodes belonging to the same community) and structural equivalence (nodes playing the same role).

Deep learning on weighted, attributed multi-layer graphs

Our goal is to apply these techniques to weighted and attributed multi-layer graphs.

For that purpose, we adapted the matrix normalization step of RGCNs Kipf and Welling [121] to weighted adjacency matrices, so as to make them amenable to weighted graphs.

Several approaches have been proposed to extend node2vec to learn node embeddings for multi-layer graphs [149]. Among those, OhmNet [288] uses a hierarchical regularizer to tie node embeddings across layers. Multi-node2vec [267] and MultiNet [16] use random walks across layers as node sampling strategies. We propose an alternative approach, which consists in using as input to the skip-gram model random walks generated on each separate layer of our multi-layer graph.

Finally, several approaches, such as GAT2Vec [222] or GraphSAGE [93], extend node2vec to graphs with attributed nodes.

To perform supervised classification on the generated node embeddings, we used both a logistic regression and a multi-layer perceptron (MLP).

A.4 Preliminary experimental results

This study led us to propose a computational framework for network-based gene prioritization. This library provides an efficient and convenient way to evaluate a range of network propagation and artificial neural network methods on the task of disease gene prioritization, and ensures the reproducibility of the experiments. A first release is available at `https://github.com/RausellLab/Tiresias`.

We ran experiments on a data set containing 288 known primary immuno deficiency genes [72] and four biological networks (protein-protein interactions, co-expression [240], whole blood regulation [211] and Marbach immune organs regulation [160]), spanning a total of 18 842 nodes.

Our results indicate that propagation-based methods perform better than artificial neural networks. This could be due to our failure to expand enough resources and time to optimize the hyperparameters for the neural network approaches. This raises an interesting question on how much resources we are ready to devote to a potential increase in performance.

We note, however, that the predictions of propagation-based methods, on the one hand, and artificial neural networks, on the other hand, are only weakly correlated. This suggests that the two families of approaches capture complementary information, and that a hybrid method could succeed in creating a classifier with better than state-of-the-art performance.

Although combining biological networks leads to better performance than working on a single of these biological networks, we fail to see a clear improvement between adapting methods to multi-layer graphs and merely merging the four biological networks by collapsing their edges.

Finally, our attempts to include node attributes, that is to say, features describing the genes independently from the biological networks, in the models were also not fruitful. It is possible that our attributes were poorly chosen; that the information carried by the node attributes is redundant with the structural information already present in the networks; or that the number of attributes (8) was too small with respect to the size of our embeddings.

A.5 Conclusion

In this appendix, I have shown how biomarker discovery using biological networks can be formulated, not as a feature selection problem on genotype-phenotype data, but as a semi-supervised node labeling problem in the contexte of disease gene prediction.

While recent developments in devising artificial neural networks that learn on graph-structured data have promising applications to the problem of disease gene prediction, our results are mixed. These methods are indeed much more computationally intensive than their classic propagation-based counterparts, and we did not see an increase in performance. An interesting aspect is the complementarity of the predictions of neural networks versus propagation methods, which opens the door to a potentially powerful hybrid approach.

The work I have presented in this appendix is still preliminary, and many experiments remain to run – in particular, experiments on additional data sets and validation on external data – to corroborate our conclusions.

B

EFFICIENT MULTITASK CHEMOGENOMICS

The work I have presented in Chapter 4 in the context of supervised multitask learning was focused on feature selection. In addition, I found the ability to use task descriptors to make predictions on tasks for which no training data was ever available interesting in the context of *chemogenomics*. Chemogenomics can be viewed as an attempt to complete a large matrix, for which rows correspond to molecules, columns correspond to proteins, and each entry indicates whether or not the molecule binds to the protein [254].

Chemogenomics approaches can be used both to suggest new drugs for a particular therapeutic target, or new targets for a particular drug or drug candidate. In this second scenario, these new targets can be indicative of both new therapeutic indications for the drug – what one calls drug repurposing – or of secondary targets potentially responsible for adverse drug reactions.

In this appendix, I will introduce in Section B.1 state-of-the-art multitask learning approaches for chemogenomics. In Section B.2, I will describe how we evaluated various methods in orphan settings, which occur when the training data contains no binding partner of either the small molecule or the protein, as well as how we investigated the impact of the similarity between queries and the training data. In Section B.3, I will describe how our observations led us to propose NNMT, a nearest-neighbor multitask SVM, which is trained on a limited number of data points. While many existing multitask approache for chemogenomics are limited by their computational complexity, our approach only requires training on a dataset of size similar to those used by single-task methods.

The contents of this appendix are based on joint work with Benoît Playe and Véronique Stoven, published as:

> Benoit Playe, Chloé-Agathe Azencott, and Véronique Stoven. Efficient multi-task chemogenomics for drug specificity prediction. *PLoS ONE*, 13(18):e0204999, 2018.

B.1 Multitask learning for chemogenomics

Adverse drug reactions and drug specificity

Our ability to tailor treatment to patients is tied to our understanding of *adverse drug reactions*, or drug side effects. The incidence of severe ADR among hospitalized patients in the USA is estimated to be 1.9%–2.3%, while the incidence of fatal ADR is 0.13%–0.26% [135]. 462 medicinal products were withdrawn from market due to ADR between 1950 and 2014 [185]; 114 of those were associated with deaths. The ability to identify ADR early on is therefore an important public health concern.

Side effects frequently occur when drugs lack specificity, which means that they bind to proteins other than their intended target [215]. This suggests that one approach to ADR detection is the identification of such secondary drug targets. From a precision medicine point of view, this approach can also lead to determine subgroups of patients who may

not suffer from the corresponding side effects, in particular because they do not express this secondary target.

In the context of drug specificity prediction, a single-task approach consists in classifying proteins according to whether or not they bind a given molecule, based on known targets for this molecule. A multitask approach, by contrast, predicts for any (molecule, protein) pair whether it binds or not, and leverages all known protein-ligand interactions, including those involving neither the molecule nor the protein of interest.

Multitask learning for chemogenomics

Multiple approaches have been developed for chemogenomics prediction, including multitask Support Vector Machines (SVM) [70, 109, 177, 254], kernel ridge linear regression [132, 133, 276], bipartite local models (BLM) [25, 163], and matrix factorization [81, 150, 286]. Most of the proposed approaches, however, are limited by their computational complexity and have only been applied to predict interactions of molecules within proteins belonging to the same family.

Both multitask SVM and kernel ridge linear regression use kernels to compute dot products between either small molecules or proteins (see Section 1.2.6).

Kernels for small molecules A standard approach to build kernels from molecular fingerprints consists in using dot products or Euclidean distances, possibly composed with another suitable function, such as a Gaussian exponential [217].

However, given that these vectorial representations are binary, it is more common to use the *Tanimoto similarity* measure between two binary fingerprints, defined as the ratio of the number of common bits set to one to the total number of bits set to one in the two fingerprints:

$$\begin{aligned} k :&\{0,1\}^m \times \{0,1\}^m \to [0,1] \\ &(\boldsymbol{x}, \boldsymbol{x}') \mapsto \frac{\sum_{p=1}^{m} \left(x_p \text{ AND } x'_p\right)}{\sum_{p=1}^{m} \left(x_p \text{ OR } x'_p\right)}. \end{aligned} \tag{B.1}$$

In the case of count fingerprints, the Tanimoto similarity can be extended with the *MinMax similarity* [202]:

$$\begin{aligned} k :&\mathbb{N}^m \times \mathbb{N}^m \to [0,1] \\ &(\boldsymbol{x}, \boldsymbol{x}') \mapsto \frac{\sum_{p=1}^{m} \min(x_p, x'_p)}{\sum_{p=1}^{m} \max(x_p, x'_p)}. \end{aligned} \tag{B.2}$$

Both these similarity measures are known to be kernels [238], which makes them amenable to use with kernel-based machine learning methods such as SVMs (see Section 1.2.6). Similar ideas appear for example in Mahé et al. [156].

Kernels for proteins For a proteome-wide study, the most appropriate kernels are based on the protein sequences (three-dimensional structures or binding pocket information not being available for all proteins) [127, 212].

Kernels for (molecule, protein) pairs Molecule and protein kernels can be combined by their Kronecker product [66] to create a kernel on (protein, small molecule) pairs, and the multitask learning problem is recast as a single-task one on (protein, small molecule) pairs.

B.2 Orphan and quasi-orphan settings

Developing a state-of-the-art chemogenomics approach that can be applied to the entire druggable proteome requires not only computational scalability, but also the ability to make good predictions in *orphan settings*, that is to say, when some of the proteins for which we want to make predictions have no known ligands. This situation is often encountered in large scale studies, and single-task methods are not applicable. To this end, Pahikkala et al. [189] propose to evaluate multitask algorithms in settings where the queried (protein, molecule) pairs contain proteins and/or molecules that are *not* in the training set.

We investigated the impact of the similarity between the query (protein, small molecule) pair and the training data on the prediction performance, on data extracted from the DrugBank [134] and containing 3 980 molecules, 1 821 proteins, and 9 536 protein-ligand interactions.

Orphan situations

We created 5-fold cross-validation sets of our data in the following way:

- S_1: randomly and balanced in positive and negative pairs;
- S_2 (corresponding to the "orphan ligand" case): (protein, molecule) pairs in one fold only contain molecules that are absent from all other folds; prediction on each test set (each fold) is performed using train sets (the four other folds) in which no the ligands of the test set are absent.
- S_3 (corresponding to the"orphan protein" case): (protein, molecule) pairs in one fold only contain proteins that are absent from all other folds; prediction on each test set is performed using train sets in which no the proteins of the test set are absent.
- S_4 (corresponding to the"double orphan" case): (protein, molecule) pairs in one fold only contain proteins *and* molecules that are both absent from all other folds. Prediction on each test set is performed using train sets in which no the proteins and the ligands of the test set are absent. The folds of S_4 were built by intersecting those of S_2 and S_3 and S_4. Thus, S_4 contains 25 folds and not 5.

