DISS. ETH NO. 26854

EXPLORING THE GENOTYPE-PHENOTYPE RELATIONSHIP USING BIG DATA AND MACHINE LEARNING

A dissertation submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by

MATTEO TOGNINALLI
Ing. Bioing. Dipl. EPF, EPFL

born on 29 August 1992
citizen of Geneva, Switzerland

accepted on the recommendation of

Prof. Dr. Karsten M. Borgwardt, examiner
Prof. Dr. Edward S. Buckler, co-examiner
Prof. Dr. Richard A. Neher, co-examiner
Prof. Dr. William B. Noble, co-examiner

2020

Abstract

Biology is orchestrated by a myriad of beautifully complex phenomena. While scientists have only been able to uncover a tiny part of all existing biological mechanisms, they have done so by methodically relying on data, which sometimes came in enormous amounts. The advent of computational methods have therefore been instrumental for the latest discoveries in the field. Genome-Wide Association Studies (GWAS), for instance, have been able to pinpoint the location of causal mutations that lead to given phenotypes. Advances in computational and statistical methods are themselves driven by newer and larger data sets. In fact, GWAS only came after the large genotyping efforts started bearing their fruit. However, this symbiotic and virtuous relationship between data and methods is sometimes hindered.

In particular, the current genomic efforts have yielded massive data sets, which in turn have led to colossal amounts of results that are highly unorganised and not always comparable. It is therefore hard to combine these rich outcomes to derive new findings and large data sets sit completely unused. Another blocking issue is that proposed computational methods oftentimes rely on simplifying assumptions. While they have allowed certain discoveries, they are also limiting many advanced approaches: machine learning methods to investigate biological phenomena sometimes suffer from oversimplifications, which in turn leads to limited prediction capabilities.

In this thesis, solutions and approaches relying on big data and machine learning are presented to tackle both these issues. A first part focuses on enabling the comparability of GWAS results across phenotypes and study designs. On the data side, we present a large curatorial effort to homogenise GWAS results across a multitude of phenotypes for *Arabidopsis thaliana*. Moreover, we present a new GWAS summary statistics imputation method to palliate to the problem of non-overlapping summary statistics, which limit the downstream applications that rely on GWAS results. A second part introduces a new family of similarity measures for complex structured objects. We propose a new class of kernels relying on optimal transport to better capture differences between structured objects, such as graphs or time series; it can be applied to many biological problems where complex interplay of signals are found. Finally, a third part tackles the problem of complex phenotype prediction. There, we introduce a new deep learning method to predict wheat crop yield by using genotypic information and also accounting for environmental and developmental factors.

In summary, we provide several solutions to mitigate the two undesirable effects in certain situations. Our contributions unlock new ways of combining GWAS results and new directions to model complex biological phenomena: we believe that machine learning will greatly benefit the biological sciences.

Résumé

La biologie est orchestrée par une myriade de phénomènes merveilleusement complexes. Si les scientifiques n'ont pu découvrir qu'une infime partie de tous les mécanismes biologiques existants, ils l'ont fait en s'appuyant méthodiquement sur des données, parfois très nombreuses. L'avènement des méthodes de calcul a donc été déterminant pour les dernières découvertes dans ce domaine. Les études d'association pangénomiques (Genome-Wide Association Studies, GWAS en anglais), par exemple, ont permis de localiser les mutations causales qui conduisent à des phénotypes donnés. Cela étant, les progrès des méthodes de calcul et de statistique sont eux-mêmes dus à des ensembles de données plus récents et plus importants. De ce fait, les GWAS ne sont apparues qu'après que les grands efforts de génotypage aient commencé à porter leurs fruits. Cependant, cette relation symbiotique et vertueuse entre les données et les méthodes est parfois entravée.

En particulier, les efforts de génomique récents ont produit des ensembles de données massifs, qui ont à leur tour conduit à des quantités colossales de résultats très peu organisés et pas toujours comparables. Il est donc difficile de combiner ces riches résultats pour en tirer de nouvelles conclusions et de grands ensembles de données restent totalement inutilisés. Autre problème bloquant : les méthodes de calcul proposées reposent souvent sur des hypothèses simplificatrices. Si elles ont permis certaines découvertes, elles limitent également de nombreuses approches avancées : les méthodes d'apprentissage machine pour étudier les phénomènes biologiques souffrent parfois de simplifications excessives, ce qui conduit à des capacités de prédiction limitées.

Dans cette thèse, des solutions et des approches s'appuyant sur les grandes données et l'apprentissage machine sont présentées pour pallier ces deux problèmes. Une première partie s'attache à permettre la comparabilité des résultats des GWAS entre les phénotypes et les plans d'étude différents. Du côté des données, nous présentons un grand effort de curation pour homogénéiser les résultats des GWAS à travers une multitude de phénotypes pour l'*Arabidopsis thaliana*. En outre, nous présentons une nouvelle méthode d'imputation des statistiques sommaires de GWAS pour remédier au problème des statistiques sommaires sans chevauchement, qui limitent les applications en aval s'appuyant sur les résultats des GWAS. Une deuxième partie présente une nouvelle famille de mesures de similarité pour les objets structurés complexes. Nous proposons une nouvelle classe de kernels reposant sur le transport optimal pour mieux saisir les différences entre les objets structurés, tels que les graphes ou les séries temporelles ; elle peut être appliquée à de nombreux problèmes biologiques où l'on trouve une interaction complexe de signaux. Enfin, une troisième partie aborde le problème de la prédiction de phénotypes complexes. Nous y présentons une nouvelle méthode d'apprentissage profond pour prédire le rendement des cultures de blé en utilisant les informations génotypiques et en tenant compte également des facteurs environnementaux et de développement.

En résumé, nous proposons plusieurs solutions pour atténuer les deux effets indésirables dans certaines situations. Nos contributions ouvrent de nouvelles voies

pour combiner les résultats des GWAS et de nouvelles directions pour modéliser des phénomènes biologiques complexes : l'apprentissage machine est donc et continuera à être bénéfique pour les sciences biologiques.

Acknowledgements

First and foremost, I would like to deeply thank my advisor Prof. Dr. Karsten Borgwardt for his continuous supervision and advice during my doctoral studies. He constantly provided me with the necessary resources to develop my skills, generously funded the attendance to the many conferences of the field, set up exciting research collaborations and put me in front of the relevant challenges to become a machine learning researcher. Moreover, I would also like to acknowledge his strong support in my entrepreneurial side endeavours.

I would then like to extend my gratitude to Prof. Dr. Edward Buckler, Prof. Dr. Richard Neher, and Prof. Dr. William Noble for agreeing to be part of my thesis examination committee as well as Prof. Dr. Caroline Uhler for chairing the examination.

It is hard to put into words the gratitude I have for the co-authors of the publications I worked on. I am extremely thankful to Dr. Bastian Rieck, whose stoic guidance in most of the projects I had the chance to work on was paramount, both for the ones included in this thesis and the ones that were not. He is the best mentor one could hope for and I will cherish his advice and friendship for life. I would like to thank Prof. Dr. Dominik Grimm, who guided me in the very first projects I worked on as a PhD student. His experience, guidance and organization allowed me to be on the fast-track of research since the very first day. Furthermore, I am profoundly grateful to Dr. Damián Roqueiro, whose relentless help and support directly and indirectly enabled many of the contributions in this thesis. I with to thank Dr. Felipe Llinares–López, who guided many of my modeling choices with his extensive machine learning knowledge. I am also enormously indebted to my colleagues and friends Christian Bock and Elisabetta Ghisu, with whom I shared principal authorship on several publications. The fascinating scientific discussions we had were pleasantly alternating with good times that turned into unforgettable memories. Moreover, I would also like to extend my gratitude to Max Horn, whose great software development experience and computational biology know-how combined with our respective desks' proximity nudged many of my projects in the better direction. I also wish to thank Michael Moor for the insightful medical and thought-provoking discussions, I am truly indebted to him for a more rationalist way of seeing the world. Finally, I am also thankful to Thomas Gumbsch, Anja Gumpinger, Dr. Katharina Heinrich, Dr. Catherine Jutzeler, and Caroline Weis who have been wonderful colleagues and friends.

Throughout my doctoral studies, I had the chance to discuss and exchange with many other talented scientists. In that regard, I am truly thankful to all the present and past members of Prof. Karsten Borgwardt's lab in Basel, Dr. Lukas Folkman, Dr. Dean Bodenham, Dr. Xiao He, Dr. Daisuke Yoneoka, Dr. Laetitia Papaxanthos, Dr. Katharina Heinrich, Giulia Muzio, and Leslie O'Bray. Similarly, many people at the Biosystems Science and Engineering department contributed in making my PhD a splendid experience. I would therefore like to extend a warm thank you to Olivier Belli, Arthur Dondi, and Mariia Cherepkova. Additionally, I had to chance to work

with the people from the PhD Students and Postdoc Association of the department (VMB) and I would like to thank all its past and present members for the important role they play for the department and its scientific staff. Lastly, I am truly grateful to all the department's staff and in particular to Cindy Malnasi, for the continuous administrative support.

During my PhD experience, I had fantastic opportunities to collaborate with many researchers from other fields and horizons. I am therefore grateful to Ümit Seren and Prof. Dr. Arthur Korte, for the work on AraPheno and AraGWAS, to Dr. Jakob Nilsson for the rich collaboration on clinical transplant data, and to Prof. Dr. Jesse Poland and Dr. Xu Wang for the exciting work on wheat yield prediction.

Finally, I am extremely thankful to all my friends for all the marvellous moments outside my doctoral studies. I also wish to thank my family: my parents Alexandra and Danilo and my brothers David and Oscar for their understanding and kindness. And last but certainly not least, I am forever indebted to Aryane, for her selfless encouragement and unconditional love throughout these amazing yet very busy years.

Contents

I Introduction 1

1 Introduction 3
1.1 Instilling coherence across large biological data set 4
1.1.1 Genome-Wide Association Studies 4
1.1.2 Organising and ensuring comparability of GWAS results . . . 6
1.2 Embracing complexity in biological phenomena 6
1.2.1 Phenotype prediction 7
1.2.2 Crop yield prediction for crop breeding 8
1.3 Organisation and contributions of this thesis 9
1.3.1 Comparable GWAS for *Arabidopsis thaliana* 9
1.3.2 Imputation of GWAS summary statistics 10
1.3.3 Wasserstein kernels for structured objects 10
1.3.4 Crop yield prediction using deep learning 11

II Enabling comparable GWAS 13

2 Comparable GWAS for *A. thaliana* 15
2.1 Introduction 15
2.2 AraPheno 17
2.2.1 Content and features 17
2.2.2 RNA-Seq data 21
2.2.3 Architecture and implementation 21
2.3 AraGWAS Catalog 22
2.3.1 Content and features 22
2.3.2 Standardised GWAS pipeline 28
2.3.3 Architecture and implementation 29
2.3.4 Concluding remarks 29

3 Imputation of GWAS Summary Statistics 31
3.1 Introduction 32
3.2 Summary Statistics Imputation as Gaussian Process Regression . . . 34
3.2.1 A Gaussian process regression primer 34
3.2.2 Summary statistics imputation 36
3.2.3 Automatic Relevance Determination 37
3.2.4 Implementation 38

3.3 Experimental results . . . 40
3.3.1 Data sets . . . 40
3.3.2 Experimental design . . . 42
3.3.3 COPDGene . . . 43
3.3.4 Insomnia complaints . . . 53
3.3.5 Speed performance . . . 54
3.3.6 Concluding remarks . . . 55

III Wasserstein kernels 59

4 Wasserstein Kernels for Structured Objects 61
4.1 Introduction . . . 62
4.1.1 $\mathcal{R}$-Convolution kernels . . . 62
4.1.2 Optimal transport . . . 63
4.2 Wasserstein Graph Kernels . . . 66
4.2.1 Graph kernels . . . 66
4.2.2 Wasserstein distance on graphs . . . 67
4.2.3 From distance to graph kernels: theoretical considerations . . . 70
4.2.4 Experimental evaluation . . . 78
4.3 Wasserstein Time series Kernels . . . 86
4.3.1 Time series kernels . . . 86
4.3.2 A subsequence-based Wasserstein kernel . . . 88
4.3.3 Theoretical considerations . . . 92
4.3.4 Experimental evaluation . . . 93

IV Crop yield prediction 101

5 Deep learning for crop yield prediction 103
5.1 Introduction . . . 104
5.1.1 Crop yield prediction . . . 105
5.2 Multiple Instance Learning for phenotype prediction . . . 107
5.2.1 Background . . . 107
5.2.2 Data fusion with attention-based MIL . . . 110
5.2.3 Implementation . . . 112
5.3 Experimental results . . . 112
5.3.1 Data set . . . 112
5.3.2 Phenotype prediction . . . 114
5.3.3 Feature importance analysis . . . 120
5.3.4 Concluding remarks . . . 125

6 Conclusions and Outlook 127

Acronyms 133

List of Figures 135

List of Tables 141

Bibliography 143

Part I

Introduction

1 Introduction

In which the motivation behind this doctoral thesis is laid.

Since its inception, biology has been a field that heavily relies on data. From the first inheritance experiments performed by Gregor Mendel on more than 28,000 plants [157] to the large-scale data sets analysed by Ronald Fisher at the Rothamsted Experimental Station [76], large biological evidence has always driven methodological and theoretical advances. More recently, the early successes of the genome sequencing initiatives of the beginning of the new millennium motivated the development of new statistical tools to identify links between genotypes and phenotypes: shortly thereafter came Genome-Wide Association Studies (GWAS) [167]. Similarly, the advent of Next Generation Sequencing techniques have made the availability of sequenced genotypes explode, resulting in studies with an ever-increasing number of genetic variants and participants. The high-dimensional nature of the resulting data sets and the interest in exploring higher-order interaction effects between genetic variants inspired the creation of significant pattern mining [146].

It should therefore come as no surprise that the sheer amount of biological data generated nowadays is seen as *the* key element to unravel life sciences mysteries and incite methodological developments. During the last decade, many have put forward the promises of precision medicine, which, thanks to the abundance of molecular data sets, would enable personalised disease prevention and treatment [9]. Nevertheless, our ability to make sense of large biological data sets is currently hindered in two main ways.

Firstly, the ever-increasing amount of data that is being generated in the biological and medical fields is generally unorganised. It is therefore hard to efficiently combine and cross-reference findings across studies. This is particularly problematic because the data collection efforts continue without any sign of slowdown. Hence, it is necessary to provide solutions to organise these data and to make them coherent and comparable.

Secondly, the current approaches to answer biological questions rely on simplifying assumptions and, by consequence, are oftentimes inapt to capture and model all involved mechanisms. Since biological phenomena are highly complex, simple approaches to elucidate them are limited to scenarios where everything is controlled and understood. However, to grasp realistic problems and answer practical questions, computational methods need to get rid of simplifications and embrace complexity.

With this thesis, our goal is to provide pointers on how to best tackle these two limitations with machine learning and show what practical benefits can result from considering them.

1.1 Instilling coherence across large biological data set

The ease of genotype collection mentioned above led to tens of thousands of data sets with millions of biological measurements for thousands of patients [236]. For each genotyped individual, information about structural variations (SV), i.e. differences between an individual's genotype and a reference genotype, could be pooled. These include Single Nucleotide Polymorphisms (SNPs), for which single bases are different, Insertions and Deletions (InDels), where individuals present additional short sequences or miss some, and Copy Number Variations (CNVs), which indicate differences in the number of copies of single genes. The idea of linking the observed structural variations to the observed variations in phenotypes to better understand how the first potentially causes the second came very naturally. The best example of such analysis is given by the previously mentioned GWAS.

1.1.1 Genome-Wide Association Studies

The goal of GWAS is to identify regions in the genome that could be causal for a specific phenotype. Here, phenotype is meant in the most general sense of the term and can take the form of any observable characteristic of an organism, including binary phenotypes (e.g. diabetic patients versus healthy controls), categorical phenotypes (e.g. patient groups with different responses to a treatment), or continuous phenotypes (e.g. human height). Insights gathered with GWAS therefore have multiple applications: from better disease understanding [154], to personalised treatment plans [9] or to optimise crops for better yield in plant research [53].

More practically, a GWAS consists in collecting the genotypes and phenotypes of a large individual cohorts. Then, association tests between each SNP and the phenotype are run to identify highly associated variations. The association signal can be measured in several ways. For binary phenotypes, a GWAS can rely on simple two sample tests: for each variant, a contingency table counting the different variants for each of the phenotype classes is obtained and used to test for association using a discrete test statistics like a χ^2 test or a Fisher's exact test. Alternatively, linear models are a popular way to test individual variants' association with a phenotype. They rely on the assumption that the phenotypes $\boldsymbol{y}$ can be seen as a linear additive combination of genotype values $\boldsymbol{x}$: $\boldsymbol{y} = \beta_0\mathbf{1} + \beta_1\boldsymbol{x} + \epsilon$, where $\boldsymbol{y}$ and $\boldsymbol{x}$ are the vector of phenotypes and genotypes of the cohort individuals respectively, $\beta_0 \in \mathbb{R}$ is the offset, β_1 the genotype effect, $\mathbf{1}$ is a vector containing ones, and ϵ are the residuals following a known distribution. The parameters can then be estimated using a maximum likelihood estimator and the deviation of β_1 from 0 indicates the

effect size for that given variant, it can then be used in a statistical test to assess if its deviation from 0 is statistically significant. Linear models come in various shapes: linear regression and linear mixed models can be used for continuous phenotypes, logistic regression can be used for binary and categorical phenotypes.

The associations are then summarised with p-values that indicate the degree of the association between the given SNP and the phenotype: under a certain computed threshold, these associations are deemed *significant*, meaning that, statistically, they are very unlikely to result from random effects. Given the high dimensional nature of genotype data and the relatedness of all genomes, certain steps need to be taken to avoid false positives. First, *population structure* correction can be performed: when related individuals are present in the same study, the combined signal of their similar genotypes will augment and false positive associations can be found. The genomic inflation factor [64] can be used to assess the degree of population structure. If this value deviates from 1, correcting for population structure is necessary. This can be achieved by using the principal components of a genetic similarity matrix as covariates or by adapting the significance threshold using the genomic inflation factor mentioned above [63, 189]. Secondly, SNPs need to be filtered to ensure the lowest possible number of false positive results. This is usually performed using the Hardy–Weinberg equilibrium [250], to discard SNPs that are the results of sampling or genotyping errors. SNPs can also be filtered by using their minor allele frequency (MAF); rare variants (i.e. SNPs with MAF below 0.05 or 0.01) need to be removed because standard GWAS are underpowered to uncover associations related to theses SNPs [6]. Finally, conducting so many univariate tests (we sometimes speak about millions of variants) will necessarily result in spurious associations, this is often referred to as the problem of *multiple hypothesis testing*. To avoid this issue, one can control the Family-Wise Error Rate (FWER) by applying a Bonferroni correction [28] (i.e. dividing the significance threshold by the number of conducted tests to obtain a new significance threshold). Alternatively, one can control the False Discovery Rate (FDR) by using the Benjamini–Hochberg procedure [20], this approach is less conservative but avoids a larger number of false negatives. Therefore, the number of considered SNPs in a GWAS can vary considerably due to several quality control procedures performed before and during the study.

Since they are usually conducted on a subset of all mutations, GWAS seldom identify causal SNPs; instead, high association signals are often attributed to SNPs that are in *linkage disequilibrium* with causal variations, meaning that they are both located on highly heritable portions of the genotype. Nonetheless, GWAS have enabled the discovery of thousands of genetic risk loci for hundreds of diseases in humans and continue to be a useful tool in identifying relevant portions among the 3 billion base pairs of the human genome [154].

In other organisms too, GWAS have elucidated some genotype-phenotype relationships: significant associations have been reported for many traits in several organisms such as rice [266], tomatoes [142], fruit flies [156], mice [125], and *A. thaliana* [10].

The rapid increase of genotyped individuals led to an explosion of GWAS results with millions of reported association scores. Navigating and leveraging this wealth of

results is therefore not trivial. While drawing conclusions at the individual phenotype level is relatively easy given a GWAS, combining insights across phenotypes becomes drastically challenging. Not only due to the high complexity of the genome and interplay of its structural variations with the environment (see the next section on complex biological phenomena for more details on this), but also given the high variability and disparateness of the reported results.

1.1.2 Organising and ensuring comparability of GWAS results

First, GWAS results - also referred to as *summary statistics*, i.e. association scores under the form of p-values, are not comparable across studies because of the design of the experiments itself. As briefly hinted at above, a p-value is only useful if used in combination with the significance threshold of the study. Yet, the significance threshold of a GWAS heavily depends on the design of the study. For instance, the number of evaluated SNPs has an impact on the correction used for the significance threshold. Given that most studies are performed on diverse sets of participants and on different sets of variants, the direct comparison of both p-values and significant association indication becomes meaningless.

Moreover, even when the studies are of comparable sizes and setups, the obtained summary statistics cannot be combined because of their highly disparate nature. Data sets obtained from studies that were performed with different genotyping platforms or filtering criteria result in summary statistics for sets of SNPs that are *not* overlapping. Therefore, only a considerably smaller subset of SNPs will have summary statistics for all the considered studies. This severely limits the types of analyses that can be conducted because of the incomplete overlap of the genetic variants and their associations scores.

Hence, it is obvious that, in order to develop the next generation of algorithms that rely on association statistics, a curation effort of these latter is needed. To this end, we explore two ways of palliating to the above-mentioned issues. In Chapter 2 we establish online resources that enable a comparative analysis between GWAS results for the model organism *Arabidopsis thaliana* by re-calculating all GWAS across a large set of phenotypes using a best-practice pipeline, an updated version of the genotype data, and permutation-based statistical significance threshold to account for the phenotypic distributions. The outcome is a catalog of standardised GWAS results for all *A. thaliana* phenotypes that can easily promote comparative analyses across different phenotypes. In Chapter 3 we introduce Ardiss, an accurate, fast and effective method to impute missing association summary statistics in mixed-ethnicity cohorts. The method can be used to ensure a complete overlap of the SNPs of interest when dealing with results from multiple GWAS.

1.2 Embracing complexity in biological phenomena

Having coherent and organised data sets is key to answer questions about biological phenomena. But biological phenomena are highly complex, and existing methods

can fail at taking this into account. For example, GWAS are excellent at identifying individual variants' associations with a given phenotype. However, they could never capture higher-order *interactions* between structural variants in relation with a phenotype. Several approaches have been put forward to overcome these limitations. Significant pattern mining is a prime example [146]: by leveraging statistical tricks, this family of methods enable the automatic identification of larger genomic regions associated with a given phenotype. Similarly, Azencott et al. [11] incorporated information about the underlying biological pathways that connect seemingly unrelated SNPs using notions from graph theory resulting in a higher power in detecting causal SNPs.

This second example showcases how important complex objects are in biology. We often consider the genome as a single string, while forgetting that it translates in an elaborate interplay of signals that are distributed through time and space. Structured objects such as time series and graphs can be found everywhere in biological mechanisms. Being able to handle said data structures is therefore paramount to the development of reliable algorithms for the life sciences. That is why, in Chapter 4 we introduce a new family of similarity measures, or *kernels*, for graphs and time series that are able to better distinguish structured objects as compared to existing measures. In turn, this can be applied to a variety of machine learning methods that rely on kernels for prediction tasks.

1.2.1 Phenotype prediction

The limitation of current approaches in dealing with complex biological phenomena is also well exemplified with phenotype prediction methods. As seen above, one of the ultimate goals of genomics is to be able to leverage genetic information to better understand phenotypic variations and guide decision making. In practice, this is represented by tasks such as phenotype prediction, which has been a problem of interest since the beginnings of genomics [219]. Being able to identify individuals with a high genetic risk for specific conditions has immense potential benefits for public health [202]. Similarly, being able to identify high-potential crops that maximise yield while ensuring resistance and resilience is critical for food security [53, 78]. Throughout the many attempts that were made, it quickly became obvious that genotype alone cannot be used to accurately predict all phenotypic differences. This can be partly imputed to the problem of "missing heritability" [162]: even on large cohorts, the variability of individual genetic variations cannot explain *all* the variability of the heritable portion of observed phenotypes. While several hypotheses have been proposed to explain missing heritability, the problem is yet to be fully solved. A concrete example can be found in the highly heritable trait human height. Human height is reported to have an approximate empirical heritability of 80% but the 50 most associated loci together only account for 5% of the observed phenotypic variance [162, 242]. Several hypotheses around the causes of missing heritability were put forth and some were validated, but part of the heritability still remains unexplained. First, initial GWAS were only focusing on highly associated SNPs. When

considering *all* common SNP variants, the explained variance increases considerably. For human height, accounting for all common SNP variants can explain between 45% to 55% of the observed phenotypic variance [136, 259]. Hence, accounting for all variants increases the explainability thanks to the weak effects of many variants. Another supposition is that genetic variants' effects are not only additive but also present an interactive nature: specific *combinations* of variants are causal for given phenotypes [163]. Finally, epigenetics is also seen as a potential source of heritability [85]. But most importantly, a large portion of the phenotypic variations is caused by environmental and developmental factors, which simply cannot be explained by genomic variability [99].

Therefore, external factors need to be accounted for in order to achieve acceptable phenotype prediction performance. Combining genotypic data with environmental data is all but straightforward. Most statistical genetics studies that have a link with phenotype prediction either try to control the environmental conditions so as not to have to account for them [24] or design experiments to minimise the impact of the environment [188]. Other attempts to incorporate environmental variables in predictive models are usually performed by including covariates that are linked to certain environmental factors. For example, in humans this can be done by including covariates such as sex and age [122]: risk predictors for coronary artery disease reach an area under the receiver operating characteristic (ROC) curve of 0.81 when including sex and age, whereas the performance only reaches 0.64 when solely considering genetic variants [121].

While these approaches showed some promising results, the simple linear combination of genotypic and environmental information is certainly not sufficient to capture interactions between an individual's genome and the environment in which they evolve. Therefore, models able to capture higher-order interactions between genotype and environment are necessary. Plant breeding offers excellent opportunities in this direction and has a rich literature around phenotype prediction.

1.2.2 Crop yield prediction for crop breeding

Food security is a critical problem that recently attracted considerable interest due to the recent global population growth and important environmental changes, as optimised crops are not sufficiently resilient to certain climatic conditions [53, 78]. Since the early 1980s, molecular markers have been used in plant breeding programs to improve quantitative traits with stark economic and social importance [23]. As soon as SNP data on various crops became available, they were extensively used to identify quantitative trait loci (QTLs), genomic areas highly associated with a phenotype of interest. Individuals presenting relevant QTLs were then crossed in phenotypic selection assays using marker-assisted selection [53]. Nevertheless, approaches based on QTLs alone failed to yield successful crops partly due to the problem of missing heritability and to the lack of consideration for interactions between QTLs and environment [22]. From these limitations stemmed the field of genomic selection (GS),

which relies on *all* available genetic markers for phenotype prediction to select crops in a test population that has been genotyped but not phenotyped.