Figure B.1 illustrates the difficulty of orphan settings: while the performance in the double orphan setting remains significantly above random, the performance on all orphan situations is clearly degraded compared to a random split of the data.

These results suggest that the performance of multitask SVM is driven by known (protein, small molecule) pairs that are similar to the query pair, in the sense that they share either their protein or their molecule.

Quasi-orphan situations

To evaluate the impact on performance of the similarity between training and test pairs, we re-folded the pairs of S following the "clustered cross-validation" approach [126].

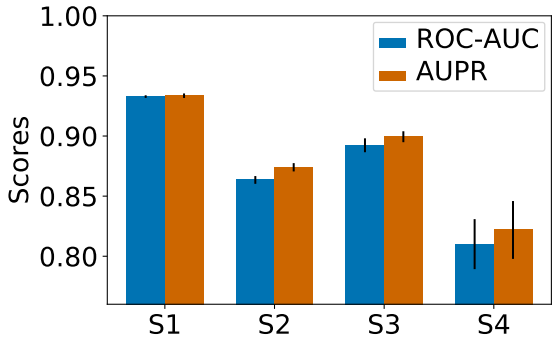


Figure B.1: **Nested 5-fold cross-validated area under the ROC curve (ROC-AUC) and area under the precision-recall curve (AUPR) of a multitask SVM on the $S_1 - S_4$ datasets.**

More precisely, we clustered proteins (resp. ligands) into 5 clusters by hierarchical clustering [113]. We then built four cross-validation datasets, $S'_1 - S'_4$, generated based on folds similarly as $S_1 - S_4$, but with the added constraint that all pairs in a given fold are made of proteins from a single protein cluster and ligands from a single ligand cluster. Therefore, test pairs are more dissimilar from train pairs than in the $S_1 - S_4$ datasets, which makes the problem more difficult.

Our results are illustrated by Figure B.2.

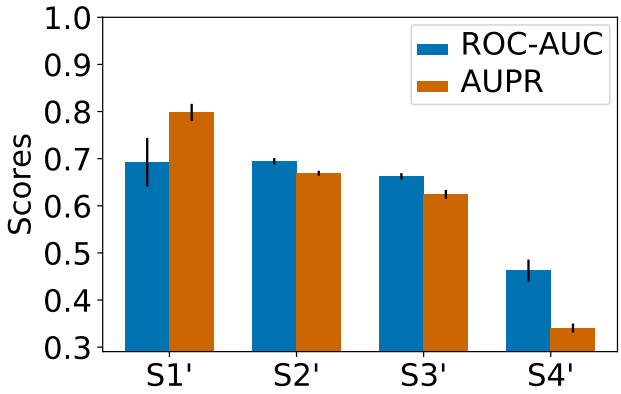


Figure B.2: **Nested 5-fold cross-validated area under the ROC curve (ROC-AUC) and area under the precision-recall curve (AUPR) of a multitask SVM on the $S'_1 - S'_4$ datasets.**

For all the datasets, we observed a strong decrease in prediction scores with respect to those obtained on the corresponding $S_1 - S_4$ datasets. This suggests that good performance on a query pair (p^*, m^*) is driven by the presence in the training set of pairs made *both* of proteins similar to p^* and of molecules similar to m^*, even if the query pair (p^*, m^*) is a double orphan, as in S_4.

These results suggest that pairs in the training set that are very dissimilar to the query pair do not help making more accurate predictions. In other words, although the kernels used in multi-task approaches modulates how information available in one task is shared for training other tasks (the further the tasks are, the less information is shared), using information from distant tasks seems to degrade performance. This insight is interesting since the multitask SVM requires computes the Kronecker kernel on all (protein, molecule) pairs,

which is computationally demanding. Therefore, we proposed to remove distant pairs from the training set to improve computational efficiency, without degrading performance.

B.3 Nearest-neighbors multitask learning with kernels

Our a nearest-neighbor multitask SVM, NNMT, is trained on a limited number of data points: for a query (protein, molecule) pair (p^*, m^*), the training data is composed of

- all *intra-task* (protein, ligand) pairs defined by pairs (p, m) with either $p = p^*$ or $m = m^*$;
- a limited number of *extra-task* (protein, ligand) pairs, defined by pairs (p, m) with $p \neq p^*$ and $m \neq m^*$, chosen based on the similarity of p and m to p^* and m^*, respectively;
- randomly picked negative examples (about ten times more than positive training pairs).

Our results show that NNMT outperforms all its comparison partners [81, 132, 133, 150, 163, 271, 286], independently of the number of known (protein, ligand) interacting pairs involving the same or similar ligands or proteins as the query pair. In addition, it requires much fewer training pairs than the classical multitask SVM approach, and its computational time is therefore close to that of a single-task method. Finally, in the most challenging setting where no similar intra-task nor extra-task training data is available, it performs significantly better than random, in a context where a single-task approach can not make any prediction.

We also observe that adding extra-task pairs to the train set dramatically improves performance. When no close intra-task pairs are available, performance is driven mainly by extra-task training pairs. On the contrary, performance does not improve when the extra-task training pairs are chosen at random, and therefore, are on average further from the test pair. It might even degrade when the number of extra-task pairs becomes large.

Our benchmark study concluded that NNMT is a good default method providing state-of-the-art or better performances, in a wide range of prediction scenarii that can be encountered in real-life studies: proteome-wide prediction, protein family prediction, test (protein, ligand) pairs dissimilar to pairs in the train set, and orphan cases.

All datasets and codes are available at `https://github.com/bplaye/efficient_MultiTask_SVM_for_chemogenomics/`. In addition, we incorporated NNMT to the PyDTI package [150] and also added to that package key cross-validation schemes as well as the DrugBank-based dataset we built for this study. The updated PyDTI package is available at `https://github.com/bplaye/PyDTI/`.

B.4 Conclusion

As noted in Chapter 4, having access to task descriptors makes it possible to make predictions for tasks for which no training data is available. Such orphan settings are common in chemogenomics.

However, multitask learning is not a magic wand, and if the tasks are too different, prediction ability will be degraded. We leveraged this information to reduce the number of

training samples to use in a multitask SVM approach, keeping comparable prediction capacities while drastically reducing computational times.

BIBLIOGRAPHY

[1] Alexandre Abraham, Elvis Dohmatob, Bertrand Thirion, Dimitris Samaras, and Gael Varoquaux. "Extracting brain regions from rest fMRI with total-variation constrained dictionary learning". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2013, pp. 607–615.

[2] Ivan A Adzhubei et al. "A method and server for predicting damaging missense mutations". In: *Nature methods* 7.4 (2010), p. 248.

[3] Simon Anders and Wolfgang Huber. "Differential expression analysis for sequence count data". In: *Genome Biol.* 11.10 (2010), R106.

[4] Samuel J. Aronson and Heidi L. Rehm. "Building the foundation for genomics in precision medicine". In: *Nature* 526.7573 (2015), pp. 336–342.

[5] Nachman Aronszajn. "Theory of reproducing kernels". In: *Transactions of the American mathematical society* 68.3 (1950), pp. 337–404.

[6] Alan Ashworth, Christopher J. Lord, and Jorge S. Reis-Filho. "Genetic Interactions in Cancer Progression and Treatment". In: *Cell* 145.1 (2011), pp. 30–38.

[7] Sezin Kircali Ata, Le Ou-Yang, Yuan Fang, Chee-Keong Kwoh, Min Wu, and Xiao-Li Li. "Integrating node embeddings and biological annotations for genes to predict disease-gene associations". In: *BMC Systems Biology* 12.Suppl 9 (2018).

[8] Susan Athey, Guido W. Imbens, and Stefan Wager. "Approximate residual balancing: debiased inference of average treatment effects in high dimensions". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* (2018).

[9] Susanna Atwell et al. "Genome-wide association study of 107 phenotypes in Arabidopsis thaliana inbred lines". In: *Nature* 465.7298 (2010), pp. 627–631.

[10] Chloé-Agathe Azencott. "Machine learning and genomics: precision medicine vs. patient privacy". In: *Philosophical Transactions of the Royal Society A* 376.2128 (2018).

[11] Chloé-Agathe Azencott, Dominik Grimm, Mahito Sugiyama, Yoshinobu Kawahara, and Karsten M. Borgwardt. "Efficient network-guided multi-locus association mapping with graph cuts". In: *Bioinformatics* 29.13 (2013). Proceedings of the 21st Annual International Conference on Intelligent Systems for Molecular Biology (ISMB 2013), pp. i171–i179.

[12] Chloé-Agathe Azencott, Alexandre Ksikes, S. Joshua Swamidass, Jonathan H. Chen, Liva Ralaivola, and Pierre Baldi. "One- to four-dimensional kernels for virtual screening and the prediction of physical, chemical and biological properties". In: *Journal of Chemical Information and Modeling* 47.3 (2007), pp. 965–974.

[13] Chloé-Agathe Azencott et al. "The inconvenience of data of convenience: computational research beyond post-mortem analyses". In: *Nature methods* 14.10 (2017), p. 937.

[14] Francis Bach. "Structured sparsity-inducing norms through submodular functions". In: *NIPS*. 2010.

[15] Francis Bach. “Learning with Submodular Functions: A Convex Optimization Perspective”. In: *Found Trends Mach Learn* 6.2-3 (2013), pp. 145–373.

[16] Arunkumar Bagavathi and Siddharth Krishnan. “Multi-Net: A Scalable Multiplex Network Embedding Framework”. In: *International Conference on Complex Networks and their Applications*. Springer. 2018, pp. 119–131.

[17] Albert-László Barabási, Natali Gulbahce, and Joseph Loscalzo. “Network medicine: a network-based approach to human disease”. In: *Nature Reviews Genetics* 12.1 (2011), pp. 56–68.

[18] Albert-László Barabási and Zoltán N. Oltvai. “Network biology: understanding the cell’s functional organization”. In: *Nature Reviews Genetics* 5.2 (2004), pp. 101–113.

[19] Sergio E. Baranzini, Nicholas W. Galwey, Joanne Wang, Pouya Khankhanian, et al. “Pathway and network-based analysis of genome-wide association studies in multiple sclerosis”. In: *Hum Mol Genet* 18.11 (2009), pp. 2078–2090.

[20] Rina Foygel Barber and Aaditya Ramdas. “The p-filter: multilayer false discovery rate control for grouped hypotheses”. In: *J. R. Stat. Soc. B* 79.4 (2017), pp. 1247–1268.

[21] Brett K. Beaulieu-Jones, Patryk Orzechowski, and Jason H. Moore. “Mapping Patient Trajectories using Longitudinal Extraction and Deep Learning in the MIMIC-III Critical Care Database”. In: *Biocomputing 2018*. WORLD SCIENTIFIC, 2017, pp. 123–132.

[22] Claude J. P. Bélisle, H. Edwin Romeijn, and Robert L. Smith. “Hit-and-Run Algorithms for Generating Multivariate Distributions”. In: *Mathematics of Operations Research* 18.2 (1993), pp. 255–266.

[23] Victor Bellon, Véronique Stoven, and Chloé-Agathe Azencott. “Multitask feature selection with task descriptors”. In: *Pacific Symposium on Biocomputing*. Vol. 21. 2016, pp. 261–272.

[24] H. C. P. Berbee, C. G. E. Boender, A. H. G. Rinnooy Ran, C. L. Scheffer, R. L. Smith, and J. Telgen. “Hit-and-run algorithms for the identification of nonredundant linear inequalities”. In: *Mathematical Programming* 37.2 (1987), pp. 184–207.

[25] Kevin Bleakley and Yoshihiro Yamanishi. “Supervised prediction of drug–target interactions using bipartite local models”. In: *Bioinformatics* 25.18 (2009), pp. 2397–2403.

[26] Edwin V. Bonilla, Kian M. Chai, and Christopher Williams. “Multi-task Gaussian process prediction”. In: *NIPS* 20 (2007), pp. 153–160.

[27] Yuri Boykov and Vladimir Kolmogorov. “An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 26.9 (2004), pp. 1124–1137.

[28] Evan A. Boyle, Yang I. Li, and Jonathan K. Pritchard. “An Expanded View of Complex Traits: From Polygenic to Omnigenic”. In: *Cell* 169.7 (2017), pp. 1177–1186.

[29] Benjamin Brachi et al. “Linkage and association mapping of Arabidopsis thaliana flowering time in nature”. In: *PLoS genetics* 6.5 (2010), e1000940.

[30] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi. “Federated learning of predictive models from federated Electronic Health Records.” In: *International journal of medical informatics* 112 (2018), pp. 59–67.

[31] Yana Bromberg. “Disease gene prioritization”. In: *PLoS computational biology* 9.4 (2013), e1002902.

[32] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. “Spectral networks and locally connected networks on graphs”. In: *arXiv preprint arXiv:1312.6203* (2013).

[33] Damian Brzyski, Christine B. Peterson, Piotr Sobczyk, Emmanuel J. Candès, Malgorzata Bogdan, and Chiara Sabatti. “Controlling the Rate of GWAS False Discoveries”. In: *Genetics* 205.1 (2017), pp. 61–75.

[34] William S. Bush and Jason H. Moore. “Chapter 11: Genome-Wide Association Studies”. In: *PLoS Comput Biol* 8.12 (2012), e1002822.

[35] Clément Chatelain, Guillermo Durand, Vincent Thuillier, and Franck Augé. “Performance of epistasis detection methods in semi-simulated GWAS”. In: *BMC Bioinformatics* 19.1 (2018).

[36] Andrew Chatr-Aryamontri, Bobby-Joe Breitkreutz, Rose Oughtred, Lorrie Boucher, Sven Heinicke, et al. “The BioGRID interaction database: 2015 update”. In: *Nucleic Acids Res* 43.Database issue (2015), pp. D470–478.

[37] Bernard Chazelle, Ronitt Rubinfeld, and Luca Trevisan. “Approximating the minimum spanning tree weight in sublinear time”. In: *SIAM J. Comput.* 34.6 (2005), pp. 1370–1379.

[38] Gary K. Chen and Yunfei Guo. “Discovering epistasis in large scale genetic association studies by exploiting graphics cards”. In: *Frontiers in Genetics* 4 (2013).

[39] Lin S. Chen et al. “Insights into Colon Cancer Etiology via a Regularized Approach to Gene Set Analysis of GWAS Data”. In: *Am J Hum Genet* 86.6 (2010), pp. 860–871.

[40] Xiaohui Chen et al. “A two-graph guided multi-task Lasso approach for eQTL mapping”. In: *AISTATS*. 2012.

[41] Judy H. Cho et al. “Identification of novel susceptibility loci for inflammatory bowel disease on chromosomes 1p, 3q, and 4q: Evidence for epistasis between 1p and IBD1”. In: *Proceedings of the National Academy of Sciences* 95.13 (1998), pp. 7502–7507.

[42] Han-Yu Chuang, Eunjung Lee, Yu-Tsueng Liu, Doheon Lee, and Trey Ideker. “Network-based classification of breast cancer metastasis”. In: *Mol Syst Biol* 3 (2007), p. 140.

[43] Marc Claesen, Jesse Davis, Frank De Smet, and Bart De Moor. “Assessing binary classifiers using only positive and unlabeled data”. In: *arXiv:1504.06837 [cs, stat]* (2015).

[44] Robert Clarke et al. “The properties of high-dimensional data spaces: implications for exploring gene and protein expression data”. In: *Nature Reviews Cancer* 8.1 (2008), pp. 37–49.

[45] Héctor Climente-Gonzá and Chloé-Agathe Azencott. *martini: GWAS incorporating networks in R*. 2017. url: `https://bioconductor.org/packages/devel/bioc/html/martini.html`.

[46] Héctor Climente-González, Chloé-Agathe Azencott, Samuel Kaski, and Makoto Yamada. “ Block HSIC Lasso: model-free biomarker detection for ultra-high dimensional data ”. In: *Bioinformatics* 35.14 (2019).

[47] Héctor Climente-González and Azencott Chloé-Agathe. "R package for network-guided Genome-Wide Association Studies". 25th Conference on Intelligent Systems for Molecular Biology (poster). Prague, Czech Republic, 2017.

[48] Héctor Climente-González, Lonjou Christine, Lesueur Fabienne, Stoppa-Lyonnet Dominique, Andrieu Nadine, and Azencott Chloé-Agathe. "Judging genetic loci by the company they keep: Comparing network-based methods for biomarker discovery in familial breast cancer". 68th Annual Meeting of the American Society of Human Genetics (poster). San Diego, CA, 2018.

[49] Onofre Combarros, Mario Cortina-Borja, A. David Smith, and Donald J. Lehmann. "Epistasis in sporadic Alzheimer's disease". In: *Neurobiology of Aging* 30.9 (2009), pp. 1333–1349.

[50] Karen N. Conneely and Michael Boehnke. "So many correlated tests, so little time! Rapid adjustment of P values for multiple correlated tests". In: *Am. J. Hum. Genet.* 81.6 (2007), pp. 1158–1168.

[51] Heather J. Cordell. "Detecting gene–gene interactions that underlie human diseases". In: *Nature Reviews Genetics* 10.6 (2009), pp. 392–404.

[52] Heather J. Cordell, Geoffrey C. Wedig, Kevin B. Jacobs, and Robert C. Elston. "Multilocus Linkage Tests Based on Affected Relative Pairs". In: *The American Journal of Human Genetics* 66.4 (2000), pp. 1273–1286.

[53] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. 2nd. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2006.

[54] Lenore Cowen, Trey Ideker, Benjamin J. Raphael, and Roded Sharan. "Network propagation: a universal amplifier of genetic associations". In: *Nature Reviews Genetics* 18.9 (2017), pp. 551–562.

[55] Nancy J. Cox et al. "Loci on chromosomes 2 (NIDDM1) and 15 interact to increase susceptibility to diabetes in Mexican Americans". In: *Nature Genetics* 21.2 (1999), pp. 213–215.

[56] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. "Convolutional neural networks on graphs with fast localized spectral filtering". In: *Advances in neural information processing systems*. 2016, pp. 3844–3852.

[57] David Dernoncourt, Blaise Hanczar, and Jean-Daniel Zucker. "Analysis of feature selection stability on high dimension and small sample data". In: *Computational Statistics & Data Analysis* 71 (2014), pp. 681–693.

[58] Chris Ding and Hanchuan Peng. "Minimum Redundancy Feature Selection from Microarray Gene Expression Data". In: *Journal of Bioinformatics and Computational Biology* 03.02 (2005), pp. 185–205.

[59] Alexandre Drouin, Gaël Letarte, Frédéric Raymond, Mario Marchand, Jacques Corbeil, and François Laviolette. "Interpretable genotype-to-phenotype classifiers with performance guarantees". In: *Scientific Reports* 9.1 (2019), p. 4071.

[60] Joseph L. Durant, Burton A. Leland, Douglas R. Henry, and James G. Nourse. "Reoptimization of MDL Keys for Use in Drug Discovery". In: *Journal of Chemical Information and Computer Sciences* 42.6 (2002), pp. 1273–1280.