The advantage of genomic selection is to provide indications on a new crop's quantitative trait without the need for phenotyping it. Genomic selection therefore shortens the breeding cycle by reducing the time necessary to phenotype the evaluated crossings [265]. Due to the pronounced interest for good phenotype predictors, many advances in the statistical methods used were made. Similar methods than the one used for phenotype prediction in human were adopted and extended. The ridge-regression model used by Yang et al. [259] for height prediction is one of the most widespread model and was extended to account for interactions between genotypes and environment [40]. While initial contributions modeled the environment as an additional linear component in the regression models, newer approaches attempt to truly capture *interactions* [149], that were then extended to non-linear interactions using kernel methods [54].

More recently, advances in high-throughput phenotyping unlocked new possibilities to capture interactions between genotype and phenotype [172]. In Chapter 5, we present a way do so on a crop yield prediction problem, leveraging a multitude of different environmental-related data sources and combining them with genotypic information. Moreover, we show that using more complex deep learning models that are not based on simplifying assumptions needs not to be made at the detriment of interpretability: we manage to quantify the contributions of individual data sources in the final prediction.

This thesis is therefore a collection of solutions and methods to deal with the issues of coherence and complexity in contemporary biological data sets.

1.3 Organisation and contributions of this thesis

This thesis is organised in four chapters that comprise the main contributions in terms of new resources and methods to better elucidate genotype-phenotype relationships in simple and more structured settings. Each chapter is self contained, meaning that it includes all relevant background for the understanding of the identified problems and the proposed solutions. Chapters are based on published and unpublished work and the detailed contributions are presented below. Chapter 6 contains the conclusions and an outlook towards the open problems along the various directions explored in the thesis.

1.3.1 Comparable GWAS for *Arabidopsis thaliana*

Chapter 2 introduces a set of online resources that organise phenotype and GWAS results for *Arabidopsis thaliana* and prepare them for the next generation of analyses. The chapter is based on the following publications:

- M. Togninalli, Ü. Seren, D. Meng, J. Fitz, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "The AraGWAS Catalog: A curated and standardized Arabidopsis thaliana GWAS catalog". *Nucleic Acids Research* 46:D1, 2018
- M. Togninalli, Ü. Seren, J. A. Freudenthal, J. G. Monroe, D. Meng, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "AraPheno and the AraGWAS Catalog 2020: a major database update including RNA-Seq and knockout mutation data for Arabidopsis thaliana". *Nucleic Acids Research* 48:D1, 2020

For both studies, Matteo Togninalli, Ümit Seren, Karsten Borgwardt, Arthur Korte, and Dominik Grimm conceived the platforms and studies. Matteo Togninalli and Ümit Seren developed and maintained the online platforms. Arthur Korte and Dominik Grimm performed the Genome-Wide Association Study experiments. Magnus Nordborg hosted the resources. Magnus Nordborg and Detlef Weigel provided initial data for the resources. J. Grey Monroe analysed the gene knockout data. Matteo Togninalli, Ümit Seren, Karsten Borgwardt, Arthur Korte, and Dominik Grimm wrote the publications with contributions from all authors.

1.3.2 Imputation of GWAS summary statistics

Chapter 3 introduces Ardiss, an association summary statistics imputation method that works efficiently in mixed-ethnicity cohorts without the need to rely on privacy-sensitive covariates. The chapter is based on the following publication:

- M. Togninalli, D. Roqueiro, I. COPDGene, and K. M. Borgwardt. "Accurate and adaptive imputation of summary statistics in mixed-ethnicity cohorts". *Bioinformatics* 34:17, 2018

Matteo Togninalli, Damiàn Roqueiro, and Karsten Borgwardt designed the study. Matteo Togninalli and Damiàn Roqueiro performed the comparison experiments. The COPDGene Investigators provided access to genotypes of patients. Matteo Togninalli, Damiàn Roqueiro, and Karsten Borgwardt wrote the manuscript.

1.3.3 Wasserstein kernels for structured objects

Chapter 4 presents a new class of kernels for structured data based on optimal transport theory. The chapter is based on the following publications:

- M. Togninalli, E. Ghisu, F. Llinares-López, B. Rieck, and K. Borgwardt. "Wasserstein Weisfeiler-Lehman Graph Kernels". In: *Advances in Neural Information Processing Systems*. 2019
- C. Bock, M. Togninalli, E. Ghisu, T. Gumbsch, B. Rieck, and K. Borgwardt. "A Wasserstein Subsequence Kernel for Time Series". In: *19th IEEE International Conference on Data Mining (ICDM 2019)*. 2019

For the first study, Matteo Togninalli, Elisabetta Ghisu, Bastian Rieck, and Karsten Borgwardt conceived the study. Matteo Togninalli, Elisabetta Ghisu, and Bastian

Rieck implemented the method and performed the experiments. Matteo Togninalli and Bastian Rieck derived the theoretical considerations regarding positive definiteness of the kernels. Felipe Llinares–Lopez gave inputs on the experimental results. Matteo Togninalli, Elisabetta Ghisu, Bastian Rieck, and Karsten Borgwardt wrote the manuscript with inputs from Felipe Llinares–Lopez. For the second study, Karsten Borgwardt highlighted the meaninglessness of certain subsequence time series kernels. Christian Bock, Matteo Togninalli, Bastian Rieck and Karsten Borgwardt designed the study. Christian Bock, Matteo Togninalli, Elisabetta Ghisu, Thomas Gumbsch, and Bastian Rieck performed the experiments. Finally, all authors contributed to the writing of the manuscript.

1.3.4 Crop yield prediction using deep learning

Chapter 5 introduces a new crop yield prediction method that efficiently combines genotype information with multiple data sources related to the investigated plants. The chapter is based on the unpublished work:

- M. Togninalli, X. Wang, J. Poland, and K. Borgwardt. "Deep learning enables accurate grain yield prediction using image and genotype information". Unpublished Manuscript. 2020

Matteo Togninalli, Xu Wang, Jesse Poland, and Karsten Borgwardt imagined the study. Xu Wang coordinated the data acquisition and processing. Matteo Togninalli implemented the methods and computational experiments. Xu Wang and Jesse Poland gave field-informed inputs on the proposed approaches. Matteo Togninalli drafted the manuscript with inputs from the other authors.

Part II

Enabling comparable GWAS

2 Comparable GWAS for *A. thaliana*

In which online resources grouping genotype, phenotype, and association scores for *Arabidopsis thaliana* are presented.

The ease of collection of abundant experimental data from model organisms such as *Arabidopsis thaliana* have made them the ideal subjects of large genetic research efforts. However, despite the large interest of the community for understanding the link between genotypes and phenotypes of *A. thaliana*, it remains hard to compare results across studies, even for such a well-documented and standardised plant.

In this chapter, we present AraPheno [212, 232], a database of *A. thaliana* phenotypes enriched with RNA-Seq data and the AraGWAS Catalog [232, 233], a resource that was developed to allow researchers to easily access and browse standardised GWAS results. AraPheno was initially developed by collaborators [212] and the contributions of the author of this thesis are centered around extensions thereof. However, for sake of completeness, we here report on the entirety of the project. The presented content is partly based on the following publications:

- M. Togninalli, Ü. Seren, D. Meng, J. Fitz, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "The AraGWAS Catalog: A curated and standardized Arabidopsis thaliana GWAS catalog". *Nucleic Acids Research* 46:D1, 2018
- M. Togninalli, Ü. Seren, J. A. Freudenthal, J. G. Monroe, D. Meng, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "AraPheno and the AraGWAS Catalog 2020: a major database update including RNA-Seq and knockout mutation data for Arabidopsis thaliana". *Nucleic acids research* 48:D1, 2020

The chapter is organised as follows. Section 2.1 explains the motivations for providing these resources to the community. Section 2.2 presents the features and particularities of AraPheno. Section 2.3 details the characteristics of AraGWAS and of the standardised GWAS pipeline that was developped for the effort.

2.1 Introduction

Arabidopsis thaliana is a dicotyledonous species and a member of the Brassicaceae or mustard family. It has a rapid life cycle – with only 6 weeks from germination to mature seeds, is easy to cultivate in a controlled environment and limited space, and produces large self progeny. Additionally, its genome, which was the first plant genome to be sequenced, is relatively small (114.5–125 Mb in total). All-in-all, these many advantages have made *A. thaliana* the reference model organism in plant

biology [46, 169, 197, 226]. Moreover, *Arabidopsis thaliana* is a naturally inbred plant, meaning that it can and often does self-pollinate, resulting in lines with completely homozygous genomes. This is a highly desirable property as it allows the study of genetically identical plants and several of their phenotypes under different and *controlled* environmental conditions [10], which represents a colossal advantage when studying complex trait variation in general and interactions between genotype and environment in particular.

In fact, over the past years, large efforts were carried to identify causative genetic variation for a wide variety of different phenotypes. Genome-wide association studies (GWAS) became the reference tool to link genetic variation in a population with the observed phenotypic differences, and after being pioneered in humans, were rapidly adopted and adapted by researchers in the broader biological sciences [94]. GWAS correlate genomic markers with phenotypic differences and report a likelihood of the association under the form of a p-value. On the one hand, it is desirable to have a high marker density to obtain meaningful results: in *A. thaliana*, GWAS have been regularly performed using $214,000$ markers relying on hybridization technology [107]. On the other hand, the statistical power of the analysis increases with the number of samples in the study, hence the interest for increasingly large populations in humans, where, additionally, control over the environmental variables is virtually nonexistent. Having identical samples with homozygous genomes as it is the case for *Arabidopsis thaliana* is therefore very helpful, as it allows for easily reproducible results and enables the re-analysis of collected results once that udpated versions of the genotypic data for the available lines become available.

Therefore, the homorozygous nature of *Arabidopsis thaliana* combined with the availability of high-quality full genome genotype for more than a thousand organisms (available here) provide a rare platform for reproducible research [4] and make *A. thaliana* a prime model organism for genetic research beyond plants [128].

Nevertheless, while these resources enabled the development and benchmarking of tools in the machine learning and data mining communities [147, 176, 223], the lack of centralised information related to phenotypes and GWAS results made it difficult for researchers to (i) conveniently access phenotypic data sets; and (ii) easily compare GWAS results across different phenotypes.

In the next sections, we introduce and detail two online resources for *Arabidopsis thaliana* that centralise and standardise information on phenotypes and GWAS results. AraPheno is a centralised repository of phenotypic information from thousands of *A. thaliana* lines and the AraGWAS Catalog is a catalog of standardised GWAS results computed using the phenotypes of AraPheno. These resources are of great relevance for both the *A. thaliana* and the data mining community as they represent a source of new biological insights as well as one of untapped data for the development and assessment of machine learning methods.

2.2 AraPheno

AraPheno (arapheno.1001genomes.org) is a database for *A. thaliana* phenotypes. It was originally created to organise and centralise all the published phenotypes reported by the research community, and its primary purpose is therefore to provide information about the collected phenotypes and the studies they were obtained from.

2.2.1 Content and features

AraPheno is accessible online at arapheno.1001genomes.org through a user-friendly interface. Phenotypes are grouped by studies, where a study is a collection of multiple phenotypes obtained for a given publication or research effort. Users can therefore select which elements to inspect from a list of phenotypes or a list of studies. However, considering the ever growing number of phenotypes, a fulltext search functionality is available to search for specific phenotypes, studies or other terms. To help users navigate the large diversity of AraPheno data, all phenotypes are linked to plant trait ontologies (bioportal.bioontology.org/ontologies/PTO), therefore relying on a predetermined vocabulary to describe observed traits in *A. thaliana* and allowing users to group phenotypes. Every phenotype reported in AraPheno was also used in a standardised GWAS pipeline and the results of the study are reported in the AraGWAS Catalog, see Section 2.3 for more details.

Moreover, AraPheno also provides access to a comprehensive list of available *A. thaliana* accessions that were collected in the wild (arapheno.1001genomes.org/accessions/). The list regroups meta-information (e.g. geographic positions) as well as information related to the public genotype releases for a given accession, such as RegMap, 1001Genomes or others. For accessions where seeds are available, a link to The *Arabidopsis* Information Resource (TAIR) page where the germplasm can be ordered is provided. Each reported phenotype is therefore linked to a specific accession allowing for easy retrieval of geographical information and for computation of genotype-phenotype associations with constantly updated genotype information, thanks to the above-mentioned homozygous nature of the species. This also enables users to easily access all phenotypes related to a particular accession of interest.

Being a constantly evolving resource, AraPheno's content grew over the last years. A list of currently up-to-date values can be found in Table 2.1. The platform contains 22 Studies relaying information about 462 phenotypes for $1,496$ different accessions. The large volume of presented data implied some technical challenges that were tackled with modern web-development frameworks, as presented in Section 2.2.3.

AraPheno offers several views to display all the relevant information to users. The study view (an example can be found here) presents a description of the study, summary statistics about the reported phenotypes, a link to the relevant publication, and a list of phenotypes. The phenotype view (see Figure 2.1 or this link) lists characteristics of the phenotype such as its ontological attributes, the unit it was measured in as well as details on how it was scored. A list of relevant publication for

the phenotype is shown. It also displays the geographic distribution of the measured accessions as well as some visualizations on the phenotype values themselves: the Explorer widget allows users to further dive into the distribution of a particular phenotype. The view also shows a link to the associated AraGWAS results (see Section 2.3). The accession view (shown in Figure 2.2 or accessible here) provides the coordinates of where the accession was collected from and shows them on a map. It also lists all the reported measured phenotypes for that given accession as well as some summary statistics related to their ontology.

To make the resource practically useful, all information can be downloaded directly from the website in various file formats (CSV, JSON, and PLINK for single phenotypes or PLINK and ISA-TAB for complete studies). Additionally, users can access the data programmatically via a Representational State Transfer Application Programming Interface (REST API, more details can be found at arapheno.1001genomes.org/faq/rest). This enables a fast and direct integration of the data into existing programming pipelines. Moreover, the platform provides the possibility to download the full AraPheno database in a single ZIP component, including all phenotypes and its meta-information (see arapheno.1001genomes.org/faq/content for a detailed explanation of the ZIP content).

AraPheno aims to be a participative resource, that is why it offers the possibility to users to upload their own studies to the website via a user friendly form or via the REST API. The submitted studies go through a manual curation step to ensure all relevant and important information such as scoring details and trait ontology terms are provided. Once the study is approved, the phenotypes are made public and a Digital Object Identifier (DOI) is associated to each of them by DataCite (https://datacite.org/) to make them easily referenceable and citeable in order to encourage researchers to upload their phenotypes even if they are not published yet.

AraPheno also offers two handy tools: a widget to evaluate and visualise the correlation between a set of phenotypes and a tool to apply transformation to any phenotype. Measuring correlation between phenotypes can be useful to decouple the shared environmental and shared genetic effects: several methods rely on phenotypic correlation to map the underlying genetic components [129, 225, 268]. Additionally,

Table 2.1: AraPheno content and summary statistics as of January 2020.

Studies	22
Phenotypes	462
Phenotype Values	193,616
Accessions	7,426
Phenotyped accessions	1,496
RNA-Seq studies	2
Unique RNA-Seq genes	28,819
Total RNA-Seq expression values	20,371,657
Accessions with RNA-Seq data	788

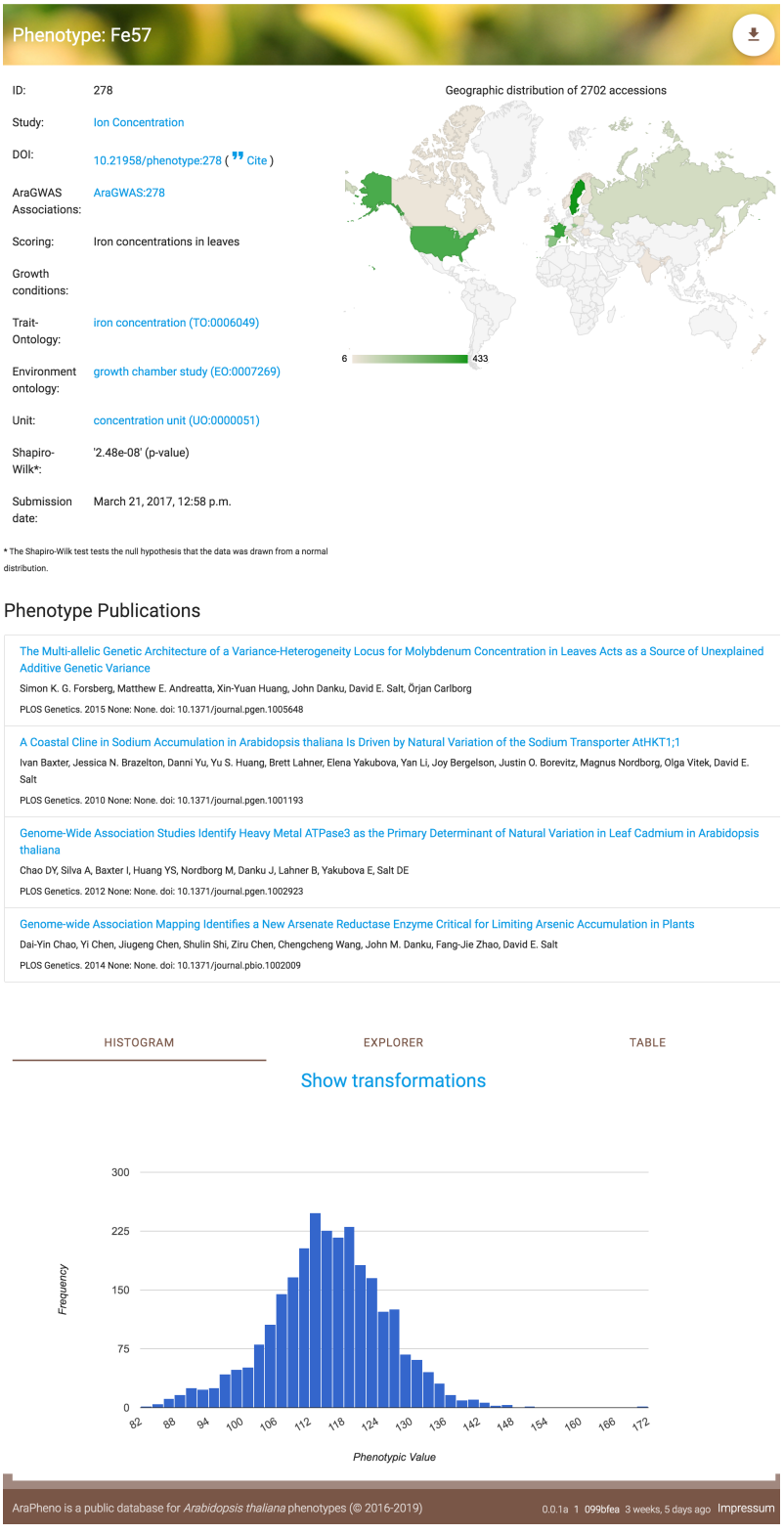


Figure 2.1: AraPheno phenotype view, containing details related to the Iron Concentration in leaves. Users can easily cite the phenotype using the DOI or download the reported values with the download button on the top right.

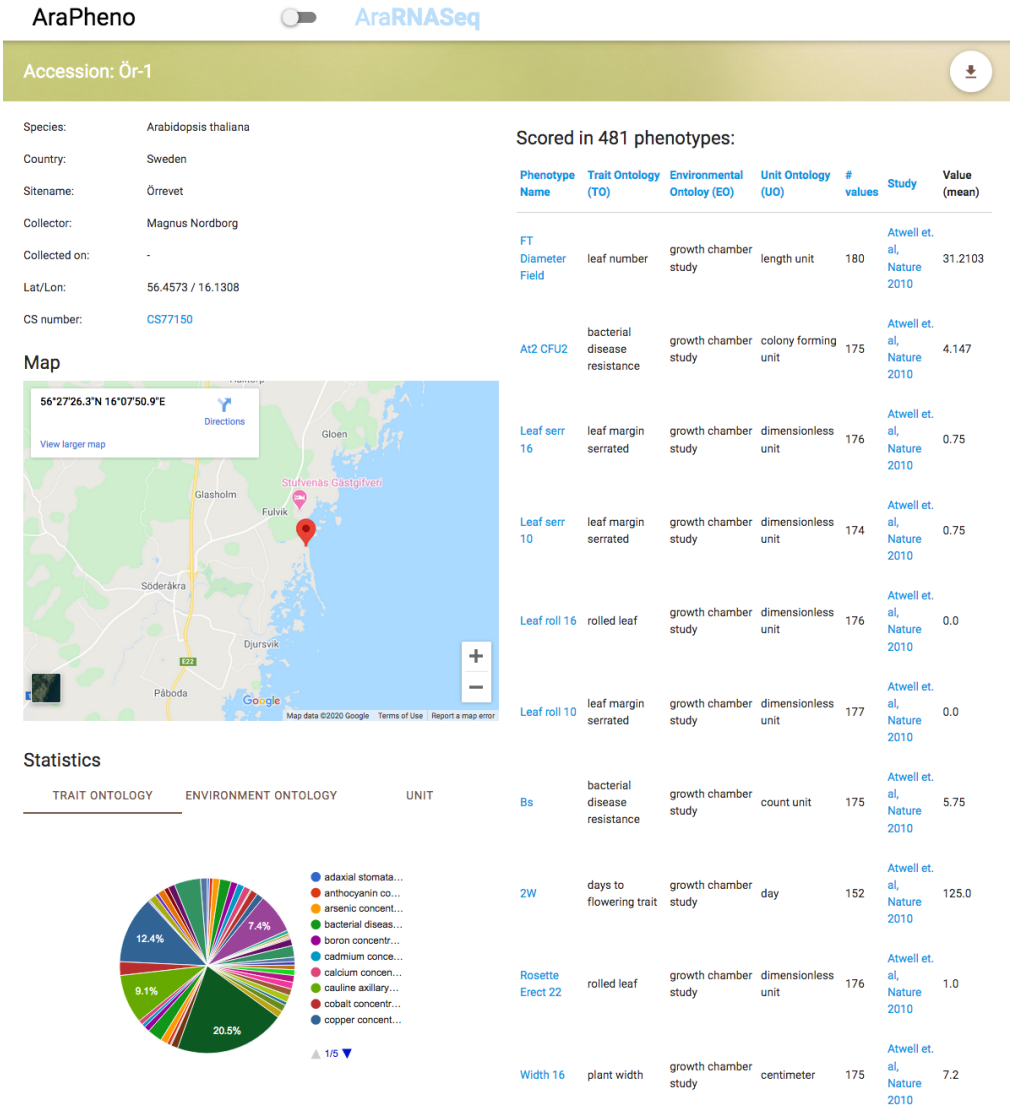


Figure 2.2: AraPheno accession view, containing details related to the Ör-1 accession, collected in Sweden. Users can easily download the details related to the accession with the download button on the top right.

for many statistical analysis, phenotypic values need to be normalised to match the assumptions of the employed statistical test, hence we provide an easy way to apply transformations such as log, anscombe or box-cox transformations to AraPheno phenotypes.

Finally, AraPheno also has a detailed FAQ, tutorials and guided tours to help users understand how to use all the available functionalities.

2.2.2 RNA-Seq data

Recently, AraPheno was extended to also provide gene expression data from RNA-Seq experiments. This major overhaul of the backend of the platform (see Section 2.2.3) allows us to now present the gene expression values of more than 750 accessions for more than 28,800 genes. Unlike regular phenotypes, these measurements do not have a corresponding study in the AraGWAS Catalog as of yet. However, it is possible to treat gene expression data as high-dimensional phenotypic data and run GWAS or transcription-wide association studies (TWAS) on them [244]. Users can access the RNA-Seq data by toggling the switch on the homepage of AraPheno. A clone of the website built around the peculiarities of gene expression is then accessed and can easily be recognised by the theme color of the interface (blue for RNA-Seq and brown for phenotypes).

The AraPheno RNA-Seq interface is built around the study view, which summarises information of a given RNA-Seq study, and a detailed gene view. The RNA-Seq view shows all measured genes as the phenotype study view would show the collected phenotypes. The gene view highlights the distribution of the RNA-Seq values measured for the different accessions. As of now, there are 2 RNA-Seq studies on AraPheno, but this number is expected to grow constantly, similarly to what happened with the number of phenotypes.

2.2.3 Architecture and implementation

AraPheno was built with open-source, popular, and modern web development frameworks. The platform relies on Django (www.djangoproject.com), a Python-based web-application framework. Its popularity simplifies many extensions, the Django REST framework (www.django-rest-framework.org), for example, allowed us to easily build REST endpoints that were then documented using the django-rest-swagger (GitHub link), an open-source swagger implementation for Django REST.

For the backend, Django easily interfaces with the high-performance PostgreSQL (www.postgresql.org) database in which the data are stored. During the last update [232], the backend components were modified to improve performance for multiple users and enable easier download and upload of large amounts of phenotypes. The frontend visualizations are obtained using the free google charts library (developers.google.com/chart) and D3.js (d3js.org), a popular Javascript library for insightful data visualization. The two phenotype manipulation tools (correlation analysis and phenotype normalization) are built using open-source Python libraries: NumPy

(www.numpy.org), SciPy (www.scipy.org), and Pandas (pandas.pydata.org), three very well known high-performance scientific computing and data handling libraries.

Finally, AraPheno is deployed using docker (www.docker.com), an open-source software containerization platform. Docker ensures a reproducible environment for the deployment of the AraPheno platform without encountering issues with dependencies. The platform is hosted under the 1001genomes organization (1001genomes.org) and its framework is accessible as an open-source software on GitHub.

2.3 AraGWAS Catalog

The AraGWAS Catalog (aragwas.1001genomes.org) is a manually curated database for standardised GWAS results for *Arabidopsis thaliana*. It was originally conceived after noticing that, despite the abundance and sharing of collected phenotypes for *A. thaliana* (see Section 2.2), it was impossible to find an overview of all SNP-trait associations that guarantees comparability across phenotypes. In fact, due to the large variety of accessions, phenotypes, and growth conditions, it was hard to find two large-scale studies for which the reported associations scores were comparable. That is why we re-calculated all GWAS for the available phenotypes from AraPheno, using a best practice pipeline (see section 2.3.2) and the most up-to-date version of the genotype data. We then standardised all statistical significance thresholds using a permutation-based approach that accounts for the phenotypic distribution that can differ across phenotypes. Having a standard procedure for phenotype normalization and processing, association computation, and permutation-based significance threshold computation ensures the comparability of the scores and the significance of the reported associations. Finally, the AraGWAS Catalog enables easy access to standardised GWAS results for all AraPheno phenotypes with the latest release of genomic data and promotes comparative analyses across different phenotypes.