[61] Diane Duroux et al. "Improving efficiency in epistasis detection with a gene-based analysis using functional filters". 28th International Genetic Epidemiology Society meeting (poster). Houston, TX, 2019.

[62] Simone Ecker, Vera Pancaldi, Daniel Rico, and Alfonso Valencia. “Higher gene expression variability in the more aggressive subtype of chronic lymphocytic leukemia”. In: *Genome Med* 7.1 (2015), p. 8.

[63] Federica Eduati et al. “Opportunities and limitations in the prediction of population responses to toxic compounds assessed through a collaborative competition”. In: *Nature Biotechnology* 33.9 (2015), pp. 933–940.

[64] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. “Least angle regression”. In: *The Annals of statistics* 32.2 (2004), pp. 407–499.

[65] Liat Ein-Dor, Itai Kela, Gad Getz, David Givol, and Eytan Domany. “Outcome signature genes in breast cancer: is there a unique set?” In: *Bioinformatics* 21.2 (2005), pp. 171–178.

[66] Dumitru Erhan, Pierre-Jean L’Heureux, Shi Yi Yue, and Yoshua Bengio. “Collaborative filtering on a family of biological targets”. In: *Journal of chemical information and modeling* 46.2 (2006), pp. 626–635.

[67] Sinan Erten, Gurkan Bebek, and Mehmet Koyutürk. “Vavien: an algorithm for prioritizing candidate disease genes based on topological similarity of proteins in interaction networks”. In: *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology* 18.11 (2011), pp. 1561–1574.

[68] Theodoros Evgeniou, Charles A Micchelli, and Massimiliano Pontil. “Learning multiple tasks with kernel methods”. In: *J Mach Learn Res* (2005), pp. 615–637.

[69] Paolo Fardin, Annalisa Barla, Sofia Mosci, Lorenzo Rosasco, Alessandro Verri, and Luigi Varesio. “The l1-l2 regularization framework unmasks the hypoxia signature hidden in the transcriptome of a set of heterogeneous neuroblastoma cell lines”. In: *BMC Genomics* 10 (2009), p. 474.

[70] Jean-Loup Faulon, Milind Misra, Shawn Martin, Ken Sale, and Rajat Sapra. “Genome scale enzyme–metabolite and drug–target interaction predictions using the signature molecular descriptor”. In: *Bioinformatics* 24.2 (2008), pp. 225–233.

[71] Hongliang Fei and Jun Huan. “Structured feature selection and task relationship inference for multi-task learning”. In: *Knowledge and Information Systems* 35.2 (2013), pp. 345–364.

[72] Alain Fischer and Antonio Rausell. “Primary immunodeficiencies suggest redundancy within the human immune system”. In: *Science immunology* 1.6 (2016), eaah5861.

[73] Ronald A. Fisher. “XV.—The Correlation between Relatives on the Supposition of Mendelian Inheritance.” In: *Transactions of the Royal Society of Edinburgh* 52.02 (1919), pp. 399–433.

[74] Satoru Fujishige. *Submodular Functions and Optimization*. 2005.

[75] Laura I. Furlong. “Human diseases through the lens of network biology”. In: *Trends in Genetics* 29.3 (2013), pp. 150–159.

[76] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan. “A fast parametric maximum flow algorithm and applications”. In: *SIAM Journal on Computing* 18.1 (1989), pp. 30–55.

[77] Elbert Geuze, Eric Vermetten, and J. Douglas Bremner. “MR-based in vivo hippocampal volumetrics: 1. Review of methodologies currently employed”. In: *Molecular Psychiatry* 10.2 (2005), pp. 147–159.

[78] Jesse Gillis and Paul Pavlidis. ""Guilt by association" is the exception rather than the rule in gene networks". In: *PLoS computational biology* 8.3 (2012), e1002444.

[79] Vladimir Gligorijević, Noël Malod-Dognin, and Nataša Pržulj. "Integrative methods for analyzing big data in precision medicine". In: *Proteomics* 16.5 (2016), pp. 741–758.

[80] Kwang-Il Goh, Michael E Cusick, David Valle, Barton Childs, Marc Vidal, and Albert-László Barabási. "The human disease network". In: *Proceedings of the National Academy of Sciences* 104.21 (2007), pp. 8685–8690.

[81] Mehmet Gönen. "Predicting drug–target interactions from chemical and genomic kernels using Bayesian matrix factorization". In: *Bioinformatics* 28.18 (2012), pp. 2304–2310.

[82] Benjamin Goudey et al. "GWIS - model-free, fast and exhaustive search for epistatic interactions in case-control GWAS". In: *BMC Genomics* 14.Suppl 3 (2013), S10.

[83] D. M. Greig, B. T. Porteous, and A. H. Seheult. "Exact maximum a posteriori estimation for binary images". In: *J. R. Stat. Soc.* 51.2 (1989).

[84] A. Gretton, O. Bousquet, Alexander Smola, and Bernhard Schölkopf. "Measuring statistical dependence with Hilbert-Schmidt norms". In: *ALT*. 2005.

[85] Arthur Gretton, Kenji Fukumizu, Choon H Teo, Le Song, Bernhard Schölkopf, and Alex J Smola. "A Kernel Statistical Test of Independence". In: *Advances in Neural Information Processing Systems 20*. Ed. by J C Platt, D Koller, Y Singer, and S T Roweis. Curran Associates, Inc., 2008, pp. 585–592.

[86] Arthur Gretton et al. "Kernel Constrained Covariance for Dependence Measurement". In: *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*. 2005, pp. 1–8.

[87] Dominik Grimm et al. "The evaluation of tools used to predict the impact of missense variants is hindered by two types of circularity". In: *Human Mutation* 36.5 (2015), pp. 513–523.

[88] Aditya Grover and Jure Leskovec. "node2vec: Scalable Feature Learning for Networks". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*. San Francisco, California, USA: ACM Press, 2016, pp. 855–864.

[89] Dimitri Guala and Erik L. L. Sonnhammer. "A large-scale benchmark of gene prioritization methods". In: *Scientific reports* 7 (2017), p. 46598.

[90] Daniel F Gudbjartsson et al. "Many sequence variants affecting diversity of adult human height". In: *Nature Genetics* 40.5 (2008), pp. 609–615.

[91] Anja C Gumpinger, Damian Roqueiro, Dominik G Grimm, and Karsten M Borgwardt. "Methods and Tools in Genome-wide Association Studies". In: *Computational Cell Biology*. Springer, 2018, pp. 93–136.

[92] Isabelle Guyon and André Elisseeff. "An introduction to variable and feature selection". In: *J. Mach. Learn. Res.* 3 (2003), pp. 1157–1182.

[93] Will Hamilton, Zhitao Ying, and Jure Leskovec. "Inductive representation learning on large graphs". In: *Advances in Neural Information Processing Systems*. 2017, pp. 1024–1034.

[94] William L. Hamilton, Rex Ying, and Jure Leskovec. “Representation Learning on Graphs: Methods and Applications”. In: *arXiv:1709.05584 [cs]* (2017).

[95] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2001.

[96] Trevor Hastie, Saharon Rosset, Robert Tibshirani, and Ji Zhu. “The entire regularization path for the support vector machine”. In: *Journal of Machine Learning Research* 5 (2005), pp. 1391–1415.

[97] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.

[98] Anne Claire Haury, Fantine Mordelet, Paola Vera-Licona, and Jean Philippe Vert. “TIGRESS: Trustful Inference of Gene REgulation using Stability Selection”. In: *BMC Systems Biology* 6 (2012).

[99] Anne-Claire Haury, Pierre Gestraud, and Jean-Philippe Vert. “The Influence of Feature Selection Methods on Accuracy, Stability and Interpretability of Molecular Signatures”. In: *PLoS ONE* 6.12 (2011), e28210.

[100] David Heckerman, Carl Kadie, and Jennifer Listgarten. “Leveraging information across HLA alleles/supertypes improves epitope prediction”. In: *J Comput Biol* 14.6 (2007), pp. 736–746.

[101] Ottar Hellevik. “Linear versus logistic regression when the dependent variable is a dichotomy”. In: *Quality & Quantity* 43.1 (2009), pp. 59–74.

[102] Gibran Hemani, Athanasios Theocharidis, Wenhua Wei, and Chris Haley. “EpiGPU: exhaustive pairwise epistasis scans parallelized on consumer level graphics cards”. In: *Bioinformatics* 27.11 (2011), pp. 1462–1465.

[103] Joshua W. Ho, Moriztio Stefani, Cristobaldo G. dos Remedios, and Michael A. Charleston. “Differential variability analysis of gene expression and its application to human diseases”. In: *Bioinformatics* 24.13 (2008), pp. i390–398.

[104] Ludwig A. Hothorn, Ondrej Libiger, and Daniel Gerhard. “Model-specific tests on variance heterogeneity for detection of potentially interacting genetic loci”. In: *BMC Genet.* 13 (2012), p. 59.

[105] Xiaohan Hu et al. “SHEsisEpi, a GPU-enhanced genome-wide SNP-SNP interaction scanning algorithm, efficiently reveals the risk genetic epistasis in bipolar disorder”. In: *Cell Research* 20.7 (2010), pp. 854–857.

[106] Junzhou Huang, Tong Zhang, and Dimitris Metaxas. “Learning with structured sparsity”. In: *J. Mach. Learn. Res.* 12 (2011), pp. 3371–3412.

[107] Trey Ideker, Owen Ozier, Benno Schwikowski, and Andrew F. Siegel. “Discovering regulatory and signalling circuits in molecular interaction networks”. In: *Bioinformatics* 18.suppl 1 (2002), S233–S240.

[108] Laurent Jacob, Guillaume Obozinski, and Jean-Philippe Vert. “Group lasso with overlap and graph lasso”. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 433–440.

[109] Laurent Jacob and Jean-Philippe Vert. “Protein-ligand interaction prediction: an improved chemogenomics approach”. In: *Bioinformatics* 24.19 (2008), pp. 2149–2156.

[110] Shantanu Jain, Martha White, and Predrag Radivojac. "Recovering True Classifier Performance in Positive-Unlabeled Learning". In: *arXiv:1702.00518 [cs, stat]* (2017).

[111] Vuk Janjić and Nataša Pržulj. "Biological function through network topology: a survey of the human diseasome". In: *Briefings in functional genomics* 11.6 (2012), pp. 522–532.

[112] Peilin Jia et al. "Network-Assisted Investigation of Combined Causal Signals from Genome-Wide Association Studies in Schizophrenia". In: *PLoS Comput Biol* 8.7 (2012), e1002587.

[113] Stephen C. Johnson. "Hierarchical clustering schemes". In: *Psychometrika* 32.3 (1967), pp. 241–254.

[114] Iain M. Johnstone and D. Michael Titterington. "Statistical challenges of high-dimensional data." In: *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences* 367.1906 (2009), pp. 4237–53.

[115] Tony Kam-Thong, Benno Pütz, Nazanin Karbalai, Bertram Müller-Myhsok, and Karsten Borgwardt. "Epistasis detection on quantitative phenotypes by exhaustive enumeration using GPUs". In: *Bioinformatics* 27.13 (2011), pp. i214–i221.

[116] Tony Kam-Thong et al. "EPIBLASTER-fast exhaustive two-locus epistasis detection strategy using graphical processing units". In: *European Journal of Human Genetics* 19.4 (2011), pp. 465–471.

[117] Tony Kam-Thong et al. "GLIDE: GPU-based linear regression for the detection of epistasis". In: *Human Heredity* 73 (2012), pp. 220–236.

[118] Matthew A. Kayala, Chloé-Agathe Azencott, Jonathan H. Chen, and Pierre Baldi. "Learning to predict chemical reactions". In: *Journal of Chemical Information and Modeling* 51.9 (2011), pp. 2209–2222.

[119] Seyoung Kim, Kyung-Ah Sohn, and Eric Xing. "A multivariate regression approach to association analysis of a quantitative trait network." In: *Bioinformatics (Oxford, England)* 25.12 (2009), p. 12.

[120] Gad Kimmel and Ron Shamir. "A Block-Free Hidden Markov Model for Genotypes and Its Application to Disease Association". In: *Journal of Computational Biology* 12.10 (2005), pp. 1243–1260.

[121] Thomas N. Kipf and Max Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *arXiv:1609.02907 [cs, stat]* (2016).

[122] Martin Kircher, Daniela M Witten, Preti Jain, Brian J O'Roak, Gregory M Cooper, and Jay Shendure. "A general framework for estimating the relative pathogenicity of human genetic variants". In: *Nature genetics* 46.3 (2014), p. 310.

[123] Sebastian Köhler, Sebastian Bauer, Denise Horn, and Peter N. Robinson. "Walking the Interactome for Prioritization of Candidate Disease Genes". In: *The American Journal of Human Genetics* 82.4 (2008), pp. 949–958.

[124] Martin A. Kohli et al. "The Neuronal Transporter Gene SLC6A15 Confers Risk to Major Depression". In: *Neuron* 70.2 (2011), pp. 252–265.

[125] Vladimir Kolmogorov and Ramin Zabin. "What energy functions can be minimized via graph cuts?" In: *IEEE Trans. Pattern Anal. Mach. Intell.* 26.2 (2004), pp. 147–159.

[126] Christian Kramer and Peter Gedeck. "Leave-cluster-out cross-validation is appropriate for scoring functions derived from diverse protein data sets". In: *Journal of chemical information and modeling* 50.11 (2010), pp. 1961–1969.

[127] Rui Kuang et al. “Profile-based string kernels for remote homology detection and motif extraction”. In: *Journal of bioinformatics and computational biology* 3.03 (2005), pp. 527–550.

[128] Ludmila I. Kuncheva. “A Stability Index for Feature Selection”. In: *Proceedings of the 25th Conference on Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications*. ACTA Press, 2007, pp. 390–395.

[129] Ludmila I. Kuncheva, Christopher J. Smith, Yasir Syed, Christopher O. Phillips, and Keir E. Lewis. “Evaluation of feature ranking ensembles for high-dimensional biomedical data”. In: *ICDM Workshops*. 2012, pp. 49–56.

[130] I. Kuperstein et al. “Atlas of Cancer Signalling Network: a systems biology resource for integrative analysis of cancer data with Google Maps”. In: *Oncogenesis* 4.7 (2015), e160.

[131] Lydia Coulter Kwee, Dawei Liu, Xihong Lin, Debashis Ghosh, and Michael P. Epstein. “A Powerful and Flexible Multilocus Association Test for Quantitative Traits”. In: *The American Journal of Human Genetics* 82.2 (2008), pp. 386–397.

[132] Twan van Laarhoven and Elena Marchiori. “Predicting drug-target interactions for new drug compounds using a weighted nearest neighbor profile”. In: *PloS one* 8.6 (2013), e66952.

[133] Twan van Laarhoven, Sander B Nabuurs, and Elena Marchiori. “Gaussian interaction profile kernels for predicting drug–target interaction”. In: *Bioinformatics* 27.21 (2011), pp. 3036–3043.

[134] Vivian Law et al. “DrugBank 4.0: shedding new light on drug metabolism”. In: *Nucleic acids research* 42.D1 (2013), pp. D1091–D1097.

[135] Jason Lazarou, Bruce H Pomeranz, and Paul N Corey. “Incidence of adverse drug reactions in hospitalized patients: a meta-analysis of prospective studies”. In: *Jama* 279.15 (1998), pp. 1200–1205.

[136] Marine Le Morvan and Jean-Philippe Vert. “WHInter: A Working set algorithm for High-dimensional sparse second order Interaction models”. In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. Stockholmsmässan, Stockholm Sweden: PMLR, 2018, pp. 3635–3644.

[137] J. D. Lee, D. L. Sun, Y. Sun, and J. E. Taylor. “Exact post-selection inference, with application to the lasso”. In: *The Annals of Statistics* 44.3 (2016), pp. 907–927.

[138] Seunggeung Lee, Gonçalo R. Abecasis, Michael Boehnke, and Xihong Lin. “Rare-Variant Association Analysis: Study Designs and Statistical Tests”. In: *Am J Hum Genet* 95.1 (2014), pp. 5–23.

[139] Seunghak Lee and Eric P Xing. “Leveraging input and output structures for joint mapping of epistatic and marginal eQTLs”. In: *Bioinformatics* 28.12 (2012), pp. i137–i146.

[140] Caiyan Li and Hongzhe Li. “Network-constrained regularization and variable selection for analysis of genomic data”. In: *Bioinformatics* 24.9 (2008), pp. 1175–1182.

[141] Caiyan Li and Hongzhe Li. “Variable selection and regression analysis for graph-structured covariates with an application to genomics”. In: *Ann. Appl. Stat.* 4.3 (2010), pp. 1498–1516.

[142] Chumei Li. “Personalized medicine – the promised land: are we there yet?” In: *Clinical Genetics* 79.5 (2011), pp. 403–412.

[143] Jun Li and Alicia T. Lamere. “DiPhiSeq: Robust comparison of expression levels on RNA-Seq data with large sample sizes”. In: *Bioinformatics* (2018).

[144] Yongjin Li and Jinyan Li. “Disease gene identification by random walk on multigraphs merging heterogeneous genomic and phenotype data”. In: *BMC Genomics* 13.7 (2012), S27.

[145] Dawei Liu, Debashis Ghosh, and Xihong Lin. “Estimation and testing for the effect of a genetic pathway on a disease outcome using logistic kernel machine regression via logistic mixed models”. In: *BMC bioinformatics* 9.1 (2008), p. 292.

[146] Dawei Liu, Xihong Lin, and Debashis Ghosh. “Semiparametric regression of multidimensional genetic pathway data: Least-squares kernel machines and linear mixed models”. In: *Biometrics* 63.4 (2007), pp. 1079–1088.

[147] Jimmy Z. Liu, Allan F. Mcrae, Dale R. Nyholt, Sarah E. Medland, et al. “A Versatile Gene-Based Test for Genome-wide Association Studies”. In: *Am J Hum Genet* 87.1 (2010), pp. 139–145.

[148] Jin Liu, Kai Wang, Shuangge Ma, and Jian Huang. “Accounting for linkage disequilibrium in genome-wide association studies: A penalized regression method”. In: *Statistics and Its Interface* 6.1 (2013), pp. 99–115.

[149] Weiyi Liu, Pin-Yu Chen, Sailung Yeung, Toyotaro Suzumura, and Lingli Chen. “Principled Multilayer Network Embedding”. In: *arXiv:1709.03551 [physics]* (2017).

[150] Yong Liu, Min Wu, Chunyan Miao, Peilin Zhao, and Xiao-Li Li. “Neighborhood regularized logistic matrix factorization for drug-target interaction prediction”. In: *PLoS computational biology* 12.2 (2016), e1004760.

[151] Luke R. Lloyd-Jones, Matthew R. Robinson, Jian Yang, and Peter M. Visscher. “Transformation of Summary Statistics from Linear Mixed Model Association on All-or-None Traits to Odds Ratio”. In: *Genetics* 208.4 (2018), pp. 1397–1408.

[152] Joshua R Loftus and Jonathan E Taylor. “Selective inference in regression models with groups of variables”. In: *arXiv preprint arXiv:1511.01478* (2015).

[153] László Lovász. “Random Walks on Graphs: A Survey”. In: *Combinatorics* 2 (1993), pp. 1–46.