2.3.1 Content and features

The AraGWAS Catalog contains GWAS results for all the 462 phenotypes reported in AraPheno using the fully imputed data for $2,029$ *A. thaliana* lines from the 1001 Genomes Consortium [51]. While similar resources can be found for other species such as humans – e.g. the NHGRI GWAS Catalog [154, 248], the public availability of genotypes and phenotypes in *A. thaliana* allows for the systematical re-computation of the GWAS results in a best-practice way to ensure comparability across phenotypes and experiments. Permutation-based significance thresholds are computed for every trait in the catalog to account for various phenotypic distributions. In total, we identified $44,680$ significant SNP-trait associations for *Arabidopsis thaliana*. A detailed list of summary statistics can be found in Table 2.2.

Users can find a sortable table with all available GWAS under the "GWAS Studies" view, where the studies are sorted according to the number of significantly associated hits above the permutation-based threshold. All details about a specific GWA Study

can be visualised in the detailed study view (accessible by clicking on the study name, which is equivalent to the trait name, e.g. aragwas.1001genomes.org/study/144). Figure 2.3 shows an example of the view for a metabolite content phenotype. The view contains all relevant information for the phenotype and details about the study, such as the genotype version used, a link to the AraPheno entry and information about the significance thresholds used (Figure 2.3 - A). Summary statistics about the distribution of significant associations are displayed just below (Figure 2.3 - B) while a list of all associated hits with their respective p-values is accessible right next to it (Figure 2.3 - C). Users can easily filter the associations with the lateral filter option and download the selected associations via a convenient download button (Figure 2.3 - D). Additionally, interactive Manhattan plots are shown in a separate tab of the sudy view and knock-out mutations associations can be visualised in a third tab.

From the detailed study view, users can conveniently click on specific associations (i.e. SNPs in the list) and access an individual association view, as shown in Figure 2.4 (accessible at aragwas.1001genomes.org/study/144/associations/4_1269036). In fact, it can be of interest for downstream analyses to look not only at p-values but also at effect sizes, standard errors, allelic information, and phenotypic distributions for the different allelic groups. All this information can be conveniently visualised in the association view via dynamic plotting or in the accession table and is accessible when downloading study data.

Moreover, other data-centric views are shown across the platform to ease users' quick grasp of the Catalog's content. The "GWAS Hitmap" shows a high level overview of the most associated hits in different regions of each of *A. thaliana*'s 5 chromosomes (see Figure 2.5). The columns summarise the chromosomes while each row reports the 25 strongest hits per chromosome for each of the studies reported in the catalog. Each dot illustrates the top associated hit within the focal region of the chromosome, which is obtained via a sliding window of 250 kbp. The color reports the strength of the association, where red indicates a stronger association (i.e. lower p-value) than yellow. A histogram summarises the density of hits for each chromosome. This view enables users to have an overview of the associated

Table 2.2: AraGWAS Catalog content and summary statistics as of January 2020. Numbers of associated hits are filtered by minor allele count (MAC) > 5. Sig. is an abbreviation of Significant.

Studies	462
Phenotypes	462
Sig. SNP-Trait Associations at $p < 10^{-4}$	1,152,968
Sig. SNP-Trait Associations at Bonferroni threshold	104,874
Sig. SNP-Trait Associations at Permutation threshold	44,680
KO-Mutations	2,088
Sig. KO-Trait Associations at $p < 10^{-4}$	319
Sig. KO Associations at Bonferroni threshold	130
Sig. KO Associations Permutation threshold	15

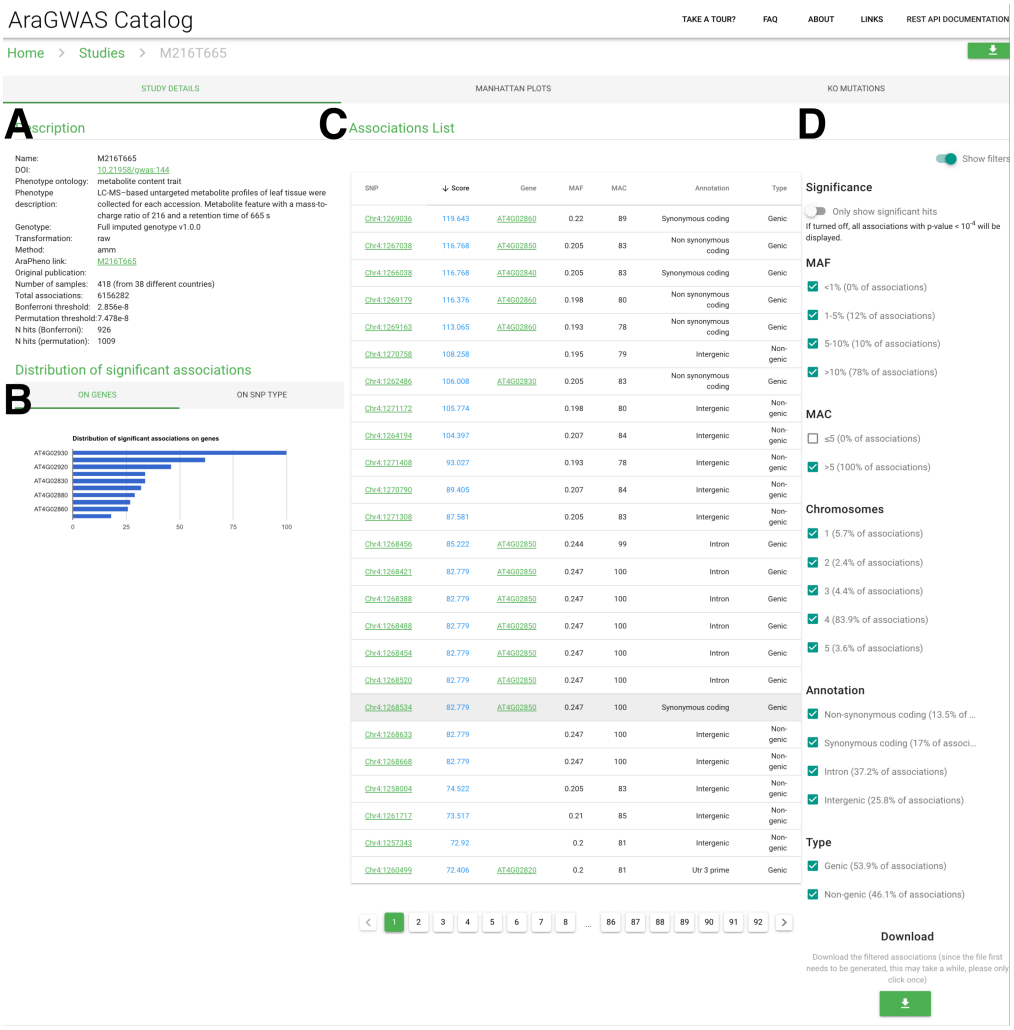


Figure 2.3: AraGWAS Catalog detailed study view, containing details about the GWA Study on M216T665 phenotype. Users can easily download the details related to the filtered associations with the download button on the bottom right. (A) Brief description about study related information with links to the phenotype and publication. (B) Summary statistics about SNP type, impact, annotation and MAF. (C) Sorted list of associated markers. (D) Filters to narrow down the list of associated hits.

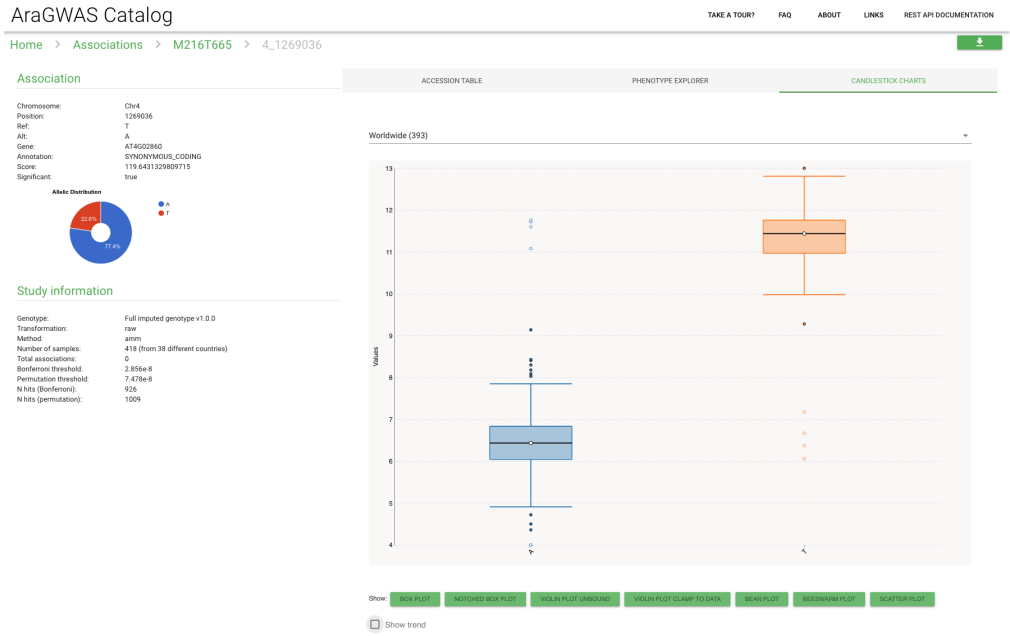


Figure 2.4: AraGWAS Catalog association view, containing details about the Chr4_1269036 accession of the GWA Study on M216T665 phenotype. Users can easily visualise the distribution of the phenotype for different allelic groups.

hits at a glance, potentially highlighting correlations between traits, patterns across chromosome areas, and uncovering pleiotropic effects.

In the "Top Associations" view, users can obtain a list of all associated hits (i.e. p-value$< 10^{-4}$) across all traits stored in the catalog. Each association has additional information related to its variant, e.g. MAF, MAC, type and annotations. These additional entries can be used to filter the hits. Each entry in the table contains links to the detailed view about the study, accession or the gene the variant was found in.

The "Top Genes" view summarises all associated hits detected in genes (or in their close proximity), grouping results by gene name. Table 2.3 shows the 10 genes with the most hits. Clicking on gene names will redirect to the gene-centric view (Figure 2.6), where dynamic visualization can be used to guide the users in their exploration around a region of interest. Information about annotations from SnpEff [49] or gene descriptions extracted from the AraPort11 GFF3 file from the TAIR resource (www.arabidopsis.org/download) are shown and users can directly access the ThaleMine entries for each gene when clicking on them.

Since loss-of-function mutations are an important source of genetic variation in the evolution of plant traits [16, 111], the AraGWAS Catalog also contains associations between reported knockout (KO) mutations and all AraPheno phenotypes. The natural KO mutations are based on loss-of-function alleles of full genes [171] and these new association results are shown in additional views in the catalog. Users can quickly scan through KO-Trait association via the interactive KO Manhattan plots

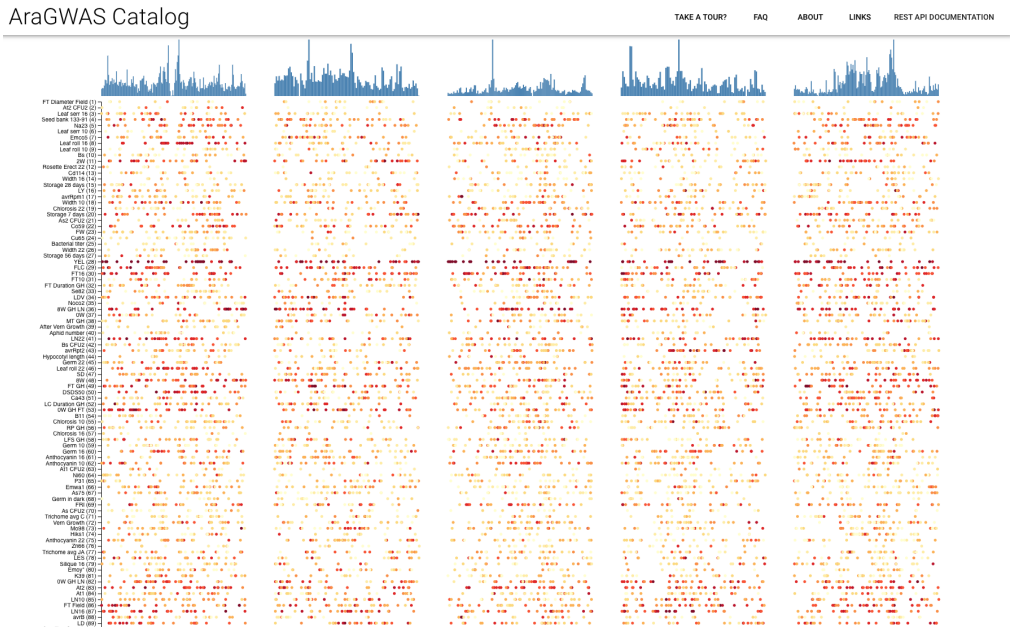


Figure 2.5: AraGWAS Catalog GWAS HitMap, containing a snapshot overview of all associated hits reported in the Catalog. Each column is a chromosome while each row represents a study of the catalog. The color (yellow to red) indicates of the strength of the association.

shown in each detailed study view and, when clicking on one of the dots, they will be redirected to the detailed gene view of the knocked out gene. Additionally, the "Top KO Mutation" view shows a full list of all significant associations between KO genes and traits and the "Top KO Genes" view provides a list of the top associated KO mutation genes, and indicates if any of the KO genes is associated to more than one phenotype.

When analyzing the reported associations between phenotypes and natural KO mutations, associations undetected by SNP-based GWAS could be uncovered. As an example, natural KO alleles in AT1G57570, a mannose-binding lectin superfamily protein expressed during seed germination, were associated with the "number of days of seed dry storage required to reach 50% germination" (DSDS50).

Data from the AraGWAS Catalog are easily downloadable through the web-interface: users can obtain full study results (summary statistics) in HDF5 format and filtered association lists in CSV format. Additionally, all related phenotype data can be obtained from AraPheno, through convenient links. A Download Center regroups various download options such as the full database download, the imputed genotype download or the KO mutation data download to ensure reproducibility of the results. The AraGWAS Catalog also provides a series of REST endpoints for a programmatic access to the data. Users can therefore obtain hits for a specific gene or a given genomic region in their custom analysis pipelines. A full documentation is

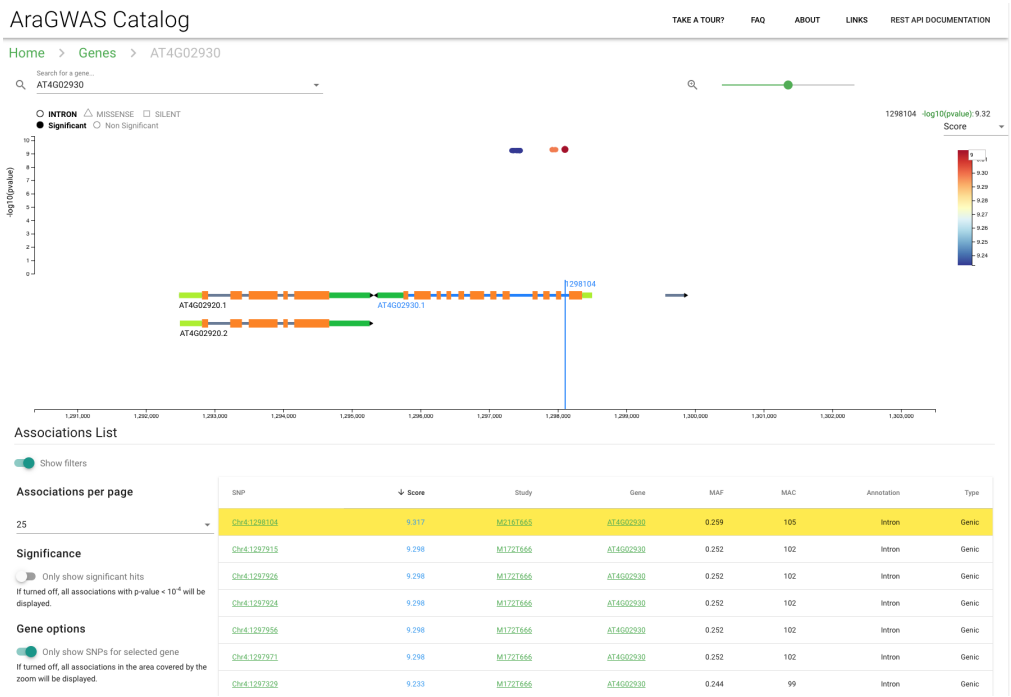


Figure 2.6: AraGWAS Catalog Gene view, showing details of associations around specific genes. Detailed gene descriptions are available when hovering with the cursor over a certain gene.

Table 2.3: AraGWAS Catalog Top Genes according to the number of significant hits as of January 2020. The number of associated loci per gene are based on permutation-based thresholds and minor allele count (MAC) > 5.

Gene name	Short description	N. hits
AT4G02930	GTP binding Elongation factor Tu family protein	195
AT3G20910	Nuclear factor Y, subunit A9	190
AT5G44800	Chromatin remodeling 4	189
AT4G02850	Phenazine biosynthesis PhzC/PhzF family protein	179
AT5G40150	Peroxidase superfamily protein	159
AT5G44820	Nucleotide-diphospho-sugar transferase family protein	146
AT5G45095	Hypothetical protein	144
AT5G45190	Cyclin family protein	118
AT5G22760	PHD finger family protein	117
AT4G30150	Urb2/Npa2 family protein	112

provided online (aragwas.1001genomes.org/docs/). Finally, the AraGWAS Catalog also has a detailed FAQ and offers tutorials and guided tours to new users.

2.3.2 Standardised GWAS pipeline

The GWAS results presented in the AraGWAS Catalog are obtained with a standardised procedure to ensure comparability of the presented associations across different phenotypes. On one hand, all the accessions' genotype values come from the same SNP-Matrix. On the other hand, permutation-based threshold were used to have true comparability between traits with different measuring units and distributions (including non-Gaussian phenotype distributions).

GWAS was performed on all phenotypes of the AraPheno database. For the genotype, the latest version of the 1001 genomes project was used in combination with existing SNP chip data [107], resulting in a SNP-Matrix for $2,029$ accessions and $10,709,466$ segregating markers. Missing values were imputed with BEAGLE v3.0 and standard parameters [37]. For the phenotypes, the untransformed mean value for each phenotype across the accessions' replicates was used for the analysis.

GWAS were conducted using linear mixed models, correcting for population structure in a two-step approach. In the first step, all markers were analysed using an approximation of the mixed model (EMMAX). In the second step, the top 100 markers were analysed again using the full mixed modell (EMMA). The kinship matrix was pre-calculated using all available accessions and removing alleles with a minor allele frequency below 5%. The permutation-based threshold for each phenotype was obtained by repeating this exact procedure multiple times, with permuted trait values for each accession, resulting in a mixed-model where, supposedly, no genotype-phenotype connection is present. The AraGWAS Catalog reports the 5% permutation-based threshold per phenotype, providing a more realistic significance threshold that depends on the phenotypic distribution rather than on generic

statistical assumptions. Results were obtained using GWAS-Flow, a fast TensorFlow and Graphical Processing Unit (GPU)-compatible implementation that enables permutation-based thresholds [80].

It is interesting to notice that the permutation-based threshold is usually more stringent than the classical Bonferroni threshold for a given study (leading to fewer associated hits) but can, in certain cases, be less stringent than this latter (leading to more associated hits). It is therefore important to keep in mind that the permutation-based threshold changes across phenotypes, when comparing traits. Overall, across all studies, the number of significant associations under the permutation-based threshold is considerably lower than the Bonferroni one (see Table 2.2).

2.3.3 Architecture and implementation

Like AraPheno, the AraGWAS Catalog was built using open-source popular web development frameworks. The web-application frontend is a single-page application (SPA) that relies on HTML5 and Javascript and was built using the Vue.js framework (vuejs.org). The multiple visualizations were obtained using libraries like google charts and D3.js and the user interface relies on the Material Design system (material.io) developed by Google to ensure a smooth and pleasant user experience.

Django links two databases in the backend: a first Relational Database Management System (RDBMS) contains all information related to studies while a second elasticsearch engine (www.elastic.co) indexes genes and associations to enable extra-fast retrieval of a large number of association scores. Both databases can be accessed via a RESTful API (see documentation at aragwas.1001genomes.org/docs) and the REST endpoints are built using the Django REST framework and elasticsearch-dsl (elasticsearch-dsl.readthedocs.io), an open-source library for high-level elasticsearch queries in Python.

Finally, the AraGWAS Catalog is automatically deployed using docker coupled with Jenkins (jenkins.io), an open-source automation server that deploys the latest version of the code directly from GitHub. The platform is hosted under the 1001 Genomes Organization and the framework is entirely available on GitHub.

2.3.4 Concluding remarks

We propose two key resources to investigate phenotype-genotype relationships for *Arabidopsis thaliana*. The repositories have become central resources for the *A. thaliana* community [27, 69, 123]. Researchers use AraPheno to publish newly measured phenotypes [32, 65, 113] or use publicly available phenotype data to derive new biological insights [74]. Moreover, both platforms are greatly useful for method development and benchmarking in other areas, such as bioinformatics and machine learning [146, 223]. The rapidly-evolving nature of the two resources make them particularly interesting for these applications. The platforms are currently focused on data from *A. thaliana*, but the code is open-source and could easily be used to extend these resources to new species.

3 Imputation of GWAS Summary Statistics

In which Ardiss, a method to reliably impute GWAS summary statistics in mixed-ethnicity cohorts is presented.

While genome-wide association studies have been instrumental to many discoveries in the field of applied genetics in the last decade, issues around the privacy of the genotypes required to conduct such analyses prevented multiple collaborative efforts. To circumvent these issues, numerous approaches relying solely on GWAS summary statistics were developed. In addition to ensuring a higher privacy preservation of study participants, they often present computational cost advantages. However, due to the diversity in methodologies of conducted GWAS, researchers working with summary statistics are often faced with mismatched SNP sets and therefore need to impute missing ones.

Hence, due to the ubiquitousness of summary statistics based methods, imputation of summary statistics has become a key procedure in many bioinformatics pipelines. Nevertheless, existing imputation methods do not consider the ethnic heterogeneity of the populations originally examined in the GWAS or, to do so, rely on additional information about the original study that are not available to users for the same privacy reasons mentioned above.

In this chapter, we present Ardiss [230], a method to impute missing summary statistics in mixed-ethnicity cohorts using Gaussian Process Regression and Automatic Relevance Determination, without the need to use additional information about the original GWAS. The presented content is based on the following publication:

- M. Togninalli, D. Roqueiro, I. COPDGene, and K. M. Borgwardt. "Accurate and adaptive imputation of summary statistics in mixed-ethnicity cohorts". *Bioinformatics* 34:17, 2018

The chapter is organised as follows. Section 3.1 details the motivation and necessity of reliable summary statistics imputation methods. Section 3.2 formulates the imputation problem, presents existing approaches and introduces Ardiss. Section 3.3 introduces the experimental setup of the performed experiments and discusses the experimental results obtained with Ardiss compared to state-of-the-art methods.

3.1 Introduction

GWA Studies have been key to identifying associations between traits and genetic variants in populations. For more than a decade, GWAS have been performed in a wide variety of organisms: plants and crops [142, 168, 266], animals [125, 156], and humans [79]. The marked interest for these analyses has pushed researchers to share results via public databases web services, for humans [248] and other model organisms, such as *Arabidopsis thaliana*, as thoroughly discussed in Chapter 2. To facilitate the exchange around GWAS and the dissemination of results, the scientific community has increasingly shared so-called *association summary statistics*. These results are usually found in the form of p-values or Z-scores. They are used for meta-analyses, gene-based association tests, fine-mapping, conditional association methods, and to investigate the polygenic nature of complex phenotypes [181].

The stark popularity of methods relying on summary statistics for downstream analyses stems from two main advantages: (i) summary statistics offer noticeable computational cost benefits when compared to genotype-based methods; and (ii) they are relatively unaffected by any privacy-related concerns that inevitably impact genotype data. Nonetheless, given the large diversity of GWAS pipelines used by practitioners, scientists working with summary statistics often face very disparate data sets with non-overlapping single-nucleotide polymorphisms (SNPs). For example, even when relying on the same genotyping arrays, using different filtering criteria will lead to an incomplete overlap of the genetic variants. Likewise, when comparing results obtained from two different populations, some variants might have been discarded in one study while kept in the other, due to their relative abundance with respect to a fixed minor allele frequency (MAF) threshold. The incomplete overlap of genetic variants across multiple GWAS severely limits the outcome of downstream evaluations such as meta-analyses: SNPs with values missing from a study will in most cases be discarded. Therefore, in order to maximise the reach and power of downstream analyses, missing summary statistics need to be imputed. That is why, over the last years, several summary statistics imputation methods were proposed as standalone software solutions.

The focus of this chapter lies on *association* summary statistics, namely p-values and Z-scores, which summarise the strength with which a genomic region is associated to a given trait. We do not specifically target other type of summary statistics that are related to the *population* studied in the GWAS, like genotype counts or allele frequencies. Imputation of Z-scores is therefore the goal of this and related works, but the theoretical foundations derived for these summary statistics can also be used with other ones such as allele frequencies or β values [249].

Existing Z-scores imputation methods can be divided into (i) methods relying solely on the Z-scores (e.g. ImpG-Summary [182]); and (ii) methods requiring additional information about the original GWA Study (e.g. DISSCO [257], ImpG-SummaryLD [182], DISTMIX [135]). While both types rely on an external reference panel of genotyped individuals, the former relies on the assumption that the correlations between SNPs of the study samples are the same as the ones observed

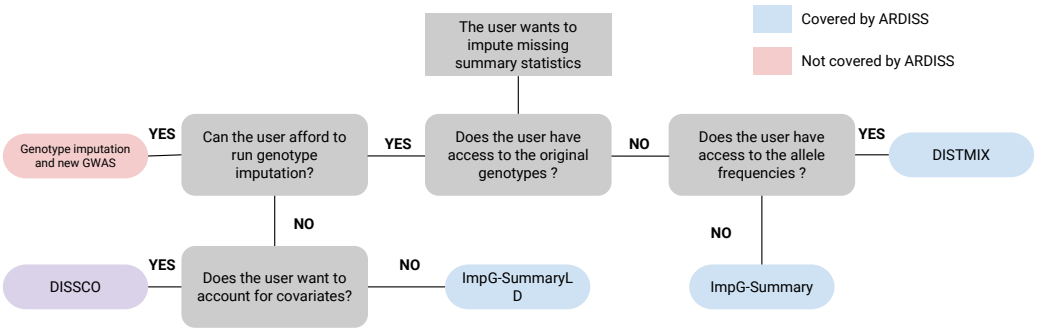


Figure 3.1: Decision flowchart for the choice of the best-suited association summary statistics imputation method. No alternative method fits all scenarios, while ARDISS covers all of them without needing additional information about the original study. Accounting for covariates is not necessary if the covariates were taken into account during the original study (see Section 3.3.3).

in the reference panel, an assumption which is oftentimes violated in practice. The latter, on the other hand, uses additional information in the form of other summary statistics (such as allele frequencies), covariates, detailed ethnic composition of the original GWAS population, or even the original genotypes to estimate the correlation between SNPs in the studied population and tend to give more accurate imputation results.