[154] Michael I. Love, Wolfgang Huber, and Simon Anders. “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2”. In: *Genome Biol.* 15.12 (2014), p. 550.

[155] Jared K. Lunceford and Marie Davidian. “Stratification and weighting via the propensity score in estimation of causal treatment effects: A comparative study”. In: *Statistics in Medicine* 23.19 (2004), pp. 2937–2960.

[156] Pierre Mahé, Nobuhisa Ueda, Tatsuya Akutsu, Jean-Luc Perret, and Jean-Philippe Vert. “Graph kernels for molecular structure-activity relationship analysis with support vector machines”. In: *Journal of chemical information and modeling* 45.4 (2005), pp. 939–951.

[157] Noël Malod-Dognin, Julia Petschnigg, and Nataša Pržulj. “Precision medicine – A promising, yet challenging road lies ahead”. In: *Current Opinion in Systems Biology* 7 (2018), pp. 1–7.

[158] Teri A. Manolio et al. “Finding the missing heritability of complex diseases”. In: *Nature* 461.7265 (2009), pp. 747–753.

[159] Jessica C. Mar et al. “Variance of gene expression identifies altered network constraints in neurological disease”. In: *PLoS Genet.* 7.8 (2011), e1002207.

[160] Daniel Marbach, David Lamparter, Gerald Quon, Manolis Kellis, Zoltán Kutalik, and Sven Bergmann. “Tissue-specific regulatory circuits reveal variable modular perturbations across complex diseases”. In: *Nature methods* 13.4 (2016), p. 366.

[161] Mathurin Massias, Alexandre Gramfort, and Joseph Salmon. “Celer: a Fast Solver for the Lasso with Dual Extrapolation”. In: *ICML 2018 - 35th International Conference on Machine Learning*. Vol. 80. PMLR. Stockholm, Sweden, July 2018, pp. 3321–3330.

[162] Matthew T. Maurano et al. “Systematic localization of common disease-associated variation in regulatory DNA”. In: *Science* 337.6099 (2012), pp. 1190–1195.

[163] Jian-Ping Mei, Chee-Keong Kwoh, Peng Yang, Xiao-Li Li, and Jie Zheng. “Drug–target interaction prediction by learning from local information and neighbors”. In: *Bioinformatics* 29.2 (2013), pp. 238–245.

[164] Nicolai Meinshausen and Peter Bühlmann. “Stability selection”. In: *J. R. Stat. Soc.* 72.4 (2010), pp. 417–473.

[165] Russell Merris. “Laplacian matrices of graphs: a survey”. In: *Linear Algebra Appl* 197 (1994), pp. 143–176.

[166] Charles A. Micchelli, Jean M. Morales, and Massimiliano Pontil. “Regularizers for structured sparsity”. In: *Adv. Comput. Math* 38.3 (2013), pp. 455–489.

[167] Vincent Michel, Alexandre Gramfort, Gaël Varoquaux, Evelyn Eger, and Bertrand Thirion. “Total variation regularization for fMRI-based prediction of behavior”. In: *IEEE transactions on medical imaging* 30.7 (2011), pp. 1328–1340.

[168] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “Efficient Estimation of Word Representations in Vector Space”. In: *arXiv:1301.3781 [cs]* (2013).

[169] Koyel Mitra, Anne-Ruxandra Carvunis, Sanath Kumar Ramesh, and Trey Ideker. “Integrative approaches for finding modular structure in biological networks”. In: *Nat Rev Genet* 14.10 (2013), pp. 719–732.

[170] Jason H. Moore. “A global view of epistasis”. In: *Nature Genetics* 37.1 (2005), pp. 13–14.

[171] Jason H. Moore, Folkert W. Asselbergs, and Scott M. Williams. “Bioinformatics challenges for genome-wide association studies”. In: *Bioinformatics* (2010), btp713.

[172] F. Mordelet and J.-P. Vert. “A bagging SVM to learn from positive and unlabeled examples”. In: *Pattern Recognition Letters* 37 (2014), pp. 201–209.

[173] Fantine Mordelet and Jean-Philippe Vert. “ProDiGe: Prioritization Of Disease Genes with multitask machine learning from positive and unlabeled examples”. In: *BMC Bioinformatics* 12.1 (2011), p. 389.

[174] James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. “Explainable Prediction of Medical Codes from Clinical Text”. In: 2018, pp. 1101–1111.

[175] Preethy Sasidharan Nair and Mauno Vihinen. “VariBench: a benchmark database for variations”. In: *Human mutation* 34.1 (2013), pp. 42–49.

[176] Shinichi Nakagawa. “A farewell to Bonferroni: the problems of low statistical power and publication bias”. In: *Behavioral Ecology* 15.6 (2004), pp. 1044–1045.

[177] Francesco Napolitano et al. “Drug repositioning: a machine-learning approach through data integration.” In: *J. Cheminformatics* 5 (2013), p. 30.

[178] Eugene Ndiaye, Olivier Fercoq, Alexandre Gramfort, and Joseph Salmon. “Gap safe screening rules for sparsity enforcing penalties”. In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 4671–4703.

[179] Yurii Nesterov. *Introductory Lectures on Convex Optimization*. Vol. 87. Springer, 2004.

[180] Pauline C. Ng and Steven Henikoff. “SIFT: Predicting amino acid changes that affect protein function”. In: *Nucleic acids research* 31.13 (2003), pp. 3812–3814.

[181] Clément Niel, Christine Sinoquet, Christian Dina, and Ghislain Rocheleau. “A survey about methods dedicated to epistasis detection”. In: *Bioinformatics and Computational Biology* (2015), p. 285.

[182] Roland Nilsson, José M. Peña, Johan Björkegren, and Jesper Tegnér. “Consistent Feature Selection for Pattern Recognition in Polynomial Time”. In: *Journal of Machine Learning Research* 8.Mar (2007), pp. 589–612.

[183] Sarah Nogueira and Gavin Brown. “Measuring the stability of feature selection”. In: *Machine Learning and Knowledge Discovery in Databases*. Lecture Notes in Computer Science 9852. Springer International Publishing, 2016, pp. 442–457.

[184] Guillaume Obozinski, Ben Taskar, and Michael I. Jordan. *Multi-task feature selection*. Tech. rep. UC Berkeley, 2006.

[185] Igho J. Onakpoya, Carl J. Heneghan, and Jeffrey K. Aronson. “Post-marketing withdrawal of 462 medicinal products because of adverse drug reactions: a systematic review of the world literature”. In: *BMC medicine* 14.1 (2016), p. 10.

[186] James B. Orlin. “A faster strongly polynomial time algorithm for submodular function minimization”. In: *Math. Program.* 118.2 (2009), pp. 237–251.

[187] Martin Oti, Berend Snel, Martijn A Huynen, and Han G Brunner. “Predicting disease genes using protein–protein interactions”. In: *Journal of medical genetics* 43.8 (2006), pp. 691–698.

[188] Karim Oualkacha et al. “Adjusted sequence kernel association test for rare variants controlling for cryptic and family relatedness”. In: *Genet. Epidemiol.* 37.4 (2013), pp. 366–376.

[189] Tapio Pahikkala et al. “Toward more realistic drug–target interaction predictions”. In: *Briefings in bioinformatics* (2014), bbu010.

[190] Gina M. Peloso and Kathryn L. Lunetta. “Choice of population structure informative principal components for adjustment in a case-control study”. In: *BMC genetics* 12.1 (2011), p. 64.

[191] H. Peng, F. Long, and C. Ding. “Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (2005), pp. 1226–1237.

[192] Jiska S. Peper, Rachel M. Brouwer, Dorret I. Boomsma, René S. Kahn, and Hilleke E. Hulshoff Pol. “Genetic influences on human brain structure: A review of brain imaging studies in twins”. In: *Human Brain Mapping* 28.6 (2007), pp. 464–473.

[193] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. “DeepWalk: Online Learning of Social Representations”. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’14* (2014), pp. 701–710.

[194] Bjoern Peters, Huynh-Hoa Bui, Sune Frankild, Morten Nielson, et al. “A community resource benchmarking predictions of peptide binding to MHC-I molecules”. In: *PLoS Comput Biol* 2.6 (2006), e65.

[195] Benoit Playe. “Machine learning approaches for drug virtual screening”. Thesis. PSL Research University, 2019.

[196] Benoît Playe, Chloé-Agathe Azencott, and Véronique Stoven. “Efficient multi-task chemogenomics for drug specificity prediction”. In: *PLoS ONE* 13.18 (2018), e0204999.

[197] Alkes L. Price, Noah A. Zaitlen, David Reich, and Nick Patterson. “New approaches to population stratification in genome-wide association studies”. In: *Nature Reviews Genetics* 11.7 (2010), p. 459.

[198] Christophe Le Priol, Chloé-Agathe Azencott, and Xavier Gidrol. “Large-scale RNA-seq datasets enable the detection of genes with a differential expression dispersion in cancer”. 20th Open Days in Biology, Computer Science and Mathematics (poster). Nantes, France, 2019.

[199] Florian Privé, Hugues Aschard, Andrey Ziyatdinov, and Michael G.B. Blum. “Efficient analysis of large-scale genome-wide data with two R packages: bigstatsr and bigsnpr”. In: *Bioinformatics* (2018).

[200] Shaun Purcell et al. “PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses”. In: *The American Journal of Human Genetics* 81.3 (2007), pp. 559–575.

[201] Long Qu, Tobias Guennel, and Scott L. Marshall. “Linear score tests for variance components in linear mixed models and applications to genetic association studies”. In: *Biometrics* 69.4 (2013), pp. 883–892.