Therefore, depending on the availability of additional data sources, researchers who wish to impute association summary statistics are faced with many options. Figure 3.1 contains a decision flowchart to guide users in the choice of the best-suited imputation method, given their scenario.

The first distinction is made if the genotypes used in the original GWAS are accessible. One might wonder about the need to impute summary statistics if genotypes are available: why not rely on well-established genotype imputation methods such as `MaCH` [138], `IMPUTE2` [108] or others [39, 213] to impute the genotypes of the missing SNPs and then recompute the GWAS scores? Despite looking like an interesting alternative, this option is often impractical due to the hefty computational cost of genotype data imputation. In fact, when the number of SNPs to be imputed is high (10^6), the genotype imputation can take up to weeks of dedicated cluster computing for studies with thousands of participants. In these situations, users can use methods like IMPG-SUMMARYLD [182] or DISSCO [257], which rely on the original study to model the SNPs covariance relationship.

If genotype data is not available, users need to evaluate if the original study relied on a mixed-ethnicity cohort to generate the summary statistics at hand. In most human studies, designers want the cohorts to be as homogeneous as possible in order to avoid any spurious associations due to population stratification. However, in practice, since ethnicity is self-reported and individuals might miss out on some of

their true genetic background, mixed-ethnicity cohorts are quite common and the obtained Z-scores reflect these considerations. Therefore, Z-scores are oftentimes derived from non-homogeneous cohorts and the imputation of missing values need to account for this to avoid yielding false positives. To do so, researchers can use DISTMIX [135], which tackles the problem of mixed-ethnicity cohorts. DISTMIX requires the allele frequencies of the original study to estimate the ethnic composition of the studied population prior to imputation.

Nevertheless, when imputing summary statistics in an association study, we cannot take lightly the fact that some methods require additional sources of data to perform an accurate imputation. In particular, because these additional data may be unavailable or hard to obtain or, in a worst-case scenario, they can pose a threat to the privacy of the individuals if used inappropriately [105].

Hence, when looking at all available possibilities in Figure 3.1, it is easy to notice that there is no one-fits-all method that can reliably and easily be used in all situations. That is why we developed Ardiss, a fast (highly parallelizable) and accurate summary statistics imputation method that can accurately approximate the ethnic composition of a GWAS population from the summary statistics alone. It does not need any additional information of the study participants, hence guaranteying their privacy, by relying on Automatic Relevance Determination (ARD). ARD is often used in Gaussian Process Regression to perform feature selection in high-dimensional spaces [155]. We use ARD to automatically weight the contribution of individual reference panel members to mimic the mixed-ethnicity composition of the study, see Section 3.2.3 for more details.

3.2 Summary Statistics Imputation as Gaussian Process Regression

3.2.1 A Gaussian process regression primer

Gaussian Processes are a versatile class of statistical models that can be used for both classification and regression problems. In this short primer, which draws inspiration from the book of Rasmussen and Williams [196], we focus on the regression case. A Gaussian Process (GP) is a stochastic process (i.e. a collection of random variables) and it can be seen as defining a *distribution over functions*:

Definition 1 (Gaussian process, from [196])**.** *A* Gaussian process *is a collection of random variables, any finite number of which have a joint Gaussian distribution.*

A Gaussian process can entirely be described by its mean function $m(\boldsymbol{x})$ and its covariance, or kernel function, $k(\boldsymbol{x}, \boldsymbol{x'})$. The Gaussian process can then be written as:

$$f(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x'})). \tag{3.1}$$

The random variables of the GP therefore represent the value of the function $f(\boldsymbol{x})$ at position $\boldsymbol{x}$. GPs can therefore be used to model a given function by capturing the

right mean and covariance between the associated variables. In practice, we take the mean function $m(\boldsymbol{x})$ to be zero, for notational and implementation simplicity. This implies that, since the GP's random variables have a joint Gaussian distribution, we can draw values from them at input value X with $\boldsymbol{f} \sim \mathcal{N}(\mathbf{0}, K(X, X))$.

However, drawing random function values from the prior given by the covariance function k is not particularly interesting nor useful. But we can leverage knowledge from the training data points X with output values $\boldsymbol{f}$ to predict on test points X_* with test outputs $\boldsymbol{f_*}$. The joint distribution of training outputs $\boldsymbol{f}$ and test outputs $\boldsymbol{f_*}$ is given by:

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f_*} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right). \tag{3.2}$$

where $K(X, X)$ is a $n \times n$ matrix containing the pairwise covariances between training points, $K(X, X_*)$ is a $n \times n_*$ matrix containing the covariances between training and testing points, and similarly for $K(X_*, X)$ and $K(X_*, X_*)$. We then need to *condition* the joint Gaussian prior distribution on the observations to make the selected functions agree with the observed training points. Using Gaussian identities (see appendix A.2 of Rasmussen and Williams [196]), we obtain:

$$\begin{aligned} \boldsymbol{f_*}|X_*, X, \boldsymbol{f} \sim \mathcal{N}(&K(X_*, X)K(X, X)^{-1}\boldsymbol{f}, \\ &K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X_*)). \end{aligned} \tag{3.3}$$

From there, one can sample the test output values and use the variance to estimate the certainty of a given prediction.

Simple Gaussian Process regression is a form of *lazy learning*, where the model only leverages the training data at time of inference. However, the covariance functions can also contain hyperparameters. These can be set manually by taking into account expert knowledge but could also greatly benefit from being tuned to best fit a given data set. Leveraging Bayesian principles it is easy to optimise the parameters of a covariance function. We can obtain the *marginal likelihood* of a given model by taking the integral of the likelihood times the prior:

$$p(\boldsymbol{y}|X) = \int p(\boldsymbol{y}|\boldsymbol{f}, X)p(\boldsymbol{f}|X)d\boldsymbol{f}. \tag{3.4}$$

In realistic settings, we prefer to use noisy observations of the data given by $y = f(\boldsymbol{x}) + \epsilon$ over the non-noisy values $\boldsymbol{f_*}$. The marginal likelihood refers to the marginalization over the function values $\boldsymbol{f}$ and can conveniently be used as a "goodness-of-fit" measure for a given covariance function and its hyperparameters $\boldsymbol{\theta}$. One can then maximise the marginal likelihood via gradient descent using the gradients of Equation 3.4 with respect to the hyperparameters $\boldsymbol{\theta}$ (see Section 3.2.3 for a practical application).

3.2.2 Summary statistics imputation

Multiple methods were proposed to impute association summary statistics. DIST [134], ImpG-Summary [182], DISSCO [257], and DISTMIX [135] are considered state-of-the-art techniques for the imputation of Z-scores and differ in how they handle additional data. Despite their differences, they all rely on an external reference panel of genotyped individuals for the imputation of missing values. Typical examples of such reference panels are the ones provided by the 1000 Genomes Project for humans [1] or the 1001 Genomes Project [4] for *Arabidopsis thaliana*. Moreover, all these methods share a common function: they impute missing Z-scores by approximating summary statistics using a multivariate Gaussian distribution over neighboring SNPs' values. In practice, we differentiate between available – or typed – Z-scores (Z_t) and missing – or untyped – Z-scores (Z_u) . Using the linkage disequilibrium (LD) structure for neighboring SNPs, all the above-mentioned methods impute the missing values using variations of the following formula:

$$Z_{u|t} = \Sigma_{ut}\Sigma_{tt}^{-1}Z_t \tag{3.5}$$

where Σ_{ut} is the correlation matrix between untyped and typed SNPs and Σ_{tt} is the correlation matrix between typed SNPs. The correlations are obtained by computing the Pearson's correlation coefficient between SNP genotypes in an external reference panel, for which genotypes are available for both typed and untyped SNPs. In practice, this approach can be seen as a naïve Gaussian Process Regression (see Section 3.2.1) that uses a simple linear kernel k and 0 mean:

$$f(\boldsymbol{x}) \sim \mathcal{GP}(0, k(\boldsymbol{x}, \boldsymbol{x'})) \tag{3.6}$$

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \sum_{i=1}^{d} x_i x_i' \tag{3.7}$$

where $\boldsymbol{x}$ and $\boldsymbol{x'}$ are the standardised feature vectors of two SNPs (i.e. the standardised genotype values for every individual in the reference panel). Analogously to what described in Equation 3.3, the Gaussian Process then outputs predicted means and variance for the missing values according to the following formulas:

$$\begin{aligned} \boldsymbol{f_u}|X_i, X_t, \boldsymbol{f}_t &\sim \mathcal{N}(\mu_K, \sigma_K) \\ \mu_K &= K(X_u, X_t)K(X_t, X_t)^{-1}\boldsymbol{f}_t \\ \sigma_K &= K(X_u, X_u) - K(X_u, X_t)K(X_t, X_t)^{-1}K(X_t, X_u) \end{aligned} \tag{3.8}$$

where $\boldsymbol{f_u}$ and $\boldsymbol{f_t}$ are the generalizations of Z_u and Z_t, respectively; X_u and X_t are the matrices of features for the untyped and typed SNPs and $K(X, X')$ is the $n \times n'$ matrix of the covariance values evaluated at all pairs of points using (3.7). To enhance the readibility of our notation, we will refer to $K(X_t, X_t)$, $K(X_u, X_u)$, $K(X_u, X_t)$ and $K(X_t, X_u)$ as K_{tt}, K_{uu}, K_{ut} and $K_{ut}^{\top}$, respectively.

In order to consider the noise present in the observed data, methods usually add a noise component to the covariance data between typed data as follows:

$$K_y = K_{tt} + \sigma^2_{noise} I \tag{3.9}$$

and replace K_{tt} by K_y. This step is usually related as $\Sigma^{adj}_{tt} = \Sigma_{tt} + \lambda I$ in the summary statistics imputation literature.

The formulas reported here are the basis of all the techniques mentioned at the beginning of this section. Some methods (DISSCO and DISTMIX) build on top of these to account for mixed-ethnicity cohorts, but they do so by using additional information about the study population. The user either needs to report the original genotypes, the allele frequencies of the study genotypes or a manual estimation of the population structure. This information can then be used to compute adjusted partial correlations between SNPs. Nevertheless, these requirements are not ideal in a realistic setting: when access to the original genotypes is possible, genotype imputation should be preferred [182] and allele frequencies are often not shared due to privacy concerns [68, 105].

3.2.3 Automatic Relevance Determination

To deal with mixed ethnicity cohorts without the need to consider additional sources of information from the original GWAS, we rely on automatic relevance determiation (ARD) and present Ardiss (ARD for Imputation of Summary Statistics). Ardiss is a summary statistics imputation method that solely uses the typed association statistics and an external reference panel of genotypes. Automatic relevance determination enables feature selection while fitting a Gaussian Process. This can be achieved by adding weights to each feature used in the kernel construction and to learn the weights during the training procedure so as to best fit the observed values.

As highlighted in Section 3.2.1, a Gaussian Process is characterised by its mean function In our case, since the mean is zero, we can compute the marginal likelihood (or evidence) to evaluate how the parameters fit the observed data using the following equation:

$$\log p(\boldsymbol{y}|X, \boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{y}^\top K_y^{-1}\boldsymbol{y} - \frac{1}{2}\log|K_y| - \frac{n}{2}\log(2\pi) \tag{3.10}$$

We can then fit the Gaussian Process by maximizing the likelihood. Therefore, as suggested in Section 3.2.1, we use a gradient based optimiser on the partial derivatives of the marginal likelihood with respect to the hyperparameters:

$$\frac{\partial}{\partial\theta_j}\log p(\boldsymbol{y}|X) = \frac{1}{2}\mathrm{tr}((\boldsymbol{\alpha}\boldsymbol{\alpha}^\top - K_y^{-1})\frac{\partial K_y}{\partial\theta_j}) \quad \text{where} \quad \boldsymbol{\alpha} = K_y^{-1}\boldsymbol{y} \tag{3.11}$$

We can now incorporate ARD. In our case, each feature in the kernel is an individual genotype in the reference panel. We therefore aim at weighing the contribution of the individual genotypes so as to ideally match the original ethnic distribution and

Algorithm 1 `ARDISS_get_weights`

Input: Standardised genotypes for typed SNPs $X_t \in \mathbb{R}^{N \times d}$, typed Z-scores $Z_t \in \mathbb{R}^N$, window size w, optimiser Opt
Output: Average ARD weights across chromosome

$W \leftarrow \lfloor \text{length of } X/w \rfloor$, $weights \leftarrow \emptyset$
for k in $\{1...W\}$ **do**
  ▷ Slice the array to get batch samples
  $X_{batch} \leftarrow X_{t,i\bullet} \quad i \in \{(k-1) \cdot w ... k \cdot w\}$
  $Z_{batch} \leftarrow Z_{t,i} \quad i \in \{(k-1) \cdot w ... k \cdot w\}$
  ▷ Initialise the ARD weights to a vector of d ones
  $\sigma^2_{\text{ARD}} \leftarrow \mathbf{1}_{1 \times d}$
  for i in $\{1...Opt.maxiter\}$ **do**:
    ▷ Compute the kernel matrix
    $K_y \leftarrow X_{batch} \text{diag}(\sigma^2_{\text{ARD}}) X^\top_{batch} + \sigma_{\text{noise}} I$
    $\alpha \leftarrow K_y^{-1} Z_{batch}$
    ▷ Compute the σ_{ARD} gradients with Equation (3.11)
    $grads \leftarrow \frac{1}{2}\text{tr}((\alpha\alpha^\top - K_y^{-1})\frac{\partial K_y}{\partial \theta_j})_{j=\{1...d\}}$
    ▷ Update the ARD weights with the optimiser of choice
    $\sigma_{\text{ARD}} \leftarrow Opt.update(grads)$
  Append σ_{ARD} to $weights$
Return average of $weights$ along second axis

the reported Z-scores. This is a proxy to represent the population of the original GWAS as closely as possible. The new linear kernel function then becomes [155]:

$$k_{\text{ARD}}(\boldsymbol{x}, \boldsymbol{x'}) = \sum_{i=1}^{d} \sigma_i^2 x_i x_i' \tag{3.12}$$

We finally fit the σ_i values by using a gradient descent optimiser on the negative log-likelihood in Equation (3.10). The partial derivative of K_y with respect to σ_j is the outer product of the genotype values for sample j across the SNPs of interest multiplied by $2\sigma_j$.

3.2.4 Implementation

Ardiss combines ARD with a moving-window imputation of untyped GWAS Z-scores. The algorithm proceeds in two steps. Initially, we iterate over the typed Z-scores of one chromosome to obtain the consensus weight for each sample of the reference panel. Then, we use the weighted genotype values to run the moving-window imputation across the chromosome.

The first phase of the procedure relies on an external library, *GPflow* [166], a TensorFlow [2] based library for Gaussian Process optimization. Since we have hundreds

Algorithm 2 `ARDISS`

Input: Standardised genotypes from reference sample $X \in \mathbb{R}^{M \times d}$, typed Z-scores $Z_t \in \mathbb{R}^N$, window size w, optimiser Opt
Output: Imputed Z-scores

1: Split genotypes in typed and untyped $X_t, X_u \leftarrow X$
2: $\sigma_{\text{ARD}} \leftarrow$ `ARDISS_get_weights`(X_t, Z_t, w, Opt)
3: ▷ Element-wise multiplication followed by standardization further speeds up operations
4: $X_{i\bullet} \leftarrow$ Standardise $X_{i\bullet} \odot \sigma_{\text{ARD}} \quad i \in \{1...M\}$
5: $X_{t,\text{window}} \leftarrow X_{t,i\bullet} \quad i \in \{0...w\}$
6: $Z_{t,\text{window}} \leftarrow Z_{t,i} \quad i \in \{0...w\}, Z_u \leftarrow \emptyset$
7: $N \leftarrow$ length of X_t
8: $K_{tt} \leftarrow X_{t,\text{window}} X_{t,\text{window}}^\top + \sigma^2_{noise} I$
9: Compute K_{tt}^{-1}
10: ▷ Boundary conditions are treated differently
11: **for** i in $\{\frac{w}{2} + 1...N - \frac{w}{2}\}$ **do**
12: Update $X_{t,\text{window}}$, $Z_{t,\text{window}}$ and K_{tt}
13: ▷ Use Sherman-Morrison formulas
14: $K_{tt}^{-1} \leftarrow$ `update_inverse`(K_{tt}^{-1}, K_{tt})
15: $X_u \leftarrow$ `get_untyped_snps_for_window()`
16: $K_{ut} \leftarrow X_u X_{t,\text{window}}^\top$
17: $Z_{u,\text{window}} \leftarrow K_{ut} K_{tt}^{-1} Z_{t,\text{window}}$
18: Append $Z_{u,\text{window}}$ to Z_u
19: Return Z_u

of thousands of available Z-scores, deriving the weights on a single large window enclosing all the typed SNPs is computationally impossible, due to the many matrix inversions required ($O(n^3)$). We therefore perform the optimization on subsets of SNPs in a window-based approach, as shown in lines $1-7$ of Algorithm 1. This implementation can benefit from parallel computing on graphics processing units (GPUs), as enabled by GPflow, and allows breakneck runtimes (see Section 3.3.5). Any gradient-based optimiser can be used for the optimization of the weights (see Input of algorithm 1). We observed that the RMSProp optimiser implementation of GPflow with a learning rate of 0.1 and momentum of 0.001 yields good results.

After the ARD weights have been optimised for the individual windows, we average them across the chromosome (line 17) and we use the following formula to impute the untyped values:

$$Z_{u|t} = K_{ut}^{\text{ARD}} [K_{tt}^{\text{ARD}} + \sigma^2_{noise} I]^{-1} Z_t \tag{3.13}$$

Where the entries of K_{tt}^{ARD} are given by Equation (3.12). To speed execution up, all the genotypes are multiplied element-wise with the ARD-derived weights and standardised (line 4 of Algorithm 2). Imputation is then run with a moving-window

using a fixed number of neighboring SNPs (a window size of 100 SNPs gives excellent results under different scenarios) rather than the commonly used approach of separating the data in chunks of fixed base pairs sizes. The moving-window approach enables faster matrix operations, accelerating execution. In particular, ARDISS implements the Sherman-Morrison formula, that enables obtaining the inverse of the correlation matrix in $O(n^2)$ rather than $O(n^3)$. Moreover, our imputation procedure always centers the window around the SNPs that are being imputed, ensuring to always find the strongest LD structures. SNPs at the boundary of the chromosome are treated slightly differently and are imputed with all the window-size SNPs at the boundary. In total, N loops are performed (where N is the number of typed SNPs). The covariance matrix K_{tt}, its inverse and the typed Z-score vector Z_t are all initialised before iterating through all typed SNPs (lines $5-9$). Every iteration then updates the necessary entries (lines $11-14$), rapidly retrieves X_u for the missing SNPs located between the two central typed SNPs (e.g. between the 50th and the 51st typed SNPs for a window size of 100) using specific Python data structures (line 15) and imputes their Z-score values (lines $16-17$).

The overall complexity of ARDISS is $O(Nkw \cdot \max(w, d))$ for the weight learning step (Algorithm 1) and $O(N(w^2 + n_u wd))$ for the imputation step (Algorithm 2), where N is the number of typed SNPs, k is the maximum number of iterations of the optimiser, w is the window size, d is the number of samples in the reference panel and n_u is the number of untyped SNPs in a single window. Moreover, we can assume that, on average, $n_u = \frac{N_u}{N}$, where N_u is the overall number of untyped SNPs and have a final runtime complexity of $O(Nw^2 + N_u wd)$ for the imputation step. Considering the simple inner products needed to obtain the covariance matrix, ARDISS scales linearly for the number of samples in the reference panel, given a fixed number of SNPs and a fixed window size. Empirical validation is reported in Figure 3.12.

3.3 Experimental results

In order to evaluate the performance of ARDISS in diverse use cases, we devise several experiments. Here, we report the results obtained on two data sets as well as on runtime experiments.

3.3.1 Data sets

COPDGene

We obtained genotype data from participants in the COPDGene study [198]. The aim of the study is to identify risk factors of genetic nature associated to chronic obstructive pulmonary disease (COPD). The study was originally performed on two ethnic groups: African Americans (AA) and non-Hispanic whites (NHW). After combining the samples of the two populations, we keep $615,906$ SNPs that overlapped in both datasets. Of these SNPs, we removed the ones that did not fulfill the following

Table 3.1: Sample size details of the COPDGene cohort. The column "Case" refers to individuals who were diagnosed with COPD. The number of SNPs in the intersection of both populations is 615,906 and we take this as the starting point of our analysis.

Population	Disease status			Gender	
	case	control	Total	male	female
African Americans (AA)	821	1,826	2,647	1,498	1,149
Non-Hispanic whites (NHW)	2,812	2,534	5,346	2,816	2,530
Total	3,633	4,360	**7,993**	4,314	3,679

criteria: (a) minor allele frequency > 0.01; or (b) Hardy-Weinberg equilibrium $> 1.0 \cdot 10^{-6}$. Furthermore, due to genotyping errors, some combinations of samples and SNPs had missing genotypes. For these cases, the missing SNP values were imputed as described in [48]. Of the 7,993 individuals in the combined data set, 3,633 are patients diagnosed with COPD (cases) and 4,360 are controls. Table 3.1 provides additional details. This combined data set is then subsampled to create cohorts of mixed ethnicities as described below.

Randomised cohorts of mixed ethnicity. To simulate mixed-ethnicities cohort, we create 11 randomised partitions of the combined COPDGene data set. On the two extremes, we have homogeneous population of 100% AA and 100% NHW samples, respectively. In between, we artificially create cohorts with a mixed ethnic composition by increments of 10%, i.e., 90% AA with 10% NHW; 80% AA with 20% NHW, all the way to 10% AA with 90% NHW. Additionally, we randomly sample individuals from the two populations in a stratified manner to ensure the same ratio of cases/controls per population. We set all randomised partitions to contain the same number of samples: 2,313.

Association analysis. For each of these randomised partitions, we conduct a GWAS using a linear mixed model to account and to correct for population structure in the mixed cohort [190]. The analyses are performed using FaST-LMM [145] and for each of the 615,906 SNPs, we obtained a Z-score of association. It is important to mention that, when imputing summary statistics in a real-life scenario, the genotypes of the individuals in the study will, most likely, not be available. However, having access to the original genotypes of the COPDGene study allows us to create randomised cohorts of varying ethnic composition and to perform the corresponding association tests. The obtained Z-scores are the starting point to the execution of the evaluated imputation methods.

SNP masking. To evaluate the performance of ARDISS and of its comparison partners, we have to simulate the absence of certain Z-scores of association. We therefore randomly mask 10% of the 615,950 SNPs across the whole genome (stratifying by chromosome) and consider those as missing (i.e. *untyped SNPs*). The other 90% (*typed SNPs*) are the ones for which we know the Z-score and that are used to

impute the untyped ones. This randomization is repeated 10 times in order to get a good genome-wide coverage. The genomic locations of the SNPs are based on the hg19 version of the human genome. All the evaluated methods are asked to impute all the missing SNPs, for a total of 11,671,761 imputed SNPs.

Insomnia complaints

Thanks to online repositories such as the Genome-wide Repository of Associations Between SNPs and Phenotypes (GRASP) [137], we were able to download additional summary statistics. The GRASp catalog currently contains association scores for more than 2,000 GWAS. We focus on a study aimed at identifying the genetic risk factors associated with insomnia complaints [100]. The original study is a large-scale study conducted on 113,006 individuals of self-reported European descent and their samples were obtained from the May 2015 release of the UK Biobank [222].

Data processing. We downloaded the results file #2 from GRASP with full summary statistics on both males and females. Among the different summary statistics, we use BETA – the β of the logistic regression, – and SE – the standard error of the logistic regression β. The Z-score for each SNP is then computed as BETA/SE. We restrict our analysis to a single chromosome: chromosome 12. Of the original 430,235 Z-scores in chromosome 12, we randomly mask 10% and impute them. In a similar way as for the COPDGene data set, we perform the masking 10 times.

Reference panel

All imputation methods rely on an external genotype reference panel to perform imputation. For our analyses we rely on the reference panel from the 1000 Genomes Project, ref. 1, release 3 [1]. The panel contains 14 populations grouped in 4 super-populations (see Table 3.2) and is based on the hg19 version of the human genome.

3.3.2 Experimental design

We benchmark Ardiss with ImpG-Summary [182], the most used summary statistics imputation method that cannot account for mixed-ethnicity cohorts, and DIST-MIX [135], a method that accounts for ethnicity of the original population, but does so by relying either on the study's original allele frequencies or on a manually-provided ethnic composition estimate of the original study.

To assess the performance of each method, we compare the imputed Z-scores with the original Z-scores on the untyped SNPs. The most commonly found evaluation metrics are the Pearson's correlation coefficient, the R^2 score and the root-mean-square-error (RMSE) between the imputed and original (masked) values. We compute all these metrics, focus on the correlation coefficient in our analysis but also report the details on R^2 scores and RMSE, see Table 3.5.

All the runtime analyses were performed on a dedicated server running Ubuntu 14.04.5 LTS, with 2 CPUs (Intel® Xeon® E5-2620 v4 @ 2.10GHz), 8 GPUs

Table 3.2: Details of the samples in the 1000 Genomes Project that are used in our analyses as reference panel. The four superpopulations are: AFR (African), AMR (admixed American), EAS (East Asian), EUR (European).