[202] Liva Ralaivola, Sanjay J. Swamidass, Hiroto Saigo, and Pierre Baldi. “Graph kernels for chemical informatics”. In: *Neural Networks* 18.8 (2005), pp. 1093–1110.

[203] Di Ran and Z. John Daye. “Gene expression variability and the analysis of large-scale RNA-seq studies with the MDSeq”. In: *Nucleic Acids Res.* 45.13 (2017), e127.

[204] Pasi Rastas, Mikko Koivisto, Heikki Mannila, and Esko Ukkonen. “A Hidden Markov Technique for Haplotype Reconstruction”. In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2005, pp. 140–151.

[205] Stephen Reid, Jonathan Taylor, and Robert Tibshirani. “A General Framework for Estimation and Inference From Clusters of Features”. In: *Journal of the American Statistical Association* 113.521 (2017), pp. 280–293.

[206] Marylyn D. Ritchie and Kristel Van Steen. “The search for gene-gene interactions in genome-wide association studies: challenges in abundance of methods, practical considerations, and biological interpretation”. In: *Annals of translational medicine* 6.8 (2018).

[207] Marylyn D. Ritchie et al. “Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer”. In: *The American Journal of Human Genetics* 69.1 (2001), pp. 138–147.

[208] Mark D. Robinson, Davis J. McCarthy, and Gordon K. Smyth. "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data". In: *Bioinformatics* 26.1 (2010), pp. 139–140.

[209] David Rogers and Mathew Hahn. "Extended-connectivity fingerprints". In: *Journal of chemical information and modeling* 50.5 (2010), pp. 742–754.

[210] Donald B. Rubin. "Estimating causal effects of treatments in randomized and nonrandomized studies." In: *Journal of Educational Psychology* 66.5 (1974), pp. 688–701.

[211] Ashis Saha et al. "Co-expression networks reveal the tissue-specific regulation of transcription and splicing". In: *Genome research* 27.11 (2017), pp. 1843–1858.

[212] Hiroto Saigo, Jean-Philippe Vert, Nobuhisa Ueda, and Tatsuya Akutsu. "Protein homology detection using string alignment kernels". In: *Bioinformatics* 20.11 (2004), pp. 1682–1689.

[213] Kuljeet Singh Sandhu, Guoliang Li, Huay Mei Poh, Yu Ling Kelly Quek, et al. "Large-Scale Functional Organization of Long-Range Chromatin Interaction Networks". In: *Cell Rep* 2.5 (2012), pp. 1207–1219.

[214] Paul Scheet and Matthew Stephens. "A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase." In: *American journal of human genetics* 78.4 (2006), pp. 629–44.

[215] Josef Scheiber et al. "Gaining insight into off-target mediated effects of drug candidates with a comprehensive systems chemical biology analysis". In: *Journal of chemical information and modeling* 49.2 (2009), pp. 308–317.

[216] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. "Modeling Relational Data with Graph Convolutional Networks". In: *arXiv:1703.06103 [cs, stat]* (2017).

[217] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels*. Cambridge, MA: MIT Press, 2002.

[218] Nicholas J. Schork. "Personalized medicine: Time for one-person trials". In: *Nature News* 520.7549 (2015), p. 609.

[219] Thierry Schüpbach, Ioannis Xenarios, Sven Bergmann, and Karen Kapur. "FastEpistasis: a high performance computing solution for quantitative trait epistasis". In: *Bioinformatics* 26.11 (2010), pp. 1468–1469.

[220] Rajen D. Shah and Richard J. Samworth. "Variable selection with error control: another look at stability selection". In: *Journal of the Royal Statistical Society* 75.1 (2013), pp. 55–80.

[221] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. "Pitfalls of Graph Neural Network Evaluation". In: *arXiv:1811.05868 [cs, stat]* (2018).

[222] Nasrullah Sheikh, Zekarias Kefato, and Alberto Montresor. "gat2vec: representation learning for attributed graphs". In: *Computing* (2018), pp. 1–23.

[223] Solveig K. Sieberts et al. "Crowdsourced assessment of common genetic contribution to predicting anti-TNF treatment response in rheumatoid arthritis". In: *Nature Communications* 7 (2016), p. 12460.

[224] Karri Silventoinen et al. “Heritability of adult body height: a comparative study of twin cohorts in eight countries”. In: *Twin Research and Human Genetics* 6.5 (2003), pp. 399–408.

[225] Matt Silver, Giovanni Montana, and Alzheimer’s Disease Neuroimaging Initiative. “Fast identification of biological pathways associated with a quantitative trait using group lasso with overlaps”. In: *Stat Appl Genet Mol Biol* 11.1 (2012), p. 7.

[226] Lotfi Slim, Clément Chatelain, Chloé-Agathe Azencott, and Jean-Philippe Vert. *kernelPSI: Post-Selection Inference for Nonlinear Variable Selection*. 2010. url: `https://cran.r-project.org/package=kernelPSI`.

[227] Lotfi Slim, Clément Chatelain, Chloé-Agathe Azencott, and Jean-Philippe Vert. *epiGWAS: Robust Methods for Epistasis Detection*. 2018. url: `https://cran.r-project.org/package=epiGWAS`.

[228] Lotfi Slim, Clément Chatelain, Chloé-Agathe Azencott, and Jean-Philippe Vert. “Novel methods for epistasis detection in genome-wide association studies”. In: *bioRxiv:10.1101/442749* (2018).

[229] Lotfi Slim, Clément Chatelain, Chloé-Agathe Azencott, and Jean-Philippe Vert. “kernelPSI: a post-selection inference framework for nonlinear variable selection”. In: *Proceedings of the Thirty-Sixth International Conference on Machine Learning (ICML)*. Vol. 97. 2019, pp. 5857–5865.

[230] Alexander J. Smola and Risi Kondor. “Kernels and regularization on graphs”. In: *Learning Theory and Kernel Machines*. Vol. 2777. 2003, pp. 144–158.

[231] Artem Sokolov, Daniel E. Carlin, Evan O. Paull, Robert Baertsch, and Joshua M. Stuart. “Pathway-Based Genomics Prediction using Generalized Elastic Net”. In: *PLoS Comput Biol* 12.3 (2016), e1004790.

[232] Le Song, Alex Smola, Arthur Gretton, Karsten M. Borgwardt, and Justin Bedo. “Supervised feature selection via dependence estimation”. In: *Proceedings of the 24th international conference on Machine learning - ICML ’07*. ACM Press, 2007.

[233] Le Song, Alexander Smola, Arthur Gretton, Justin Bedo, and Karsten M. Borgwardt. “Feature selection via dependence maximization”. In: *JMLR* 13 (2012), pp. 1393–1434.

[234] Jae Hoon Sul, Lana S Martin, and Eleazar Eskin. “Population structure in genetic studies: Confounding factors and mixed models”. In: *PLoS genetics* 14.12 (2018), e1007309.

[235] Shuying Sun, Celia M.T. Greenwood, and Radford M. Neal. “Haplotype inference using a Bayesian Hidden Markov model”. In: *Genetic Epidemiology* 31.8 (2007), pp. 937–948.

[236] S. Joshua Swamidass, Chloé-Agathe Azencott, Kenny Daily, and Pierre Baldi. “A CROC stronger than ROC: measuring, visualizing and optimizing early retrieval”. In: *Bioinformatics* 26.10 (2010), pp. 1348–1356.

[237] S. Joshua Swamidass, Chloé-Agathe Azencott, Ting-Wan Lin, Hugo Gramajo, Sheryl Tsai, and Pierre Baldi. “The Influence Relevance Voter: an accurate and interpretable virtual high throughput screening method”. In: *Journal of Chemical Information and Modeling* 49.4 (2009), pp. 756–766.

[238] S. Joshua Swamidass, Jonathan Chen, Jocelyne Bruand, Peter Phung, Liva Ralaivola, and Pierre Baldi. "Kernels for small molecules and the prediction of mutagenicity, toxicity and anti-cancer activity". In: *Bioinformatics* 21.suppl 1 (2005), pp. i359–i368.

[239] Grzegorz Swirszcz and Aurelie C. Lozano. "Multi-level Lasso for Sparse Multi-task Regression". In: *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*. 2012, pp. 361–368.

[240] Damian Szklarczyk et al. "STRING v10: protein-protein interaction networks, integrated over the tree of life". In: *Nucleic Acids Research* 43.Database issue (2015), pp. D447–452.

[241] Hao-Yang Tan et al. "Epistasis between catechol-O-methyltransferase and type II metabotropic glutamate receptor 3 genes on working memory brain function". In: *Proceedings of the National Academy of Sciences* 104.30 (2007), pp. 12536–12541.

[242] Murat Taşan, Gabriel Musso, Tong Hao, Marc Vidal, Calum A. MacRae, and Frederick P. Roth. "Selecting causal genes from genome-wide association studies via functionally coherent subnetworks". In: *Nat Methods* 12.2 (2015), pp. 154–159.

[243] Margaret A. Taub, Holger R. Schwender, Samuel G. Younkin, Thomas A. Louis, and Ingo Ruczinski. "On multi-marker tests for association in case-control studies". In: *Front. Genet.* 4 (2013), p. 252.

[244] The Wellcome Trust Case Control Consortium. "Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls". In: *Nature* 447.7145 (2007), pp. 661–678.

[245] Timothy Thornton. "Statistical Methods for Genome-Wide and Sequencing Association Studies of Complex Traits in Related Samples". In: *Curr Protoc Hum Genet* 84 (2015), pp. 1.28.1–1.28.9.

[246] Lu Tian, Ash A. Alizadeh, Andrew J. Gentles, and Robert Tibshirani. "A Simple Method for Estimating Interactions Between a Treatment and a Large Number of Covariates". In: *Journal of the American Statistical Association* 109.508 (2014), pp. 1517–1532.