Superpopulation	Population	Name	Samples
AFR	ASW	African-American SW	61
	LWK	Luhya	97
	YRI	Yoruba	88
AMR	CLM	Colombian	60
	MXL	Mexican-American	66
	PUR	Puerto Rican	55
EAS	CHB	Han Chinese	97
	CHS	Southern Han Chinese	100
	JPT	Japanese	89
EUR	CEU	CEPH (Utah residents)	85
	FIN	Finnish	93
	GBR	British	89
	IBS	Spanish	14
	TSI	Tuscan	98

(NVIDIA® GeForce® GTX 1080), and 128 GB of RAM. The code is implemented in Python 3. To evaluate the runtime required by each method, we run them independently on our server, with no other concurrent processes. We report imputation runtime for every chromosome separately as the imputation process is highly parallelizable, depending on the available computing capabilities. Speed measurements were taken for the imputation of all the missing SNPs when using Ardiss and ImpG-Summary. Given the slow nature of DISTMIX, we could only run it on five chromosomes (chromosomes 18 to 22). The same setup is used to run the speed assessments for varying reference panel sizes and window sizes.

3.3.3 COPDGene

As highlighted in the first part of Section 3.3.1, the COPDGene populations are used to create different cohorts with precise mixtures of ethnicities. The obtained cohorts enable a thorough analysis of the weights computed by Ardiss and an assessment of the accuracy of the different imputation methods under different conditions.

Weights optimization

Before imputing missing Z-scores, Ardiss optimises and outputs a specific weight for each sample from the reference panel. The individual weights can then be aggregated according to the ethnic background of the sample they are related to and used to reconstruct the population composition. Figure 3.2 shows the composition obtained when looking at the weights obtained on the COPDGene mixed cohorts. Since no

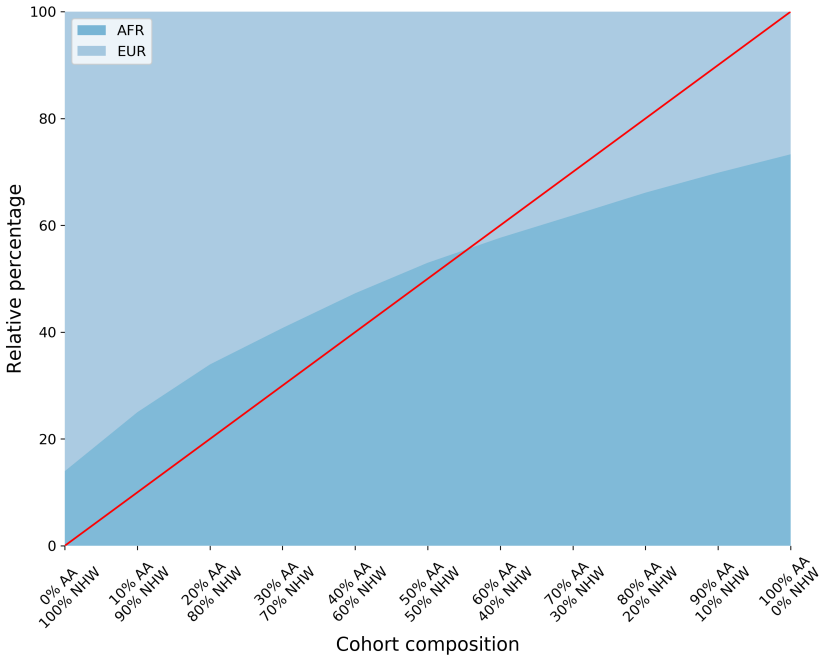


Figure 3.2: Stacked contributions of individuals from the two major super populations of interest (African-descent, AFR, and European-descent, EUR) obtained by ARDISS for sets of different ethnic compositions of the original study cohort. The theoretical composition is represented in red along the diagonal.

existing method can reconstruct the population structure of the original study from Z-scores alone, a comparison with other baselines is infeasible.

ARD also picks up residual signal from the other populations of the reference panel, as shown in Figure 3.3 and Figure 3.4. This weak noise contamination is partly due to the moving-window strategy used when optimizing the ARD weights that forces the averaging of the weights across different LD regions of the chromosome. We keep these contributions during the imputation procedure that follows.

We also compare the weights obtained with ARD to the ones derived by DISTMIX using the allele frequencies of the original study samples. The overall correlation between the two sets of weights is 0.839 and 0.936 when looking only at the populations of interest, as shown in Figure 3.5. Hence, ARDISS reconstructs the study population using only the typed Z-scores, without the need for allele frequencies of the original study.

Imputation performance

Once the weights are computed, ARDISS and the three comparison partners are applied to the Z-scores obtained from the GWAS performed on the 11 mixed-ethnicity cohorts detailed in Section 3.3.1. Figure 3.6 reports the performance in terms of

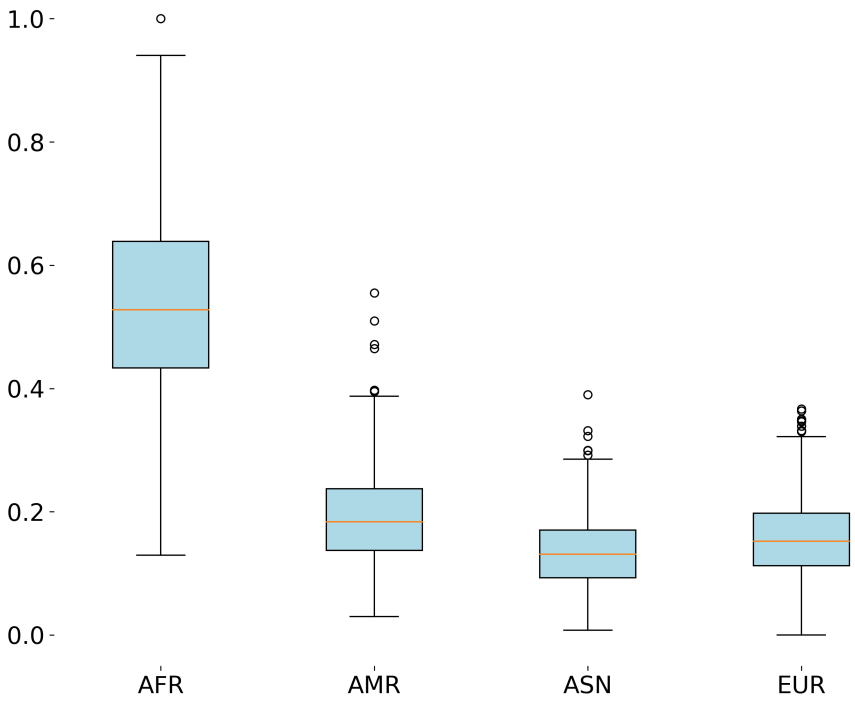


Figure 3.3: Relative contribution of individual samples as detected by the weights obtained by ARD for the 100% AA | 0% NHW mix of population. Some residual weights for non-African populations are picked up. Boxplots are obtained by taking the weight output by ARDISS, i.e. one per sample from the reference panel, and grouping them by their super-population code. Super-population codes are reported in Table 3.2.

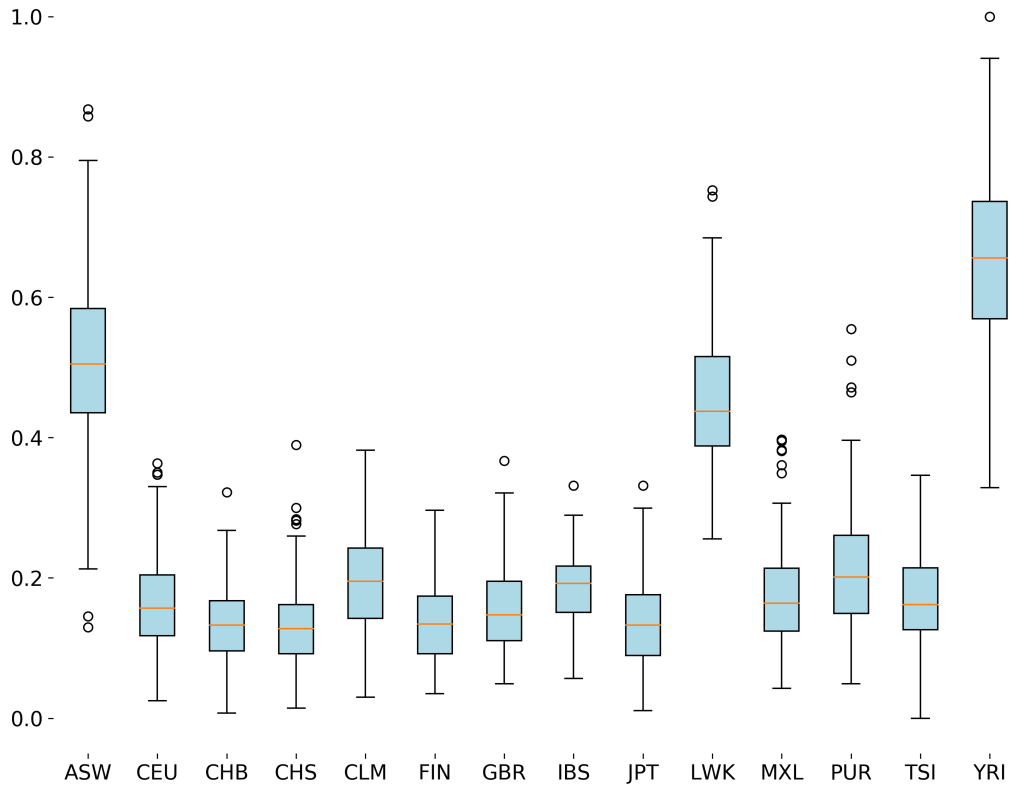


Figure 3.4: Relative contribution of individual samples as detected by the weights obtained by ARD for the 100% AA | 0% NHW mix of population. Some residual weights for non-African populations are picked up. Boxplots are obtained by taking the weight output by ARDISS, i.e. one per sample from the reference panel, and grouping them by their population code. Population codes are reported in Table 3.2.

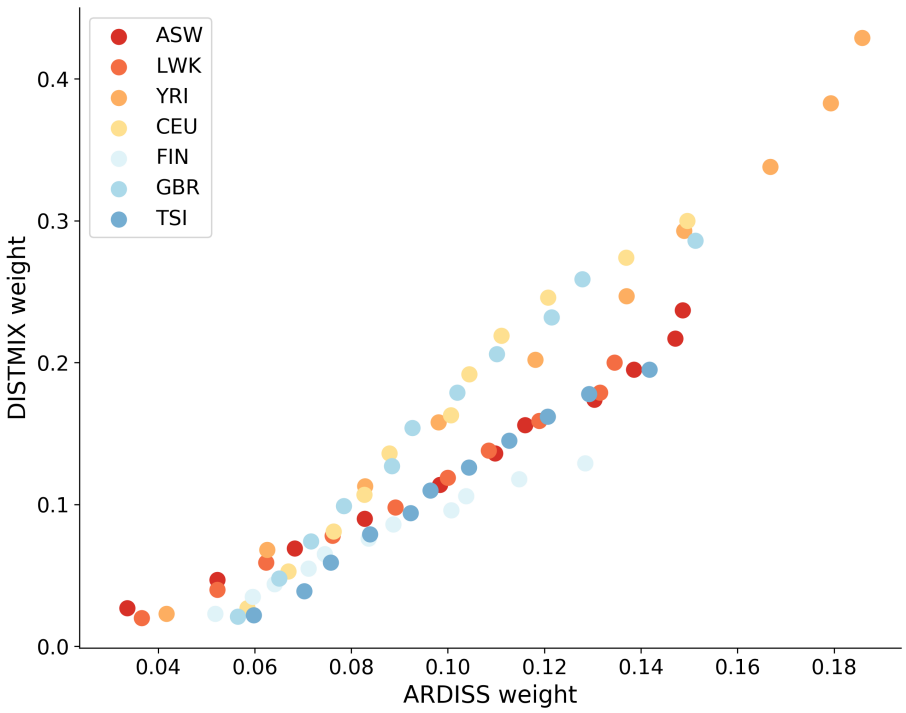


Figure 3.5: Weights obtained by ARDISS (x axis) and by DISTMIX using the allele frequencies of the original study (y axis) for a selection of populations. The color code indicates the population to which the weight belongs and the different points are obtained from the various sets of ethnicity mixture. Population codes are reported in Table 3.2.

Pearson's correlation coefficient of the three methods across the different mixed-ethnicity cohorts.

Overall, ARDISS reports better performance when compared to IMPG-SUMMARY and DISTMIX across all mixtures of population (see Table 3.5 for the complete results). The three methods all perform better with the homogeneous cohort of 100% non-Hispanic whites than with the cohort of 100% African American samples. This is hypothetically caused by two reasons: (i) there are more samples of European descent in the reference panel (379 EUR vs 246 AFR), and (ii) populations of African descent have a higher genetic diversity and less LD [42], making it harder to encompass all the haplotype diversity with few reference samples.

When looking at the performance of ARDISS with respect to the one of IMPG-SUMMARY, the improvement is stronger with non-mixed cohorts. The 0% AA | 100% NHW, 90% AA | 10% NHW and 100% AA | 0% NHW are the ethnic mixtures for

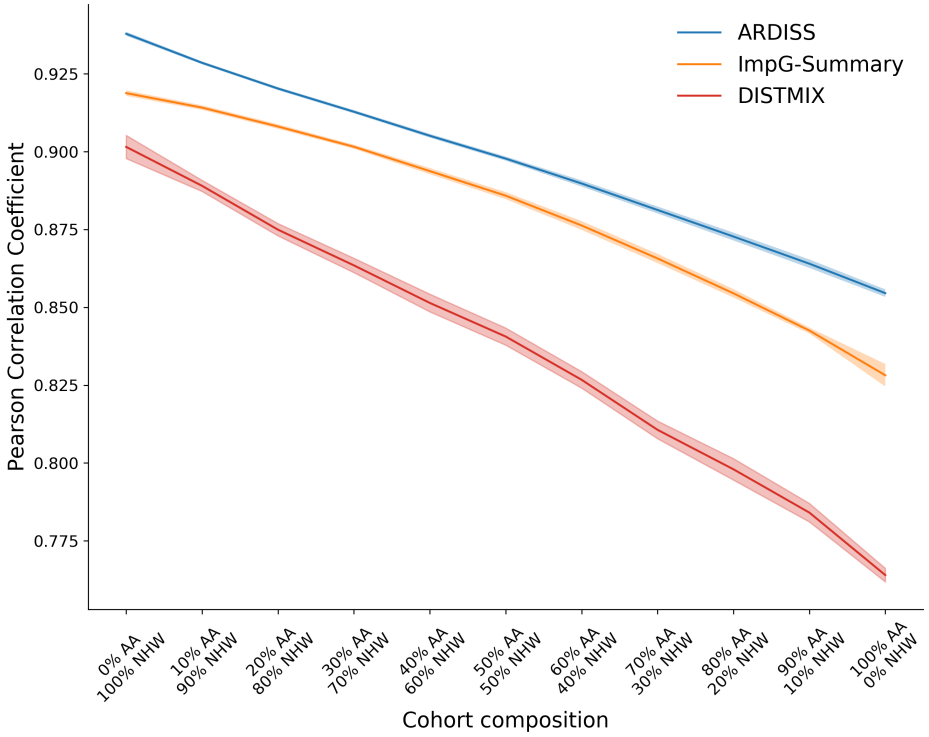


Figure 3.6: Pearson's correlation coefficients obtained during full genome imputation across different mixtures of ethnicity sets using ArDISS and comparison partners. ImpG-Summary is run using all the samples in the reference panel and DISTMIX computes the optimal weights from the allele frequencies. The shaded area represents the standard deviation interval across the 10-fold validation.

which ArDISS clearly outperforms ImpG-Summary, with 2.08%, 2.55% and 3.20% improvements respectively, as shown in Figure 3.7. Interestingly, the improvement on the African-American population is considerably higher, hinting at the ability of ArDISS to draw information from other individual samples that might not be in the same super population group. With increasing admixture, the gain derived by weighting different individual contributions decreases: at 50% AA | 50% EUR, the improvement over ImpG-Summary is down at 1.34%. This is because the underlying weight distribution gets closer to the actual distribution in the reference panel, which is the one used by ImpG-Summary. In fact, all the available samples in the reference panel were used with ImpG-Summary to mimic a realistic scenario with limited knowledge on the original study population.

Similarly, ArDISS does considerably better than DISTMIX, with improvements ranging from 4.03% for 0% AA | 100% NHW to 11.85% for 100% AA | 0% NHW, as shown in Figure 3.8.

Furthermore, we also measure the performance of ImpG-Summary using (i) only EUR samples, (ii) only AFR samples, and (iii) a combination of both on chromo-

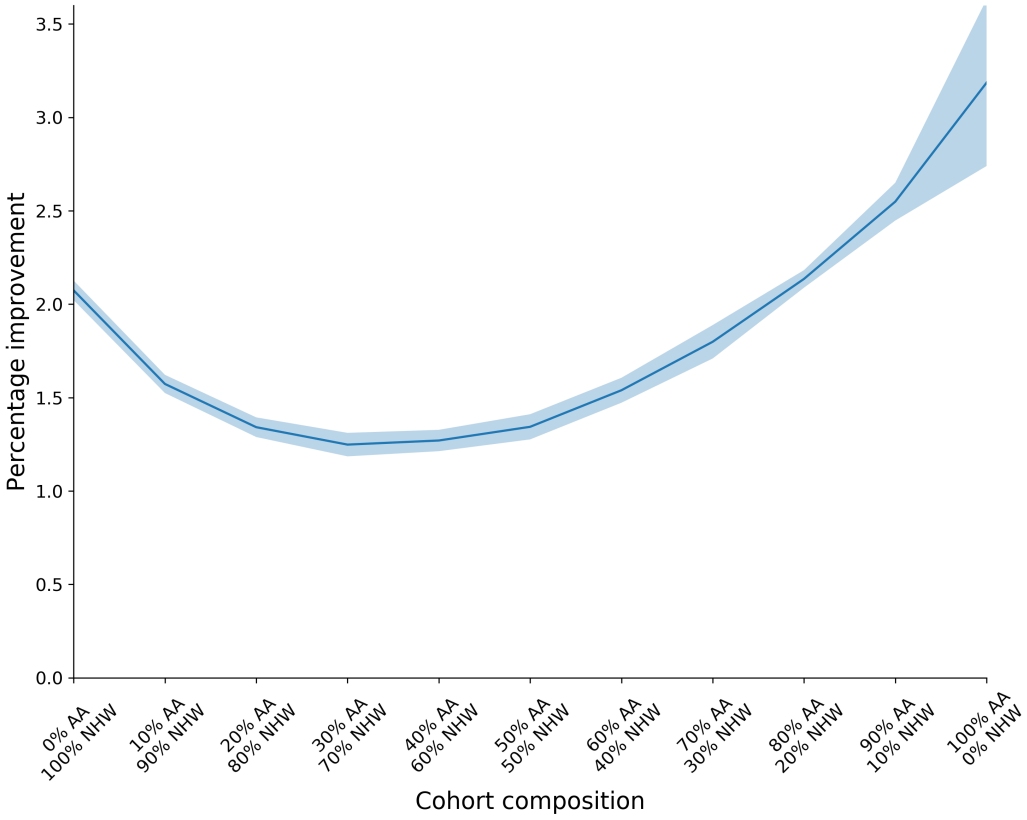


Figure 3.7: Relative improvement of ARDISS over IMPG-SUMMARY for different randomised mixed-ethnicity cohorts. ARDISS outperforms IMPG-SUMMARY in all mixture scenarios, with both methods being equally accurate in cases of very heterogeneous cohorts (with practically 50% of AA and NHW). The shaded area represents the standard deviation interval across the 10-fold validation.

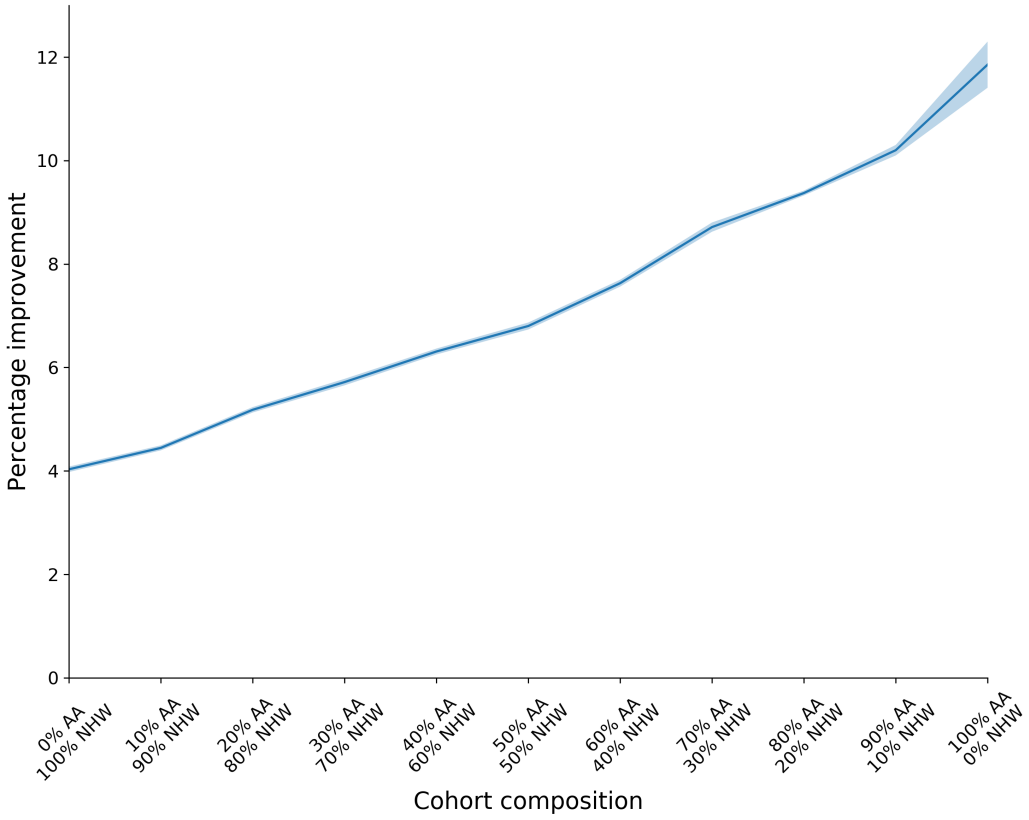


Figure 3.8: Relative improvement of ARDISS over DISTMIX for different randomised mixed-ethnicity cohorts. ARDISS outperforms DISTMIX in all mixture scenarios across the whole genome. The shaded area represents the standard deviation interval across the 10-fold validation.

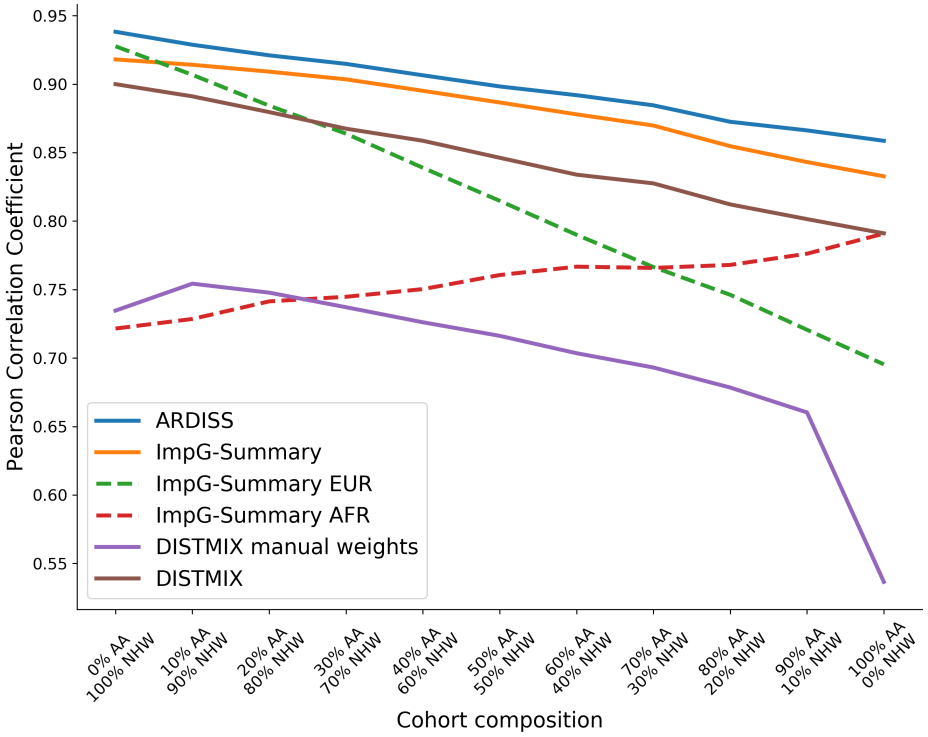


Figure 3.9: Pearson's correlation coefficients obtained during imputation across different mixtures of ethnicity sets using ARDISS and other available methods on chromosome 12. IMPG-SUMMARY was run using all the samples in the reference panel, with only the European samples (ImpG-Summary-EUR) and with only the African samples (ImpG-Summary-AFR). DISTMIX computed the optimal weights from the allele frequencies and was run with manual weight setting (for which we provided it with the original fractions of ASW and CEU).

some 12. Moreover, we evaluate the performance of DISTMIX when provided with "best-guess" weights, an approach that is realistic in a setting for which no information about the original population is known. For each ethnicity mixture, we attribute the effective percentage of weights to the ASW (Americans of African Ancestry in Southwest USA) and to the CEU (Utah Residents (CEPH) with Northern and Western European Ancestry). None of these approaches gives better results than the one mentioned above, with the "best-guess" weights approach yielding the worst performance of all. Results of these attempts are reported in Figure 3.9.