[247] Robert Tibshirani. "Regression shrinkage and selection via the Lasso". In: *J. R. Stat. Soc.* 58 (1994), pp. 267–288.

[248] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. "Sparsity and smoothness via the fused lasso". In: *J Roy Stat Soc B* 67.1 (2005), pp. 91–108.

[249] Roberto Todeschini and Viviana Consonni. *Handbook of molecular descriptors*. Vol. 11. John Wiley & Sons, 2008.

[250] Inma Tur, Alberto Roverato, and Robert Castelo. "Mapping eQTL Networks with mixed Graphical Markov Models". In: *Genetics* 198.4 (2014), pp. 1377–1393.

[251] Alberto Valdeolivas et al. "Random walk with restart on multiplex and heterogeneous biological networks". In: *Bioinformatics* 35.3 (2019), pp. 497–505.

[252] Eliezer M. Van Allen, Nikhil Wagle, and Mia A. Levy. "Clinical analysis and interpretation of cancer genome data". In: *J. Clin. Oncol.* 31.15 (2013), pp. 1825–1833.

[253] Oron Vanunu, Oded Magger, Eytan Ruppin, Tomer Shlomi, and Roded Sharan. "Associating genes and protein complexes with disease via network propagation". In: *PLoS computational biology* 6.1 (2010), e1000641.

[254] Jean-Philippe Vert and Laurent Jacob. "Machine learning for in silico virtual screening and chemical genomics: new strategies". In: *Combinatorial chemistry & high throughput screening* 11.8 (2008), pp. 677–685.

[255] Peter M. Visscher et al. "10 years of GWAS discovery: biology, function, and translation". In: *The American Journal of Human Genetics* 101.1 (2017), pp. 5–22.

[256] Janett Walters-Williams and Yan Li. "Estimation of Mutual Information: A Survey". In: *Rough Sets and Knowledge Technology*. Ed. by Peng Wen, Yuefeng Li, Lech Polkowski, Yiyu Yao, Shusaku Tsumoto, and Guoyin Wang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 389–396.

[257] Xiang Wan et al. "BOOST: A Fast Approach to Detecting Gene-Gene Interactions in Genome-wide Case-Control Studies". In: *The American Journal of Human Genetics* 87.3 (2010), pp. 325–340.

[258] Bo Wang et al. "Similarity network fusion for aggregating data types on a genomic scale". In: *Nature Methods* 11.3 (2014), pp. 333–337.

[259] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. "Deep learning for sensor-based activity recognition: A survey". In: *Pattern Recognition Letters*. Deep Learning for Pattern Recognition 119 (2019), pp. 3–11.

[260] Lili Wang, Takuya Matsushita, Lohith Madireddy, Parvin Mousavi, and Sergio E. Baranzini. "PINBPA: Cytoscape app for network analysis of GWAS data". In: *Bioinformatics* 31.2 (2015), pp. 262–264.

[261] Renxiao Wang, Yipin Lu, and Shaomeng Wang. "Comparative Evaluation of 11 Scoring Functions for Molecular Docking". In: *Journal of Medicinal Chemistry* 46.12 (2003), pp. 2287–2303.

[262] Zhong Wang, Tian Liu, Zhenwu Lin, John Hegarty, Walter A. Koltun, and Rongling Wu. "A general model for multilocus epistatic interactions in case-control studies". In: *PLoS ONE* 5.8 (2010), e11384.

[263] Zi Wang and Giovanni Montana. "The Graph-Guided Group Lasso for Genome-Wide Association Studies". In: *Regularization, Optimization, Kernels, and Support Vector Machines*. 2014, pp. 131–157.

[264] Kyoko Watanabe, Erdogan Taskesen, Arjen Van Bochoven, and Danielle Posthuma. "Functional mapping and annotation of genetic associations with FUMA". In: *Nature communications* 8.1 (2017), p. 1826.

[265] Christian Widmer et al. "Further improvements to linear mixed models for genome-wide association studies". In: *Scientific reports* 4 (2014), p. 6874.

[266] Scott M. Williams, Marylyn D. Ritchie, John A. Phillips III, Elliot Dawson, et al. "Multilocus Analysis of Hypertension: A Hierarchical Approach". In: *Hum Hered* 57.1 (2004), pp. 28–38.

[267] James D Wilson, Melanie Baybay, Rishi Sankar, and Paul Stillman. "Fast embedding of multilayer networks: An algorithm and application to group fMRI". In: *arXiv preprint arXiv:1809.06437* (2018).

[268] Michael C. Wu, Seunggeun Lee, Tianxi Cai, Yun Li, Michael Boehnke, and Xihong Lin. "Rare-variant association testing for sequencing data with the Sequence Kernel Association Test". In: *Am. J. Hum. Genet.* 89.1 (2011), pp. 82–93.

[269] Tong Tong Wu, Yi Fang Chen, Trevor Hastie, Eric Sobel, and Kenneth Lange. "Genome-wide association analysis by lasso penalized logistic regression". In: *Bioinformatics* 25.6 (2009), pp. 714–721.

[270] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. "A comprehensive survey on graph neural networks". In: *arXiv preprint arXiv:1901.00596* (2019).

[271] Zheng Xia, Ling-Yun Wu, Xiaobo Zhou, and Stephen TC Wong. "Semi-supervised drug-protein interaction prediction from heterogeneous biological spaces". In: *BMC systems biology* 4.Suppl 2 (2010), S6.

[272] Bo Xin, Yoshinobu Kawahara, Yizhou Wang, and Wen Gao. "Efficient Generalized Fused Lasso and its Application to the Diagnosis of Alzheimer's Disease". In: *Twenty-Eighth AAAI Conference on Artificial Intelligence*. 2014.

[273] Makoto Yamada, Wittawat Jitkrittum, Leonid Sigal, Eric P. Xing, and Masashi Sugiyama. "High-Dimensional Feature Selection by Feature-Wise Kernelized Lasso". In: *Neural computation* 26.1 (2014), pp. 185–207.

[274] Makoto Yamada, Yuta Umezu, Kenji Fukumizu, and Ichiro Takeuchi. "Post Selection Inference with Kernels". In: *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*. Ed. by Amos Storkey and Fernando Perez-Cruz. Vol. 84. Proceedings of Machine Learning Research. Playa Blanca, Lanzarote, Canary Islands: PMLR, 2018, pp. 152–160.

[275] Makoto Yamada et al. "Ultra High-Dimensional Nonlinear Feature Selection for Big Biological Data". In: *IEEE Transactions on Knowledge and Data Engineering* 30.7 (2018), pp. 1352–1365.

[276] Yoshihiro Yamanishi, Michihiro Araki, Alex Gutteridge, Wataru Honda, and Minoru Kanehisa. "Prediction of drug–target interaction networks from the integration of chemical and genomic spaces". In: *Bioinformatics* 24.13 (2008), pp. i232–i240.

[277] Fan Yang, Rina Foygel Barber, Prateek Jain, and John Lafferty. "Selective inference for group-sparse linear models". In: *Advances in Neural Information Processing Systems*. 2016, pp. 2469–2477.

[278] Jian Yang et al. "Common SNPs explain a large proportion of the heritability for human height". In: *Nature genetics* 42.7 (2010), p. 565.

[279] Sen Yang, Lei Yuan, Ying-Cheng Lai, Xiaotong Shen, et al. "Feature grouping and selection over an undirected graph". In: *Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 2012, pp. 922–930.

[280] Ling Sing Yung, Can Yang, Xiang Wan, and Weichuan Yu. "GBOOST: a GPU-based tool for detecting gene–gene interactions in genome–wide case control studies". In: *Bioinformatics* 27.9 (2011), pp. 1309–1310.

[281] Fuquan Zhang et al. "Increased Variability of Genomic Transcription in Schizophrenia". In: *Sci Rep* 5 (2015), p. 17995.

[282] Han Zhang, Jianxin Shi, Faming Liang, William Wheeler, Rachael Stolzenberg-Solomon, and Kai Yu. "A fast multilocus test with adaptive SNP selection for large-scale genetic-association studies". In: *Eur. J. Hum. Genet.* 22.5 (2014), pp. 696–702.

[283] Qinyi Zhang, Sarah Filippi, Arthur Gretton, and Dino Sejdinovic. "Large-scale kernel methods for independence testing". In: *Statistics and Computing* 28.1 (2018), pp. 113–130.

[284] Yu Zhang, Dit-Yan Yeung, and Qian Xu. "Probabilistic multi-task feature selection". In: *NIPS*. 2010, pp. 2559–2567.

[285] Jingyuan Zhao, Simone Gupta, Mark Seielstad, Jianjun Liu, and Anbupalam Thalamuthu. "Pathway-based analysis using reduced gene subsets in genome-wide association studies". In: *BMC Bioinformatics* 12 (2011), p. 17.

[286] Xiaodong Zheng, Hao Ding, Hiroshi Mamitsuka, and Shanfeng Zhu. "Collaborative matrix factorization with multiple similarities for predicting drug-target interactions". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2013, pp. 1025–1033.

[287] Yang Zhou, Rong Jin, and Steven Chu-Hong Hoi. "Exclusive Lasso for multi-task feature selection". In: *J. Mach. Learn. Res.* 9 (2010), pp. 989–995.

[288] Marinka Zitnik and Jure Leskovec. "Predicting multicellular function through multi-layer tissue networks". In: *Bioinformatics* 33.14 (2017), pp. i190–i198.

[289] Hui Zou and Trevor Hastie. "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2 (2005), pp. 301–320.

[290] Or Zuk, Eliana Hechter, Shamil R Sunyaev, and Eric S Lander. "The mystery of missing heritability: Genetic interactions create phantom heritability." In: *Proceedings of the National Academy of Sciences of the United States of America* 109.4 (2012), pp. 1193–8.