Additionally, leveraging the created dataset, we want to assess two other aspects of our proposed method: (i) the influence of the imputation window size and (ii) the impact of not accounting for covariates in the *original* study. We measure imputation performance for various window sizes (see Figure 3.10) and notice that performance initially improves with increasing size but starts deteriorating for larger windows. The cause of this behaviour is to be found in the non-overlapping nature of the

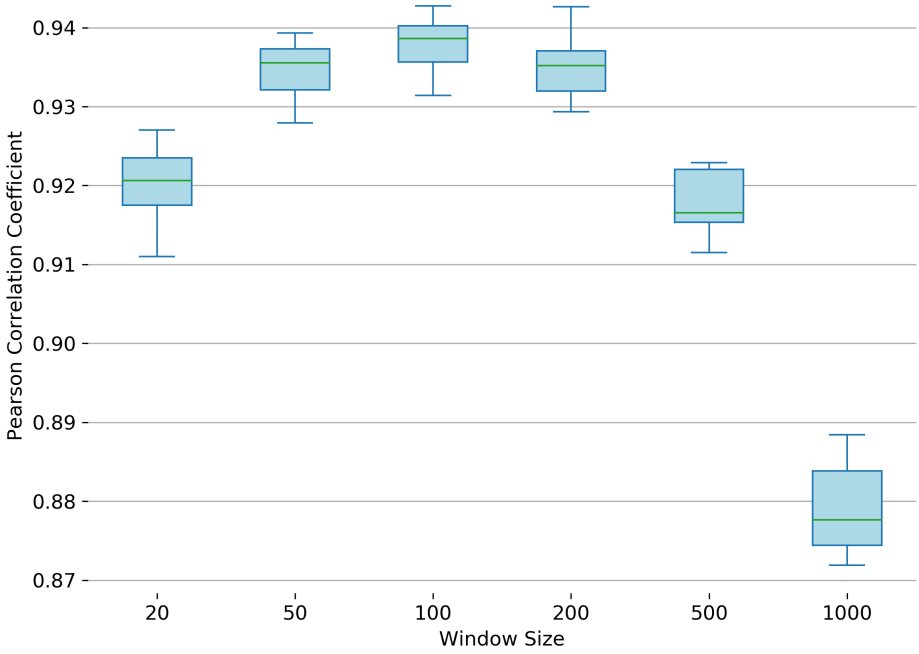


Figure 3.10: Pearson's correlation coefficients obtained during ARDISS imputation on chromosome 12 of the 0%AA | 100%NHW admixed cohort for different window sizes. The performance initially increases for larger windows but deteriorates for very large values. The reason is that larger windows lead to less successful automatic relevance determination, since large windows encompass more LD regions and dilute the signal. The imputation step is also affected as larger windows negatively impact imputation by adding considerable noise from low-LD SNPs.

ARD step: having larger windows implies overlapping more LD regions, making it harder to precisely pinpoint the composition of the original study population by giving more evenly distributed weights. Moreover, during imputation, long-distance, low-LD SNPs encompassed by larger window sizes add noise to the Gaussian Process, decreasing the quality of the imputation. For the dataset at hand, we observe optimal performance with window size of 100 SNPs. This number could be different in other studies with denser or sparser number of typed SNPs.

Furthermore, to assess the impact of not accounting for covariates on highlighting potential spurious associations, we analyse the percentage of recovered top hits. While correlation between masked and imputed Z-scores is a good global measure of imputation performance, it overlooks the ranks of the recovered Z-scores. In practice, if the original Z-score of a SNP ranks high in the study compared to the rest, the imputed value for that SNP should also rank high. We therefore select the top 100 SNPs from the *untyped* SNPs, i.e., the highest absolute value of the Z-scores marked

Table 3.3: Example percentage of recovered top 100 SNPs after imputation with Ardiss on chromosome 12 for the 10% AA | 90% NHW cohort.

	Typed	Ardiss	ImpG-Summary
With covariates	100	70	55
Without covariates	100	64	61

as missing, and compare them with the top 100 imputed Z-scores. Table 3.3 shows how Ardiss recovers a comparable number of top hits when the *original* association test is conducted with or without accounting for covariates. In the case of COPD, the covariates used as confounders were (i) age and (ii) pack-years of smoking. The results highlight the importance of accounting for covariates in GWAS. Nevertheless, it is safe to assume that most reported and published GWAS association results are obtained with the correct pipelines.

3.3.4 Insomnia complaints

The insomnia study provides a very realistic scenario for the imputation of association summary statistics: publicly available data of a study for which we have little to no previous knowledge of the evaluated population. We only compare Ardiss to ImpG-Summary due to its wider adoption and ease of use. As highlighted in Section 3.3.1, the study on insomnia complaints conducted by Hammerschlag and colleagues relies on samples from the UK Biobank, a large data set of self-reported traits and genotypes from the United Kingdom. The study participants reported their ethnicities themselves, making them potentially uncertain and an ideal scenario for our method. Table 3.4 summarises the results of the comparison. Ardiss clearly outperforms the comparison partner, suggesting that it successfully evaluates the study population's structure and ideally imputes values for the study at hand. This result highlights the benefits of using an adaptive method such as Ardiss in a setting where the ethnic background of the study participants is not clearly defined.

As mentioned in Section 3.1, Ardiss can straightforwardly be extended to perform imputation of β values. A β value is the regression coefficient obtained during the association analysis performed between a genetic variant and a phenotype. For this GWAS, the Pearson's correlation coefficient between the imputed and masked values for β values on chromosome 12 is 0.804 ± 0.008. The imputation accuracy is lower

Table 3.4: Imputation performance of Ardiss and ImpG-Summary on the publicly available summary statistics for the Insomnia Complaints GWAS.

Method	Insomnia	
	Correlation	RMSE
Ardiss	**0.956 ± 0.001**	**0.093 ± 0.002**
ImpG-Summary	0.889 ± 0.003	0.218 ± 0.005

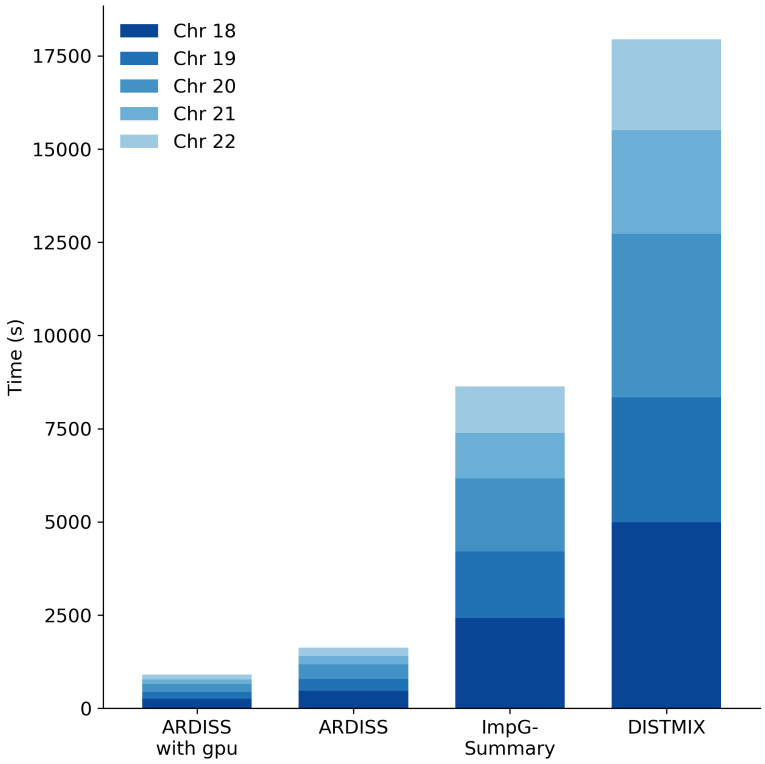


Figure 3.11: Breakdown of the run times for sequential imputation of summary statistics across chromosomes 18 to 22.

than for Z-scores because the Z-score—defined as the ratio of β over its standard error—contains more information about the association between the genetic variant and the phenotype.

3.3.5 Speed performance

Ardiss can easily be deployed on GPU architectures to speed up the ARD computation and the imputation. When compared with existing imputation methods, Ardiss shows steep improvements in runtime performance. The total elapsed time to impute the full genome missing SNPs (i.e. $11,671,761$ SNPs) described in Section 3.3.1 using ImpG-Summary was of $\sim$ 22h ($79,205.53$ s) compared to $\sim$ 4h15 min ($15,287.61$ s) for Ardiss and $\sim$ 2h20 ($8,530.12$ s) when using Ardiss on a GPU. Alternatively, users with strict time constraints also have the option to omit ARD and get even faster imputation: $\sim$ 35 minutes ($2,118.95$ s) for the whole genome, at a cost of slightly less accurate imputation. On the contrary, DISTMIX was too slow for full sequential imputation and its speed could only be measured for a subset of chromosomes. In order to impute $1,131,674$ SNPs on chromosomes 18 to 22, DISTMIX took $\sim$ 5h ($17,947.63$ s), compared to the $\sim$ 15 minutes (907.50 s) of Ardiss

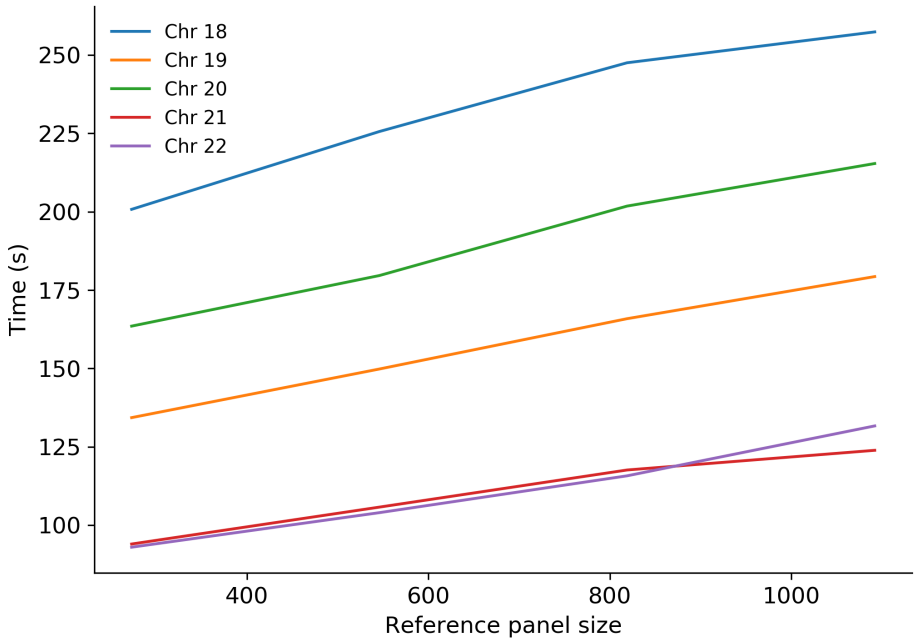


Figure 3.12: Scaling of ARDISS runtime with increasing number of samples in the reference panel. ARDISS scales linearly for an increasing number of samples as both the weight learning step and the imputation step only rely on inner products of genotypes for the computation of the covariance matrix. The experiments were run on our server, under the conditions specified in Section 3.3.2, for the 0% AA | 100% NHW study over chromosomes 18 to 22.

on a GPU. Figure 3.11 shows the sequential run times of ARDISS, IMPG-SUMMARY and DISTMIX on a subset of chromosomes. Since imputation methods are usually run in parallel to impute multiple chromosomes separately, we computed the mean ratio of run times of ARDISS and comparison partners across chromosomes: when using GPUs the user can expect, on average, a method that is 9.38 times faster than IMPG-SUMMARY and 19.88 times faster than DISTMIX. When dropping the ARD step, the fold change increases to 38.35 and 83.10 respectively.

To evaluate the theoretical runtimes highlighted in Section 3.2.4, we also measure the runtime of ARDISS for varying number of samples in the reference panel (see Figure 3.12) and for varying window size (see Figure 3.13).

3.3.6 Concluding remarks

In summary, we present ARDISS, a fast, accurate and adaptable method to impute missing Z-scores while inferring the underlying population composition without the need for any extra information such as allele frequencies or covariates of the original study population. The proposed method matches typical use-case scenarios better than any other available solution and outperforms them both in terms of performance

and runtime. ArdISS relies on open-source libraries and is publicly available on GitHub. The ever increasing body of publicly available results from association studies in plants, humans and other model organisms, enables researchers that use GWAS results to ask questions that go beyond the SNP-trait association. Integrating Z-scores from different studies makes the imputation of missing values a necessity which, coupled with the limited time researchers have to gather additional sample information from a study publication, creates opportunities for software tools that minimise the need for additional data. ArdISS is therefore a key tool to accelerate all these pipelines.

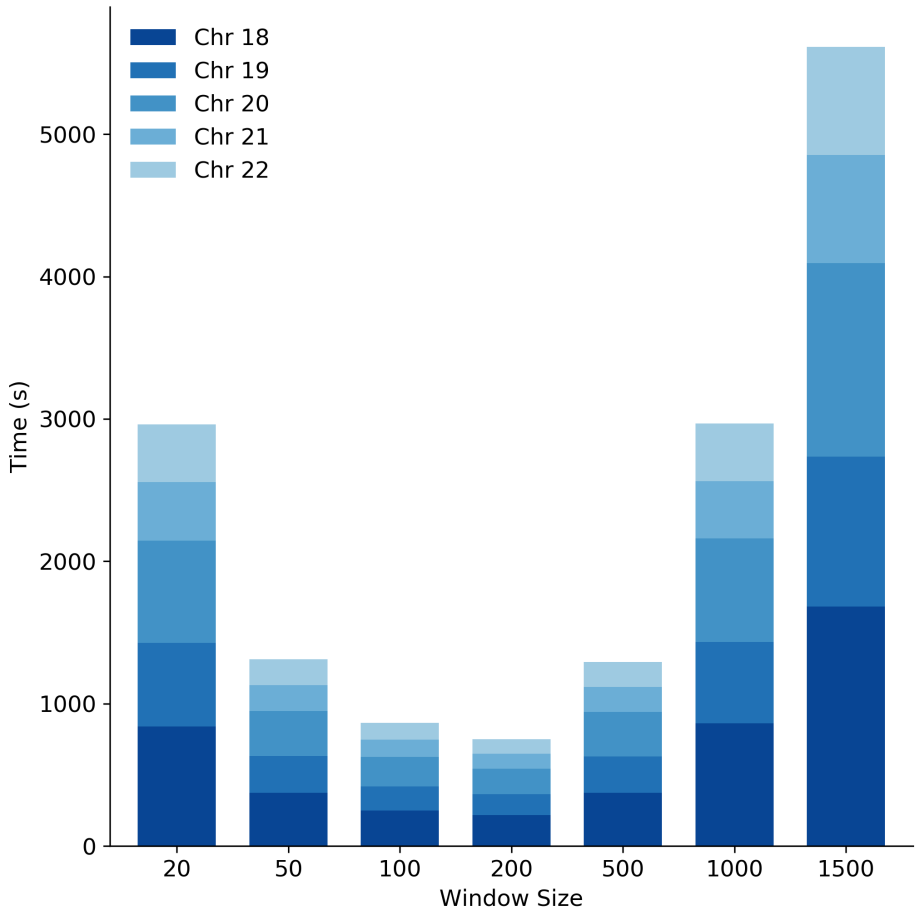


Figure 3.13: Breakdown of the runtime for imputation over chromosomes 18-22 using different window sizes. While the complexity of ArdISS is quadratic in the window size, very small window sizes also have a longer runtime in practice. This is due to the larger number of iterations the optimiser needs to converge for small window sizes. The experiments were run on our server, under the conditions specified in Section 3.3.2, for the 0% AA | 100% NHW study.

Table 3.5: Full results for the imputation performance of ARDISS, ImpG-Summary and DISTMIX for different ethnicity mixtures on Chromosome 12. AA: African American, NHW: Non-Hispanic White.

Ethnicity mixture	ARDISS			ImpG-Summary			DISTMIX		
	Correlation	R2 score	RMSE	Correlation	R2 score	RMSE	Correlation	R2 score	RMSE
0%AA 100%NHW	**0.937864**	**0.879574**	**0.120343**	0.918807	0.842510	0.157396	0.901536	0.810946	0.189162
10%AA 90%NHW	**0.928537**	**0.862112**	**0.138552**	0.914156	0.834141	0.166698	0.889022	0.786884	0.213162
20%AA 80%NHW	**0.920273**	**0.846737**	**0.154707**	0.908090	0.823282	0.178377	0.874940	0.760112	0.240772
30%AA 70%NHW	**0.912832**	**0.832950**	**0.168865**	0.901577	0.811679	0.190393	0.863474	0.738522	0.260892
40%AA 60%NHW	**0.905113**	**0.818732**	**0.183526**	0.893759	0.797866	0.204714	0.851413	0.715859	0.285431
50%AA 50%NHW	**0.897775**	**0.805242**	**0.197344**	0.885872	0.784054	0.218886	0.840613	0.695673	0.306743
60%AA 40%NHW	**0.889782**	**0.790610**	**0.211961**	0.876294	0.767376	0.235571	0.826718	0.670519	0.333172
70%AA 30%NHW	**0.881272**	**0.775119**	**0.227334**	0.865700	0.749122	0.253704	0.810649	0.641220	0.362157
80%AA 20%NHW	**0.872744**	**0.759577**	**0.243203**	0.854496	0.730003	0.273168	0.797976	0.617850	0.385391
90%AA 10%NHW	**0.864019**	**0.743779**	**0.259342**	0.842541	0.709811	0.293717	0.784062	0.593090	0.413472
100%AA 0%NHW	**0.854559**	**0.726644**	**0.274134**	0.828186	0.685850	0.315088	0.763999	0.558292	0.444259

Part III

Wasserstein kernels

4 Wasserstein Kernels for Structured Objects

In which optimal transport measures are employed to obtain fine-grained and sensitive kernels for structured data.

Structured objects have long been subjects of interest for machine learning. The high flexibility they offer make them the perfect fit to model real-life concepts and processes. The recent advances in deep learning methods have shown that higher performance can be reached with the most complex data (see Chapter 5 for a practical example). This, in turn, has revived interest in kernel-based methods for structured data. The classical $\mathcal{R}$-Convolution framework proposed by Haussler [101] allows for the easy construction of kernels for structured objects. It does so by comparing the substructures of the objects and aggregating the substructure similarities. However, the naïve application of the framework aggregates the substructures in a simple way, discarding valuable information about the *distribution* of the individual components. Moreover, it can be totally meaningless when applied to certain data structures, such as time series subsequences.

In this chapter, we propose a novel approach based on optimal transport theory that can capture subtler differences in data sets by simultaneously considering local and global characteristics of the structured object. Part of the presented content is based on the following publications:

- M. Togninalli, E. Ghisu, F. Llinares-López, B. Rieck, and K. Borgwardt. "Wasserstein Weisfeiler-Lehman Graph Kernels". In: *Advances in Neural Information Processing Systems*. 2019
- C. Bock, M. Togninalli, E. Ghisu, T. Gumbsch, B. Rieck, and K. Borgwardt. "A Wasserstein Subsequence Kernel for Time Series". In: *19th IEEE International Conference on Data Mining (ICDM 2019)*. 2019

The chapter is organised as follows. Section 4.1 gives the relevant background on $\mathcal{R}$-Convolution kernels and optimal transport. Section 4.2 extends the presented notions to graphs and guides the reader through the derivation of the Wasserstein Weisfeiler–Lehman graph kernel (WWL). Section 4.3 explores the application of the found concepts to time series and presents the Wasserstein Time series Kernel (WTK).

4.1 Introduction

Structured data can be found across all disciplines and considerable research efforts to develop novel machine learning models are devoted to tackle structured data problems. In the field of bioinformatics alone, structured data is present in multiple forms. For instance, graphs and trees have been used to model molecular structures [87] and evolutionary lineages for different organisms [33]. Strings are ubiquitous too: with the sheer amount of sequenced data collected in recent years, a myriad of models based on sequence modeling have emerged [3, 211]. Moreover, other data types can also be considered as structured data. Time series, for example, can be seen as sets of subsequences that are linked by their temporal ordering.

Developing machine learning algorithms for these data has therefore been a research goal for a long time. Tasks such as graph and time series classification generated a lot of interest from both theoretical and applied perspectives. For example, several methods have been proposed for small drug classification and property prediction, using graph regression and classification [215, 241]. Similarly, for biomedical time series, different classification and pattern mining [25] approaches were developed.

Kernel methods are particularly interesting due to the flexibility they offer for very diverse structures. This is especially the case for $\mathcal{R}$-Convolution kernels [101], which define similarity measures on *substructures* and aggregate them at the structure level (more details in Section 4.1.1). However, the vanilla application of $\mathcal{R}$-Convolution kernels to graphs and time series poses some critical challenges. For graphs, the aggregation step discards very valuable information about the distribution of substructures. This problem is better described in Section 4.2. For time series, the construction of kernels on substructures via the $\mathcal{R}$-Convolution framework often results in meaningless kernels, as elaborated in Section 4.3.

To tackle these issues, we turn to Optimal Transport (OT), a field of mathematics that has been increasingly popular in machine learning thanks to improvements of the computational strategies to efficiently obtain Wasserstein distances [5, 57]. In turn, these advancements have led to many applications ranging from generative models [8] to new loss functions [83]. Sections 4.2 and 4.3 illustrate how notions from OT can be used to improve kernels for graphs and time series. The remainder of this section presents the relevant background on the popular framework proposed by Haussler and introduces the optimal transport notions necessary to the development of our new class of kernels.

4.1.1 $\mathcal{R}$-Convolution kernels

Kernels are a class of similarity functions that offer interesting properties to be used in learning algorithms [208]. Let $\mathcal{X}$ be a set with n elements and $\mathrm{k}\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a

function. If k is (i) symmetric, and (ii) positive definite[1], i.e. $\sum_{i,j=1}^{n} c_i c_j \, \mathrm{k}(x_i, x_j) \geq 0$ for every $c_i \in \mathbb{R}$ and $x_i, x_j \in \mathcal{X}$, then k is said to be a *kernel on* $\mathcal{X} \times \mathcal{X}$.

Equivalently, there exists a *Hilbert space* $\mathcal{H}$ (a complete inner product space) and a map $\phi \colon \mathcal{X} \to \mathcal{H}$ such that $\mathrm{k}(\cdot, \cdot)$ can be equivalently expressed as

$$\mathrm{k}(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle_{\mathcal{H}}, \tag{4.1}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ indicate the inner product on $\mathcal{H}$. $\mathcal{H}$ is also referred to as a *Reproducing Kernel Hilbert Space* (RKHS) because its inner product *reproduces* the given kernel.

Thanks to this *feature space* view, a positive definite kernel can be interpreted as a dot product in a high-dimensional space. This in turn allows for their use in any learning algorithm that relies on dot products, such as support vector machines (SVMs), by virtue of the *kernel trick* [207].

Let us now consider the case of structured data. Let $x \in \mathcal{X}$ be a composite structure and $x_1, \ldots, x_D$ are its "parts", where x_d is in the set $\mathcal{X}_d$ for each $1 \leq d \leq D$. Assuming that $\mathcal{X}, \mathcal{X}_1, \ldots, \mathcal{X}_D$ are nonempty, separable metric spaces, we can define a kernel k_d on $\mathcal{X}_d$ for each $1 \leq d \leq D$. Next, suppose we have $x, y \in \mathcal{X}$ with their decompositions $\vec{x} = x_1, \ldots, x_D$ and $\vec{y} = y_1, \ldots, y_D$. We can use k_d to measure the similarity $\mathrm{k}_d(x_d, y_d)$ between the part x_d and the part y_d. We can then define the $\mathcal{R}$-*Convolution* of $\mathrm{k}_1, \ldots, \mathrm{k}_D$ denoted as $\mathrm{k}_1 \star \cdots \star \mathrm{k}_D(x, y)$ as the zero extension to $\mathcal{X} \times \mathcal{X}$ of

$$\mathrm{k}(x, y) = \sum_{\substack{x_1, \ldots, x_D \in \vec{x} \\ y_1, \ldots, y_D \in \vec{y}}} \prod_{d=1}^{D} \mathrm{k}_d(x_d, y_d) \tag{4.2}$$

Theorem 1 ($\mathcal{R}$-Convolution kernels [101])**.** *If* $\mathrm{k}_1, \ldots, \mathrm{k}_D$ *are kernels on* $\mathcal{X}_1 \times \mathcal{X}_1, \ldots, \mathcal{X}_D \times \mathcal{X}_D$*, then* $\mathrm{k}_1 \star \cdots \star \mathrm{k}_D(x, y)$ *is a kernel on* $\mathcal{X} \times \mathcal{X}$*.*

For a proof, see the seminal paper by Haussler [101]. Since, in practice, ensuring positive definiteness is not always feasible, many learning algorithms were recently proposed to extend SVMs to indefinite kernels [14, 148, 178]. Some proposed approaches are not directly based on an RKHS but rather on *Reproducing Kernel Kreĭn Spaces* (RKKS) [177]. In an RKKS, positive definiteness of the kernel function is left aside, so that kernels are allowed to be indefinite, i.e. neither positive definite nor negative definite. Moreover, previous research [96] showed that in practice, SVM classifiers can handle the integration of indefinite kernel matrices (see [140, 152, 262]) while still offering favourable predictive performance. This provides a solid foundation for developing new kernel methods, especially when considering that positive definiteness can impose restrictive conditions onto the underlying similarities induced by a kernel function [73].

4.1.2 Optimal transport

Transportation theory, from which optimal transport methods stem, is a field of mathematics aimed at comparing probability distributions by geometrical means.

[1] To simplify notation, we do not make a distinction between "positive definite" and "positive semi-definite" in this thesis.

One of its most commonly-used metrics is the *Wasserstein distance*. The Wasserstein distance is a distance function between probability distributions defined on a specific metric space. Let σ and μ be two probability distributions on a metric space M equipped with a ground distance d, such as the Euclidean distance.

Definition 2. *The L^p-Wasserstein distance for $p \in [1, \infty)$ is defined as*

$$W_p(\sigma, \mu) := \left(\inf_{\gamma \in \Gamma(\sigma,\mu)} \int_{M \times M} d(x, y)^p \, \mathrm{d}\gamma(x, y) \right)^{\frac{1}{p}}, \tag{4.3}$$

where $\Gamma(\sigma, \mu)$ is the set containing all transportation plans $\gamma \in \Gamma(\sigma, \mu)$ over $M \times M$ with marginals σ and μ on the first and second factors, respectively.

The Wasserstein distance is a metric and satisfies the required axioms (namely non-negativity, identity of indiscernibles, symmetry, and triangle inequality), provided that d is a metric (for a proof, see the book of Villani [240], chapter 6). For the scope of this thesis, we focus on the distance for $p = 1$ and, when mentioning the Wasserstein distance, we refer to the L^1-Wasserstein distance unless noted otherwise.

The Wasserstein distance is related to the optimal transport problem [240], for which one wishes to find the most "inexpensive" (in terms of the predefined ground distance) way to transport all the probability mass from distribution σ so as to match distribution μ. A more intuitive example can be made by considering the 1-dimensional case, where the two distributions can be seen as piles of dirt or sand. In this context, the Wasserstein distance is also referred to as the *Earth Mover's Distance* [204] and its solution represents the minimum effort required to move the content of the first pile of dirt to reproduce the second one.

While Definition 2 is correct, it is not very practical to deal with *finite sets*, as we will do in the remainder of the Chapter. In fact, as we saw in Section 4.1.1, when using kernel methods on structured objects, we deal with sets of parts (also referred to as substructures). Therefore, we can reformulate the Wasserstein distance as a sum rather than an integral and rely on the matrix formulation of the optimisation problem presented above. This definition is the one commonly encountered in the optimal transport literature [204] and fits our setting better.

Definition 3. *Let $X \in \mathbb{R}^{n \times m}$ and $Y \in \mathbb{R}^{n' \times m}$ be two matrices. We consider X and Y to represent sets of feature vectors of dimension m, but of varying cardinalities n and n'. The 1^{st} Wasserstein distance between X and Y is defined as*

$$\mathrm{W}_1(X, Y) := \min_{P \in \Gamma(X,Y)} \langle D, P \rangle_{\mathrm{F}}, \tag{4.4}$$

where D is an $n \times n'$ matrix containing the pairwise distances $\mathrm{dist}(x, y)$ for $(x, y) \in X \times Y$, P is the transport matrix, and $\langle \cdot, \cdot \rangle_{\mathrm{F}}$ is the Frobenius inner product.

The transport matrix P (or joint probability) contains the fractions indicating the way to transport values from X to Y with the lowest total transport effort.

Assuming that the total mass to transport equals 1 and that it is evenly distributed in the elements of X and Y, the values for the rows and columns of P must sum to $1/n$ and $1/n'$, respectively, and the sum of all the entries of P must therefore be equal to 1.

4.2 Wasserstein Graph Kernels

Many problems in chemo- and bioinformatics can be modelled using graphs. From protein-protein interaction networks interrogation [170] to molecular properties prediction [254], graph-based tasks have become ubiquitous. Graph kernels [241] have been highly successful in tackling graph-specific problems thanks to their modeling flexibility. In particular, graph kernels have shown excellent predictive performance for various classification problems [173, 215, 258].

As briefly mentioned in Section 4.1, most graph kernels rely on the $\mathcal{R}$-Convolution framework. In practice, existing kernels decompose the graphs in subgraphs, compute local similarities, and aggregate them at the global level. However, $\mathcal{R}$-Convolution kernels on graphs have known limitations: (i) the simplicity of the local similarities aggregation procedure can hinder their ability to capture complex characteristics of the graph; (ii) most proposed approaches do not generalise to graphs with high-dimensional continuous node attributes, and extensions are not straightforward. Some techniques have been suggested to address point (i). For instance, Fröhlich et al. [84] introduced kernels based on the optimal assignment of node labels, although the obtained kernels are not positive definite [239]. Lately, another method was proposed by Kriege et al. [131]: it leverages a Weisfeiler–Lehman based colour refinement scheme and then solves an optimal assignment problem to compute the kernel. Unfortunately, this method cannot handle continuous node attributes, leaving point (ii) as an unsolved problem.

To overcome the aforementioned limitations, we developed a method that combines the most distinctive vectorial representations obtained from the graph kernel literature with methods from optimal transport, which have recently gained considerable attention in machine learning in general and in graph applications in particular [255].

In this section, we present the Wasserstein Weisfeiler–Lehman (WWL) graph kernel. We provide the theoretical foundations of our method and showcase successful experimental results on categorical and continuously attributed graphs.

4.2.1 Graph kernels

This Section introduces the necessary notation and background for graph kernels. As introduced in Section 4.1.1, kernels are a class of similarity functions that have interesting properties for learning algorithms. The $\mathcal{R}$-Convolution framework is usually used to define kernels on graphs. Before detailing the way of constructing graph kernels, let us introduce some notation.

We define a graph as a tuple $G = (V, E)$, where V and E denote the set of nodes and edges, respectively. In our case, we further assume that the edges are undirected. Moreover, we denote the cardinality of nodes and edges for G as $|V| = n_G$ and $|E| = m_G$. For a node $v \in V$, we write $\mathcal{N}(v) = \{u \in V \mid (v, u) \in E\}$ and $|\mathcal{N}(v)| = \deg(v)$ to indicate its first-order neighbourhood, i.e. the set of connected nodes, and the "degree" of the node.

We say that a graph is *labelled* if its nodes have categorical labels. A label on the nodes is a function $l\colon V \to \Sigma$ that assigns to each node v in G a value $l(v)$ from a finite label alphabet Σ. Additionally, we say that a graph is *attributed* if for each node $v \in V$ there exists an associated vector $\boldsymbol{a}(v) \in \mathbb{R}^m$. For the scope of this thesis, $\boldsymbol{a}(v)$ are the node attributes and $l(v)$ are the categorical node labels of node v. In particular, the node attributes are high-dimensional continuous vectors, whereas the categorical node labels are assumed to be integer numbers (encoding either a category or an ordered discrete value). With the term "node labels", we implicitly refer to categorical node labels. Finally, a graph can have weighted edges, and the function $w\colon E \to \mathbb{R}$ defines the weight $w(e)$ of an edge $e := (v, u) \in E$.

The main idea of $\mathcal{R}$-Convolution kernels on graphs is to decompose graph G into substructures and to define a kernel value $k(G, G')$ as a combination of substructure similarities. The first kernel on graphs was introduced by Kashima et al. [115], where node and edge attributes are used to generate label sequences based on a random walk scheme. Building on this, a more efficient approach based on shortest paths [29] was proposed: it computes each kernel value $k(G, G')$ as a sum of the similarities between each shortest path in G and each shortest path in G'. However, despite the large practical success of $\mathcal{R}$-Convolution kernels, they usually rely on on aggregation strategies that ignore valuable information such as the distribution of the substructures. This is the case in the Weisfeiler–Lehman (WL) subtree kernel or one of its variants [199, 214, 215], which obtain graph-level features by simply *summing* the contribution of the node representations. To avoid these simplifications, we use concepts from optimal transport theory (see Section 4.1.2), which can help to better capture the similarities between graphs.

4.2.2 Wasserstein distance on graphs

The insufficient discriminative ability of current $\mathcal{R}$-Convolution graph kernels caused by their simple aggregation step that might mask important substructure differences by averaging, prompted us to have a finer distance measure between structures and their components. Our method is composed of the following steps:

(i) Obtain a set of node embeddings (i.e. vectorial representations) for each graph,

(ii) Measure the Wasserstein distance between each pair of graphs,

(iii) Compute a similarity measure to be used in a learning algorithm.

A schematic view of the first two steps can be found in Figure 4.1 and Algorithm 3 details the complete method. We now introduce the procedure to obtain a set of embeddings and show how to integrate them in the Wasserstein distance.

Definition 4 (Graph Embedding Scheme)**.** *Given a graph $G = (V, E)$, a graph embedding scheme $f\colon \mathcal{G} \to \mathbb{R}^{|V|\times m}$, $f(G) = X_G$ is a function that outputs a fixed-size vectorial representation for each node in the graph. For each $v_i \in V$, the i-th row of X_G is called the node embedding of v_i.*

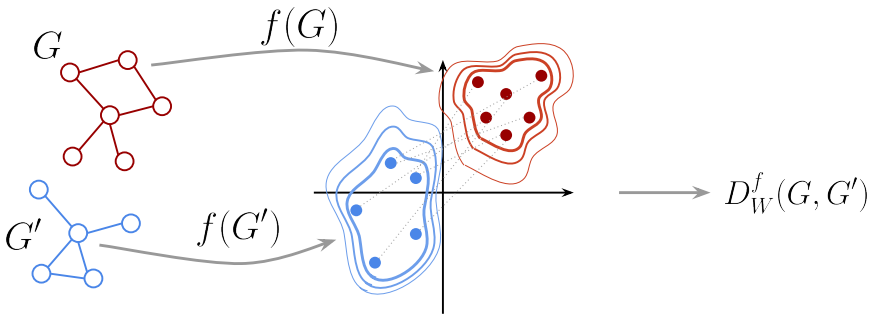


Figure 4.1: Schematic summary of the graph Wasserstein distance. First, f generates node embeddings for two input graphs G and G'. Then, the Wasserstein distance between the embedding distributions is computed.

Definition 4 permits the use of node labels, which are categorical attributes, as one-dimensional attributes with $m = 1$.

Definition 5 (Graph Wasserstein Distance)**.** *Given two graphs $G = (V, E)$ and $G' = (V', E')$, a graph embedding scheme $f\colon \mathcal{G} \to \mathbb{R}^{|V|\times m}$ and a ground distance $d\colon \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$, we define the Graph Wasserstein Distance (GWD) as*

$$D_W^f(G, G') := \mathrm{W}_1(f(G), f(G')). \tag{4.5}$$

where W_1 is the Wasserstein distance as introduced in Definition 3. Equipped with these definitions, we will now propose a graph embedding scheme inspired from the Weisfeiler–Lehman (WL) kernel, extend it to continuously attributed graphs with weighted edges, and show how to use it to obtain fine-grained GWDs.

Obtaining the graph embedding

The Weisfeiler–Lehman scheme. The Weisfeiler–Lehman subtree kernel [214, 215] was developed for labelled non-attributed graphs. It considers similarities among subtree patterns defined by a propagation scheme that iteratively compares the labels of nodes and their neighbours. At every iteration, the labels of the neighbouring nodes are put in an ordered string sequence and appended to the original label. The set of obtained strings is subsequently *hashed* to create updated *compressed* node labels that are in turn attributed to the nodes of the graph. With every iteration of the algorithm, these hashed labels represent increasingly larger neighbourhoods of each node, enabling the comparison of extended substructures.

In particular, consider a graph $G = (V, E)$, let $\ell^0(v) = \ell(v)$ be the initial node label of v for each $v \in V$, and let H be the number of iterations of the WL scheme. Then, we can define a recursive scheme to compute $\ell^h(v)$ for $h = 1, \ldots, H$ by looking at the ordered set of neighbours labels $\mathcal{N}^h(v) = \{\ell^h(u_0), \ldots, \ell^h(u_{\deg(v)-1})\}$ as

$$\ell^{h+1}(v) = \mathrm{hash}(\ell^h(v), \mathcal{N}^h(v)). \tag{4.6}$$

We denote this procedure as the WL labelling scheme. We use perfect hashing for the hash function, so that nodes at iteration $h+1$ will have the same label if and only if their label and those of their neighbours are the same at the previous iteration h.

Extension to continuous attributes. For graphs with continuous attributes $\boldsymbol{a}(v) \in \mathbb{R}^m$, we need to adapt the WL refinement step, which was not originally defined for continuous cases. Existing approaches have been implicitly investigated for node-level kernel similarity computations [173, 175], but they rely on extra hashing steps for the continuous features. The goal is to create an explicit propagation scheme that considers the node features of the entire neighbourhood and combines them for the current node of interest. It is then easy to incorporate edge weights in the average calculation of each neighbourhood. Suppose we have a continuous attribute $\boldsymbol{a}^0(v) = \boldsymbol{a}(v)$ for each node $v \in G$. Then, we recursively define

$$\boldsymbol{a}^{h+1}(v) = \frac{1}{2}\left(\boldsymbol{a}^h(v) + \frac{1}{\deg(v)} \sum_{u \in \mathcal{N}(v)} w((v,u)) \cdot \boldsymbol{a}^h(u)\right). \tag{4.7}$$

When there are no edge weights, we set $w(u,v) = 1$. We use a weighted average over the neighbourhood attributes instead of a sum. Additionally, we add the $1/2$ to ensure a similar scale of the features across iterations. As we will discuss later, we concatenate these features for building our kernel (see Definition 6) and obtain better empirical results with similarly scaled features. Despite the fact that this does not constitute a proper test of isomorphism, this refinement scheme can be seen as an intuitive extension for continuous attributes of the one used on categorical node labels by the WL subtree kernel, which has proven to be highly successful. Besides, one can see the parallel with the propagation scheme used in several graph neural networks, which have also shown brilliant performance for node classification tasks on large graphs [67, 124, 126].

Graph embedding scheme. Leveraging the recursive procedure described above, we propose a WL-based graph embedding scheme to generate node embeddings capturing both the node labels or attributes and the topology of the graph.

Definition 6 (WL features)**.** *Let $G = (V, E)$ and let H be the total number of WL iterations. Then, for every $h \in \{0, \dots, H\}$, we define the WL features as*

$$X_G^h = [\boldsymbol{x}^h(v_1), \dots, \boldsymbol{x}^h(v_{n_G})]^T, \tag{4.8}$$

where $\boldsymbol{x}^h(\cdot) = \ell^h(\cdot)$ for categorically labelled graphs and $\boldsymbol{x}^h(\cdot) = \boldsymbol{a}^h(\cdot)$ for continuously attributed graphs. We denote $X_G^h \in \mathbb{R}^{n_G \times m}$ as the node features *of graph G at iteration h. Then, the node embeddings of graph G at iteration H are defined as*

$$\begin{aligned} f^H \colon G &\to \mathbb{R}^{n_G \times (m(H+1))} \\ G &\mapsto \text{concatenate}(X_G^0, \dots, X_G^H). \end{aligned} \tag{4.9}$$

Above, m denotes the dimensionality of the node attributes, and $m = 1$ for the categorical labels. We note that a graph can be both *categorically labelled* and *continuously attributed*, it would therefore be possible to extend the above scheme by jointly considering this information (for instance, by concatenating the node features). However, this would require to define an appropriate distance measure between categorical and continuous data, a long-standing issue that will not be solved in this thesis [221].

Computing the Wasserstein distance

With a defined Graph Embedding Scheme, we can then evaluate the pairwise Wasserstein distance between graphs using their node embeddings. To do so, we start by defining and computing the ground distances between each pair of nodes. For categorical node features, we rely on the normalised Hamming distance:

$$d_{\text{Ham}}(\boldsymbol{v}, \boldsymbol{v'}) = \frac{1}{H+1} \sum_{i=1}^{H+1} \rho(v_i, v'_i), \quad \rho(x, y) = \begin{cases} 1, & x \neq y \\ 0, & x = y \end{cases} \tag{4.10}$$

The Hamming distance is equal to 1 when two vectors share no features and 0 when they are identical and can be imagined as the normalised sum of the discrete metric ρ on each of the features. The choice of the Hamming distance is motivated by the categorical nature of the Weisfeiler–Lehman features for nodes with categorical labels, whose value carry no meaning. For continuous node features, on the other hand, we use the Euclidean distance:

$$d_E(\boldsymbol{v}, \boldsymbol{v'}) = ||\boldsymbol{v} - \boldsymbol{v'}||_2. \tag{4.11}$$

We then insert the ground distance in the equation of Definition 2 and compute the Wasserstein distance via the network simplex method [186].

Computational complexity. The naïve computation of the Wasserstein distance has a complexity of $\mathcal{O}(n^3 log(n))$, where n is the number of nodes in the two graphs. However, recent advances in the optimal transport field have reduced the practical runtime. In particular, Sinkhorn regularisation [57] enables approximations that reduce the computational burden to *near-linear time* while preserving the original distance [5]. Such speedups become particularly useful for larger data sets, i.e. graphs with thousands of nodes and can be seamelessly integrated in our method. A detailed discussion of the runtime performance is reported in Section 4.2.4.

4.2.3 From distance to graph kernels: theoretical considerations

Once the Wassestein distance between two graphs is obtained, it is possible to construct a similarity measure to accommodate various learning algorithms. We therefore propose a new graph kernel, present some considerations about its theoretical properties, and show how to use it for graph classification tasks.

Algorithm 3 Compute Wasserstein graph kernel

Input: Two graphs G_1, G_2; graph embedding scheme f^H; ground distance d; λ.
Output: kernel value $k_{WWL}(G_1, G_2)$.

$X_{G_1} \leftarrow f^H(G_1)$ // Generate node embeddings for G_1
$X_{G_2} \leftarrow f^H(G_2)$ // Generate node embeddings for G_2
$D \leftarrow \text{pairwise\_dist}(X_{G_1}, X_{G_2}, d)$ // Compute the ground distance between each pair of nodes
$D_W(G_1, G_2) = \min_{P \in \Gamma} \langle P, D \rangle$ // Compute the Wasserstein distance
$k_W(G_1, G_2) \leftarrow e^{-\lambda D_W(G_1, G_2)}$ // Apply the Laplacian kernel

Definition 7 (Wasserstein Weisfeiler–Lehman)**.** *Given a set of graphs $\mathcal{G} = \{G_1, \ldots, G_N\}$, $\lambda \in \mathbb{R}_{>0}$, and the GWD defined for each pair of graph on their WL embeddings, we define the Wasserstein Weisfeiler–Lehman (WWL) kernel as*

$$K_{\text{WWL}} = e^{-\lambda D_W^{f_{\text{WL}}}}. \tag{4.12}$$

The proposed kernel is an instance of a Laplacian kernel, which offers favourable conditions for positive definiteness for non-Euclidean distances [73]. We will now detail a few theoretical considerations for the obtained kernel.

For Euclidean spaces, obtaining positive definite kernels from distance functions is a well-studied topic [97]. However, obtaining positive definite kernels from optimal transport distances, which are not isometric in their general form [75], remains an open research question. Several attempts to draw general conclusions on the definiteness of the Wasserstein distance were unsuccessful, but insightful results on particular cases were obtained along the way. Here, we first collect some of these contributions and use them to prove that our WWL kernel for categorical embeddings is positive definite. Then, we elaborate further on the case of continuous embeddings, for which we provide conjectures on practical conditions to obtain a positive definite kernel.

Theoretical considerations for categorical embeddings

Before proceeding, it is helpful to recall the notion of positive definite kernel introduced in Section 4.1.1.

Definition 8 (Schölkopf and Smola [208])**.** *A symmetric function $k \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a* positive definite (pd) kernel *if it satisfies the condition*

$$\sum_{i,j=1}^{n} c_i c_j K_{ij} \geq 0, \text{ with } K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{4.13}$$

for every $c_i \in \mathbb{R}$, $n \in \mathbb{N}$ and $\boldsymbol{x}_i \in \mathcal{X}$.

The matrix of kernel values K with entries K_{ij} is denoted *Gram matrix* of k with respect to $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. A *conditional* positive definite (cpd) kernel is a function that satisfies Equation 4.13 for all $c_i \in \mathbb{R}$ with $\sum_{i=1}^n c_i = 0$. Analogously, a conditional negative definite (cnd) kernel is a function that satisfies $\sum_{i,j=1}^n c_i c_j K_{ij} \leq 0$ for all $c_i \in \mathbb{R}$ with $\sum_{i=1}^n c_i = 0$.

As mentioned above, obtaining kernels from distance functions is a well-studied topic for Euclidean spaces.

Proposition 1 (Haasdonk and Bahlmann [97])**.** *Let $d(\boldsymbol{x}, \boldsymbol{x'})$ be a symmetric, non-negative distance function with $d(\boldsymbol{x}, \boldsymbol{x}) = 0$. If d is isometric to an L^2-norm, then*

$$k_d^{\mathrm{nd}}(\boldsymbol{x}, \boldsymbol{x'}) = -d(\boldsymbol{x}, \boldsymbol{x'})^\beta, \quad \beta \in [0, 2] \tag{4.14}$$

is a valid cpd kernel.

Nevertheless, the Wasserstein distance is not isometric in its general form, meaning that there is no metric-preserving mapping to an L^2-norm. This is because the metric space it induces strongly depends on the used ground distance [75].

Probability distributions are not the only type of data that do not always reside in Euclidean spaces. Hence, Feragen et al. [73] defined the family of exponential kernels relying on a non-Euclidean distance d as:

$$k(\boldsymbol{x}, \boldsymbol{x'}) = e^{-\lambda d(\boldsymbol{x}, \boldsymbol{x'})^q} \quad \text{for} \quad \lambda, q > 0, \tag{4.15}$$

and denote them as geodesic kernels. Using earlier considerations from Berg et al. [21], they also showed that, under certain conditions, the Laplacian kernel ($q = 1$ in Equation 4.15) is positive definite.

Proposition 2 (Feragen et al. [73])**.** *The geodesic Laplacian kernel is positive definite for all $\lambda > 0$ if and only if the geodesic distance d is conditional negative definite.*

Here too, considerations on the negative definiteness of Wasserstein distance functions cannot be made on a general level. Nonetheless, certain ground distances *guarantee* the negative definite nature of the resulting Wasserstein distance. Particularly, the Wasserstein distance equipped with the discrete metric ρ, already encountered in Equation 4.10, was proved to be conditional negative definite [86].

These conclusions can be leveraged to prove that the Wasserstein distance with a Hamming ground distance is conditional negative definite under specific conditions, yielding, therefore, a positive definite kernel for the categorical WL embeddings.

Theorem 2. *The categorical WWL kernel is positive definite for all $\lambda > 0$.*

The Weisfeiler–Lehman labelling scheme relies on a shared dictionary across the entire graph to generate the node embeddings. This, in turn, can be used to show that the solutions of the optimal transport problem are shared across iterations. We refer to the Weisfeiler–Lehman embedding scheme as defined in Definition 6 as f_{WL}^H, and let $D_W^{f_{\mathrm{WL}}}$ be the corresponding GWD on a set of graphs $\mathcal{G}$ with categorical labels.

Additionally, let $d_{\text{Ham}}(\boldsymbol{v}, \boldsymbol{v'})$ of Equation 4.10 be the ground distance of $D_W^{f_{\text{WL}}}$. The following useful results hold.

Lemma 1. *If a transportation plan γ with transport matrix P is optimal in the sense of Definition 3 for distances d_{Ham} between embeddings obtained with f_{WL}^H, then it is also optimal for the discrete distances d_{disc} between the H-th iteration values obtained with the Weisfeiler–Lehman procedure.*

Proof. We recall the matrix notation introduced in Equation 4.4, where M is the cost or distance matrix, $P \in \Gamma$ is the transport matrix, and $\langle \cdot, \cdot \rangle$ is the Frobenius dot product. Since we give equal weight (i.e., equal probability mass) to each of the vectors in each set, Γ contains all nonnegative $n \times n'$ matrices P with

$$\sum_{i=1}^{n} p_{ij} = \frac{1}{n'} \;\; , \;\; \sum_{j=1}^{n'} p_{ij} = \frac{1}{n} \;\; , \;\; p_{ij} \geq 0 \;\; \forall i, j \tag{4.16}$$

To simplify notation, we denote the Hamming distance matrix $D_{\text{Ham}}(f_{\text{WL}}^h(G), f_{\text{WL}}^h(G'))$, where the ij-th entry is given by the Hamming distance between the embedding of the i-th node of graph G and the embedding of the j-th node of graph G' at iteration h, by D_{Ham}^h. Similarly, we define D_{disc}^h to be the discrete metric distance matrix, where the ij-th entry is given by the discrete distance between feature h of node embedding i of graph G and feature h of node embedding j of graph G'. We note that $[D_{\text{Ham}}^h]_{ij} \in [0, 1]$ and $[D_{\text{disc}}^h]_{ij} \in \{0, 1\}$ and that, by definition (see Equation 4.10):

$$D_{\text{Ham}}^H = \frac{1}{H} \sum_{h=0}^{H} D_{\text{disc}}^h. \tag{4.17}$$

Additionally, by the formulation of the WL procedure, two labels that are different at iteration h will also differ at iteration $h + 1$. Therefore, the following identify holds:

$$\left[D_{\text{Ham}}^h\right]_{ij} \leq \left[D_{\text{disc}}^h\right]_{ij}, \tag{4.18}$$

which, in turn, implies that $[D_{\text{Ham}}^h]_{ij} = 0 \iff [D_{\text{disc}}^h]_{ij} = 0$.

An optimal transportation plan P^h for f_{WL}^h embeddings satisfies

$$\left\langle P^h, D_{\text{Ham}}^h \right\rangle \leq \left\langle P, D_{\text{Ham}}^h \right\rangle \; \forall P \in \Gamma. \tag{4.19}$$

If we assume that P^h is not optimal for D_d^h, we can define P^* such that

$$\left\langle P^*, D_{\text{disc}}^h \right\rangle < \left\langle P^h, D_{\text{disc}}^h \right\rangle. \tag{4.20}$$

Because the entries of D^h_{disc} are either 0 or 1, we can define the set of indices tuples $\mathcal{H} = \left\{(i,j) \mid [D^h_{\text{disc}}]_{ij} = 1\right\}$ and rewrite the inequality as

$$\sum_{i,j\in\mathcal{H}} p^*_{ij} < \sum_{i,j\in\mathcal{H}} p^h_{ij}. \tag{4.21}$$

We consider the constraints on the entries of P^* and P^h, namely $\sum_{i,j} p^*_{ij} = \sum_{i,j} p^h_{ij} = 1$, this implies that, by rearranging the transport map, there is more mass that could be transported at 0 cost. In our formalism,

$$\sum_{i,j\notin\mathcal{H}} p^*_{ij} > \sum_{i,j\notin\mathcal{H}} p^h_{ij}. \tag{4.22}$$

However, as stated before, entries of D^h_d that are 0 are also 0 in D^h_{Ham}. Therefore, a better transport plan P^* would also be optimal for D^h_{Ham}:

$$\left\langle P^*, D^h_{\text{Ham}} \right\rangle < \left\langle P^h, D^h_{\text{Ham}} \right\rangle, \tag{4.23}$$

which contradicts the optimality assumption above. Hence, P^h is also optimal for D^H_{disc}. □

Lemma 2. *If a transportation plan γ with transport matrix P is optimal in the sense of Definition 3 for distances d_{Ham} between embeddings obtained with f^H_{WL}, then it is also optimal for distances d_{Ham} between embeddings obtained with f^{H-1}_{WL}.*

Proof. Intuitively, the transportation plan at iteration h is a "refinement" of the transportation plan at iteration $h-1$, where only a subset of the optimal transportation plans remain optimal for the new cost matrix D^h_H. Using the same notation as for the previous proof, and considering the WL procedure, two labels that are different at iteration h will also differ at iteration $h+1$. Therefore, the following identities hold:

$$\left[D^h_{\text{Ham}}\right]_{ij} \leq \left[D^{h+1}_{\text{Ham}}\right]_{ij} \quad \left[D^h_{\text{disc}}\right]_{ij} \leq \left[D^{h+1}_{\text{disc}}\right]_{ij} \tag{4.24}$$

$$\left[D^h_{\text{Ham}}\right]_{ij} \leq \left[D^h_{\text{disc}}\right]_{ij}. \tag{4.25}$$

An optimal transportation plan P^h for $f^h_{WL}(G)$ embeddings satisfies

$$\left\langle P^h, D^h_{\text{Ham}} \right\rangle \leq \left\langle P, D^h_{\text{Ham}} \right\rangle \; \forall P \in \Gamma, \tag{4.26}$$

which can also be written as

$$\left\langle P^h, D^h_{\text{Ham}} \right\rangle = \frac{1}{h}\Big((h-1)\cdot\left\langle P^h, D^{h-1}_{\text{Ham}} \right\rangle + \left\langle P^h, D^h_{\text{disc}} \right\rangle\Big). \tag{4.27}$$

The values of D^h_{Ham} increase in a step-wise fashion for increasing h, and their ordering remains constant, except for entries that were 0 at iteration $h-1$ and became $\frac{1}{h}$ at

iteration h. Since our metric distance matrices satisfy monotonicity conditions and because P^h is optimal for D^h_{disc} according to Lemma 1, it follows that

$$\left\langle P^h, D^{h-1}_{\text{Ham}} \right\rangle \leq \left\langle P, D^{h-1}_{\text{Ham}} \right\rangle \forall P \in \Gamma. \tag{4.28}$$

Therefore, P^h is also optimal for $f^{h-1}_{\text{WL}}(G)$ embeddings. □

Thanks to these two lemmas, we can show that the Wasserstein distance between categorical WL node embeddings is a conditional negative definite function.

Theorem 3. *$D_W^{f_{\text{WL}}}(\cdot,\cdot)$ is a conditional negative definite function.*

Proof. Using the same notation as for the previous proofs and the formulation in Equation 4.4, we can write

$$D_W^{f_{\text{WL}}}(G, G') = \min_{P^H \in \Gamma} \langle P^H, D^H_{\text{Ham}} \rangle \tag{4.29}$$

$$= \min_{P^H \in \Gamma} \frac{1}{H} \sum_{h=0}^{H} \langle P^H, D^h_{\text{disc}} \rangle. \tag{4.30}$$

Let P^* be an optimal solution for iteration H. Then, from Lemmas 1 and 2, it is also an optimal solution for D^H_{disc} and for all $h = 0, \ldots, H-1$. We can rewrite the equation as a sum of optimal transport problems:

$$D_W^{f_{\text{WL}}}(G, G') = \frac{1}{H} \sum_{h=0}^{H} \min_{P^* \in \Gamma} \langle P^*, D^h_{\text{disc}} \rangle. \tag{4.31}$$

This corresponds to a sum of 1-dimensional optimal transport problems relying on the discrete metric, which were shown to be conditional negative functions [86]. Therefore, the final sum is also conditional negative definite. □

Finally, we can prove Theorem 2.

Proof of Theorem 2. Theorem 2 in light of Proposition 2 implies that the WWL kernel of Definition 7 is positive definite for all $\lambda > 0$. □

Let us now consider the case of continuously attributed graphs.

Theoretical considerations for continuous embeddings

While we managed to prove the positive definiteness of our kernel for the categorical case, this is considerably more difficult to do for the continuous case. We conjecture that, under certain conditions, the kernel derived for graphs with continuous features is also positive definite. Although no formal proof is provided in this subsection, we

provide arguments to support this conjecture, which is also confirmed by empirical findings.[2]

As briefly mentioned for the categorical embeddings, the metric space induced by the Wasserstein metric for a given ground distance greatly differs. In particular, the *curvature* of such space plays an important role in the possible positive definiteness. To explore this further, we need to define *Alexandrov spaces*.

Definition 9 (Alexandrov space)**.** *Given a metric space and a real number k, the space is called an Alexandrov space if its sectional curvature is $\geq k$.*

Intuitively, the curvature indicates to what extent a geodesic triangle will be deformed in the space. The case of $k = 0$ is special as no distortion happens here—hence, spaces that satisfy this property are called *flat*. The definition of Alexandrov spaces is required for the following proposition, taken from a theorem by Feragen et al. [73], which highlights the relationship between a kernel and its underlying metric space.

Proposition 3. *The geodesic Gaussian kernel (i.e., $q = 2$ in Equation 4.15) is positive definite for all $\lambda > 0$ if and only if the underlying metric space (X, d) is flat in the sense of Alexandrov, i.e., if any geodesic triangle in X can be isometrically embedded in a Euclidean space.*

Nevertheless, it is unlikely that the space induced by the Wasserstein distance is locally flat. In fact, even the geodesics (i.e., a generalisation of the shortest path to arbitrary metric spaces) between graph embeddings are not necessarily unique, as we subsequently show. That is why we use the *geodesic Laplacian kernel* instead of the Gaussian one: it poses less strict requirements on the induced space, as stated in Proposition 2. Specifically, the metric used in the kernel function needs to be cnd. While we cannot directly prove this, we can show that the converse is not true. To this end, we first notice that the metric space induced by the GWD, which we refer to as X, does *not* have a curvature that is bounded from above.

Definition 10. *A metric space (X, d) is said to be* CAT(k) *if its curvature is bounded by some real number $k > 0$ from above. This can also be seen as a "relaxed" definition, or generalisation, of a Riemannian manifold.*

Theorem 4. *X is not in* CAT(k) *for any $k > 0$, meaning that its curvature is* not *bounded by any $k > 0$ from above.*

Proof. This follows from a similar argument presented by Turner et al. [235]. Let G and G' be two graphs. Assume that X is a CAT(k) space for some $k > 0$. Then, it follows [36, Proposition 2.11, p. 23] that if $D_W^{f_{\mathrm{WL}}}(G, G') < \pi^2/k$, there is a *unique* geodesic between them. Nonetheless, we can construct a family of graph embeddings for which this is not the case. To do so, let $\epsilon > 0$ and $f_{\mathrm{WL}}(G)$ and $f_{\mathrm{WL}}(G')$ be two

[2] We empirically observe that for all considered data sets, after standardisation of the input features before the embedding scheme, GWD matrices are conditional negative definite.

graph embeddings with node embeddings $a_1 = (0, 0)$, $a_2 = (\epsilon, \epsilon)$ as well as $b_1 = (0, \epsilon)$ and $b_2 = (\epsilon, 0)$, respectively. Because we use the Euclidean distance as a ground distance, there will be two optimal transport plans: the first maps a_1 to b_1 and a_2 to b_2, whereas the second maps a_1 to b_2 and a_2 to b_1. Hence, we have found two geodesics that connect G and G'. Since we may choose ϵ to be arbitrarily small, the space cannot be CAT(k) for $k > 0$. □

While this does not provide an upper bound on the curvature, we have the following conjecture.

Conjecture 1. *X is an Alexandrov space with curvature bounded from below by zero.*

For a proof idea, we refer to Turner et al. [235]; the main argument aims at characterizing the distance between triples of graph embeddings. This first conjecture is particularly helpful because being a nonegatively curved Alexandrov space is a necessary prerequisite for X to be a Hilbert space [216]. From there, we refer to Feragen et al. [73], who show that cnd metrics and Hilbert spaces are intricately linked.

We therefore have some hope in obtaining a cnd metric, although we do not have a clear proof yet. Our empirical results indicate that it is possible to turn the GWD into a cnd metric with proper normalisation of the input features. Intuitively, for high-dimensional spaces, standardisation of input features changes the curvature of the induced space by making it locally (almost) flat.

To further support this argumentation, we look at an existing way to ensure positive definiteness of Wasserstein distances. One can use an alternative called *sliced* Wasserstein [192], where high-dimensional distributions are projected into many random one-dimensional spaces and the Wasserstein distance is obtained by combining the one-dimensional distances. Kolouri et al. [127] showed that each single one-dimensional Wasserstein distance is conditional negative definite, guaranteeing the negative definiteness of the combined Wasserstein distance.

Practical considerations

We showed that the proposed kernel is positive definite in the case of categorically labelled graphs and can thus be used in kernel-based learning algorithm while ensuring convergence. However, we could not prove that this holds for continuously attributed graphs. Therefore, to ensure the theoretical and practical correctness of our results *in the continuous case*, we employ recently developed methods for learning with indefinite kernels.

More particularly, we leverage learning methods for Kreĭn spaces, which have been expressly designed to work with indefinite kernels [178]. Kernels that are not positive definite in fact induce reproducing kernel Kreĭn spaces (RKKS), which are a generalisation of reproducing kernel Hilbert spaces. These spaces share similar mathematical properties with RKHS and are therefore amenable to certain machine learning approaches. In particular, recent algorithms [148, 177] are capable of solving learning problems in RKKS and reported results indicate that there are clear benefits (in

terms of classification performance, for example) in learning in such spaces. Therefore, when evaluating our kernel WWL in the continuous attributes scenario, we rely on a Kreĭn SVM (KSVM, [148]) as a classifier.

4.2.4 Experimental evaluation

We now analyse the empirical performance of WWL in comparison with state-of-the-art graph kernels. In particular, we observe that WWL (i) performs on-par with the best graph kernel for categorically labelled data, and (ii) outperforms all the state-of-the-art graph kernels for attributed graphs.

Data sets

For the evaluation of the different graph kernels, we rely on real-world data sets from diverse sources [30, 215, 241]. A detailed list of the used data sets can be found in Table 4.1. Our data sets belong to multiple chemoinformatics domains, including small molecules (MUTAG, PTC-MR, NCI1), macromolecules (ENZYMES, PROTEINS, D&D) and chemical compounds (BZR, COX2). We further consider a movie collaboration data set (IMDB, see [258] for a description) and two synthetic data sets Synthie and Synthetic-new, created by Morris et al. [173] and Feragen et al. [72], respectively.

Depending on the scenario, we use their categorical labels or continuous attributes for evaluation. MUTAG, PTC-MR, NCI1, and D&D only have categorical node labels; ENZYMES and PROTEINS have both categorical labels and continuous attributes; IMDB-B, BZR, and COX2 only contain continuous attributes; finally, BZR-MD and COX2-MD have both continuous node attributes and edge weights. The BZR-MD and COX2-MD data sets do not have node attributes but contain the atomic distance between each connected atom as an edge weight. We do not consider distances between non-connected nodes [130] and we equip the node with one-hot-encoding categorical attributes representing the atom type, i.e., what is originally intended as a categorical node label. On IMDB-B, IMDB-Binary was used with the node degree as a (semi-)continuous feature for each node [258]. For all the other data sets, we use the off-the-shelf version provided by Kersting et al. [119]. All the data sets have been downloaded from Kersting et al. [119].

Experimental setup

We evaluate WWL in comparison with other state-of-the-art graph kernels as well as with relevant baselines. To guarantee maximal comparability, we use the exact same data set splits for all methods. In the categorical labels scenario, we compare with WL [214] and WL-OA [131] and with the vertex (V) and edge (E) histograms. We do not include all existing graph kernels because Kriege et al. [131] already shows that the WL-OA is superior to other existing approaches.

For the continuous attributes scenario, we compare with GraphHopper (GH) [72] and two instances of the hash graph kernel (HGK-SP; HGK-WL) [173]. Additionally,

Table 4.1: Details of the experimental data sets.

Data set	Class Ratio	Node Labels	Node Attributes	Edge Weights	# Graphs	Classes
MUTAG	63/125	✓	-	-	188	2
NCI1	2053/2057	✓	-	-	4110	2
PTC-MR	152/192	✓	-	-	344	2
D&D	487/691	✓	-	-	1178	2
ENZYMES	100 per class	✓	✓	-	600	6
PROTEINS	450/663	✓	✓	-	1113	2
BZR	86/319	✓	✓	-	405	2
COX2	102/365	✓	✓	-	467	2
SYNTHIE	100 per class	-	✓	-	400	4
IMDB-Binary	500/500	-	(✓)	-	1000	2
Synthetic-new	150/150	-	✓	-	300	2
BzR-MD	149/157	✓	-	✓	306	2
COX2-MD	148/155	✓	-	✓	303	2

we compare with a simple continuous vertex histogram baseline (VH-C), which is defined as a radial basis function (RBF) kernel between the sums of the obtained graph node embeddings. Furthermore, to assess the advantage of using the Wasserstein distance in our method, we compare with a variation of our technique where we replace it with an RBF kernel. Specifically, given two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, with $|V_1| = n_1$ and $|V_2| = n_2$, we start by computing the Gaussian kernel between each pair of node embeddings obtained with the same graph embedding scheme as for WWL; therefore obtaining a kernel matrix between node embeddings $K' \in n_1 \times n_2$. We then sum up all the values $K_s = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} K'_{i,j}$ and set $K(G_1, G_2) = K_s$. We repeat this procedure for each pair of graphs to obtain the final graph Gram matrix and refer to this baseline as RBF-WL.

To solve the classification task, we rely on a Support Vector Machine (SVM), or a Kreĭn-SVM (KSVM) for WWL in the continuous scenario. For each data set, we use a 10-fold cross-validation, selecting the parameters on the training set only. We repeat each cross-validation split 10 times and report average and standard deviation of the classification accuracies. As mentioned above, the used splits are *exactly the same* for each evaluated method to ensure a fully comparable setup.

The ranges used for the hyperparameter selection are the following: the parameter of the SVM $C = \{10^{-3}, \ldots, 10^3\}$ (for continuous attributes) and $C = \{10^{-4}, \ldots, 10^5\}$ (for categorical attributes); the WL number of iterations $h = \{0, \ldots, 7\}$; the λ parameter of the WWL $\lambda = \{10^{-4}, \ldots, 10^1\}$. For RBF-WL and VH-C, we use the default γ parameter for the Gaussian kernel, i.e., $\gamma = 1/m$, where m is the size of the node attributes. For the GH kernel, we also fix the γ parameter to $1/m$. For HGK, we set the number of iterations to 20 for each data set, except for SYNTHETICnew where we use 100 (these setups were suggested by the respective authors [72, 173]). Moreover, since HGK is a randomised method, we compute each kernel matrix 10 times and average the results. When the dimensionality of the continuous attributes $m > 1$, we normalise the input features to ensure comparability among the different

Table 4.2: Classification accuracies on graphs with categorical node labels. Comparison of Weisfeiler–Lehman kernel (WL), optimal assignment kernel (WL-OA), and Wasserstein Weisfeiler–Lehman (WWL, ours).

Method	MUTAG	PTC-MR	NCI1	PROTEINS	D&D	ENZYMES
V	85.39±0.73	58.35±0.20	64.22±0.11	72.12±0.19	78.24±0.28	22.72±0.56
E	84.17±1.44	55.82±0.00	63.57±0.12	72.18±0.42	75.49±0.21	21.87±0.64
WL	85.78±0.83	61.21±2.28	85.83±0.09	74.99±0.28	78.29±0.30	53.33±0.93
WL-OA	**87.15±1.82**	60.58±1.35	**86.08±0.27**	**76.37±0.30***	79.15±0.33	**58.97±0.82**
WWL	**87.27±1.50**	**66.31±1.21***	85.75±0.25	74.28±0.56	**79.69±0.50**	**59.13±0.80**

Table 4.3: Classification accuracies on graphs with continuous node or edge attributes. Comparison of hash graph kernel (HGK-WL, HGK-SP), GraphHopper kernel (GH), and Wasserstein Weisfeiler–Lehman (WWL, ours).

Method	ENZYMES	PROTEINS	IMDB-B	BZR	COX2	BZR-MD	COX2-MD
VH-C	47.15±0.79	60.79±0.12	71.64±0.49	74.82±2.13	48.51±0.63	66.58±0.97	64.89±1.06
RBF-WL	68.43±1.47	75.43±0.28	72.06±0.34	80.96±1.67	75.45±1.53	**69.13±1.27**	71.83±1.61
HGK-WL	63.04±0.65	75.93±0.17	73.12±0.40	78.59±0.63	**78.13±0.45**	**68.94±0.65**	74.61±1.74
HGK-SP	66.36±0.37	75.78±0.17	73.06±0.27	76.42±0.72	72.57±1.18	66.17±1.05	68.52±1.00
GH	65.65±0.80	74.78±0.29	72.35±0.55	76.49±0.99	76.41±1.39	**69.14±2.08**	66.20±1.05
WWL	**73.25±0.87***	**77.91±0.80***	**74.37±0.83***	**84.42±2.03***	**78.29±0.47**	**69.76±0.94**	**76.33±1.02**

feature scales. This is performed in every data set except for BZR and COX2, due to the meaning of their node attributes, which are 3-D location coordinates.

Finally, to implement our method, we rely on existing Python implementations for the WL kernel [224] and the Wasserstein distance [77] and make our code publicly available on GitHub. Additionally, we rely on the original implementations of the comparison partners to compute WL-OA, HGK and GH. All our analyses were performed on a shared server running Ubuntu 14.04.5 LTS, with 4 CPUs (Intel Xeon E7-4860 v2 @ 2.60GHz) each with 12 cores and 24 threads, and 512 GB of RAM.

Results and discussion

Table 4.2 and Table 4.3 summarise the classification accuracy results for the categorically labelled and continuously attributed data sets, respectively. The best performing methods up to the resolution implied by the standard deviation across repetitions are highlighted in boldface. Moreover, we evaluate the significance by performing 2-sample t-tests with a significance threshold of 0.05 and Bonferroni correction for multiple hypothesis testing within each data set. Methods that are significantly outperforming the other ones are highlighted by an asterisk. Additionally, Figure 4.2 shows a visual comparison of the performance across continuously attributed data sets. Furthermore, we report results on synthetic data (Synthie and SYNTHETIC-new) in Table 4.4.

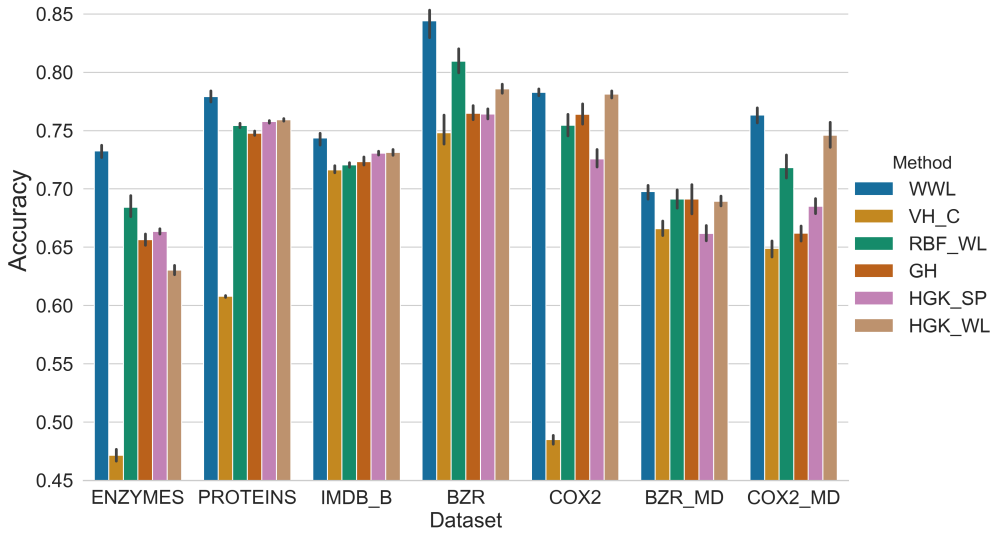


Figure 4.2: Classification accuracies on graphs with continuous node or edge attributes. Comparison of vertices histogram baseline (VH-C), RBF Weisgeiler–Lehman (RBF-WL), hash graph kernel (HGK-WL, HGK-SP), GraphHopper kernel (GH), and Wasserstein Weisfeiler–Lehman (WWL, ours).

Categorical labels. In the categorical scenario, WWL improves over the classical WL. Moreover, it largely improves over WL-OA in PTC-MR and is slightly better on D&D, whereas WL-OA is better on NCI1 and PROTEINS. The WL-OA approach offers very comparable performance with our method. This is unsurprising, as the main idea behind WL-OA is to solve the optimal assignment problem by defining Dirac kernels on histograms of node labels across multiple iterations of WL, which shares motivations with our method. However, this formulation relies on optimal assignment rather than the optimal transport, therefore requiring one-to-one mappings instead of continuous transport maps. Moreover, we solve the optimal transport problem on concatenated embeddings, therefore jointly using representations at multiple iterations of the WL scheme. On the contrary, WL-OA performs optimal assignment at each iteration separately, combining them in a second stage. WWL therefore offer strong performance for graphs with categorical labels but, most importantly, can handle continuously attributed graphs very well.

Continuous attributes. In the continuous attributes setting, WWL is statistically significantly better on 4 out of 7 data sets and is on par on the last 3. To validate these observations, we compute the average rank of each method in the continuous scenario. The ranks calculated from Table 4.3 are WWL = 1, HGK-WL = 2.86, RBF-WL = 3.29, HGK-SP = 4.14, and VH-C = 5.86. WWL always scores first: this is a considerable improvement over the state of the art. As discussed in Section 4.2.3, the kernel derived from continuous attributes is not guaranteed to be

positive definite. Nevertheless, in practice we observe the kernel matrices to be positive definite, further supporting the theoretical considerations previously discussed.

Synthetic data sets. The results for synthetic data sets are presented in a separate table due to the severely unstable and unreliable results that we obtained. For both data sets, the variation between the different methods is high and minor changes in the node features (e.g. normalisation or scaling of the embedding scheme) resulted in substantial change of performance (up to 15%). Finally, other authors showed that a WL with degree treated as a categorical node label outperforms most of the competitors on SYNTHETIC-NEW, indicating that the node attributes are not informative[72]. These reasons led us to exclude these data sets from the main analysis, as they could not be used to fairly assess the quality of the evaluated methods.

Comparison with hash graph kernels. Among the existing methods designed for continuously attributed graphs, HGK is the closest to our approach, as it shares a somewhat related propagation scheme. By relying on multiple random hashing functions, the HGK extends a set of existing graph kernels to the continuous setting, overcoming the limitations of perfect hashing, which cannot account for small differences in continuous attributes. The main drawback of the random hashing of HGK is that it requires additional parameters and introduces stochasticity in the computation of the kernel. By contrast, our propagation scheme is fully continuous and relies on the Wasserstein distance to capture small differences in distributions of continuous node attributes. Finally, the performance gap observed in practice highlights the benefits of an entirely continuous representation of the graphs over hashing.

Performance under noise. Finally, we perform an additional experiment to evaluate the difference between WL and WWL for noisy Erdős–Rényi graphs ($n = 30$, $p = 0.2$). We report the relative distance between G and its permuted and perturbed variant G', with respect to a third independent graph G'' for an increasing level of

Table 4.4: Classification accuracies on synthetic graphs with continuous node attributes. Comparison of hash graph kernel (HGK-WL, HGK-SP), GraphHopper kernel (GH), and Wasserstein Weisfeiler–Lehman (WWL, ours).

METHOD	SYNTHIE	SYNTHETIC-NEW
VH-C	27.51 ± 0.00	60.60 ± 1.60
RBF-WL	94.43 ± 0.55	86.37 ± 1.37
HGK-WL	81.94 ± 0.40	$\mathbf{95.96 \pm 0.25}^*$
HGK-SP	85.82 ± 0.28	80.43 ± 0.71
GH	83.73 ± 0.81	88.83 ± 1.42
WWL	$\mathbf{96.04 \pm 0.48}^*$	86.77 ± 0.98

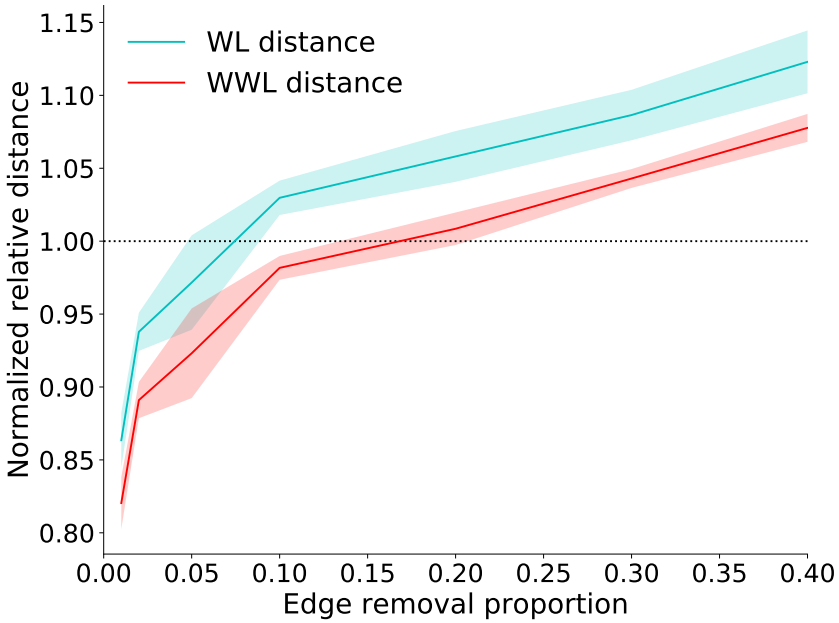


Figure 4.3: Relative distance between (Erdős–Rényi) graph G and the relative permuted and perturbed variant G' with respect to a third independent graph G'' for an increasing noise level for both the Weisfeiler–Lehman (WL) and the Wasserstein Weisfeiler–Lehman (WWL) distances.

noise (i.e., edge removal) in Figure 4.3. We see that WWL is more robust against noise.

Runtime considerations

In the categorical labels scenario, both WL and WL-OA scale linearly with the number of nodes, therefore being faster than WLL. Since the different methods we consider rely on various programming languages and implementation, it is tricky to provide an accurate runtime comparison. Nevertheless, the Wasserstein graph kernels remain competitive as the kernel matrix can still be computed in a median time of 40 s, depending on the number and size of graphs. In the continuous attributes scenario, our approach has a runtime similar to GH. Nonetheless, GH was shown to empirically scale quadratically with the number of graph nodes [72], which is faster than the computation of the Wasserstein distance with complexity $\mathcal{O}(n^3 log(n))$. On the other hand, HGK is considerably slower due to the multiple repetitions needed to balance the randomisation.

As mentioned in Section 4.1.2, the complexity of the Wasserstein distance computation can be reduced to *near linear* time by using the Sinkhorn approximation [5]. To evaluate the benefits of such an approximation on WWL, we simulate a fixed number of graphs with varying average number of nodes per graph and measured the execution speed of our method. We generate random node embeddings for 100

graphs, where the number of nodes is taken from a normal distribution centered around the average number of nodes. We then compute the kernel matrix for each set of graphs to compare the runtime of regular Wasserstein with the Sinkhorn regularised optimisation. Figure 4.4 shows how the speedup begins to be beneficial at approximately 100 nodes per graph, which is larger than the average number of nodes in the benchmark data sets we used.

Moreover, we want to ensure that using Sinkhorn approximation does not decrease our model accuracy. We therefore evaluate it on the ENZYMES data set. Recalling that the Sinkhorn method solves the following entropic regularisation problem,

$$P^{\gamma} = \underset{P \in \Gamma(X, X')}{\arg\min} \langle P, M \rangle - \gamma h(P),$$

we further need to select γ and we do that in the cross-validation step. We obtain a final accuracy of 72.08 ± 0.93, which remains above the current state of the art. Values of γ selected most of the time are 0.3, 0.5, and 1.

Concluding remarks

We here presented a new family of graph kernels: the Wasserstein Weisfeiler–Lehman (WWL) graph kernels. We provide theoretical motivations for our approach and show that our method outperforms the state of the art for graph classification in the scenario of continuous node attributes and matches the state of the art in the categorical labels scenario. WWL is a great way to better capture subtle similarities between graphs that are either categorically labelled or continuously attributed and supports the aptness of Optimal Transport for machine learning applied to structured objects, as introduced in Section 4.1.

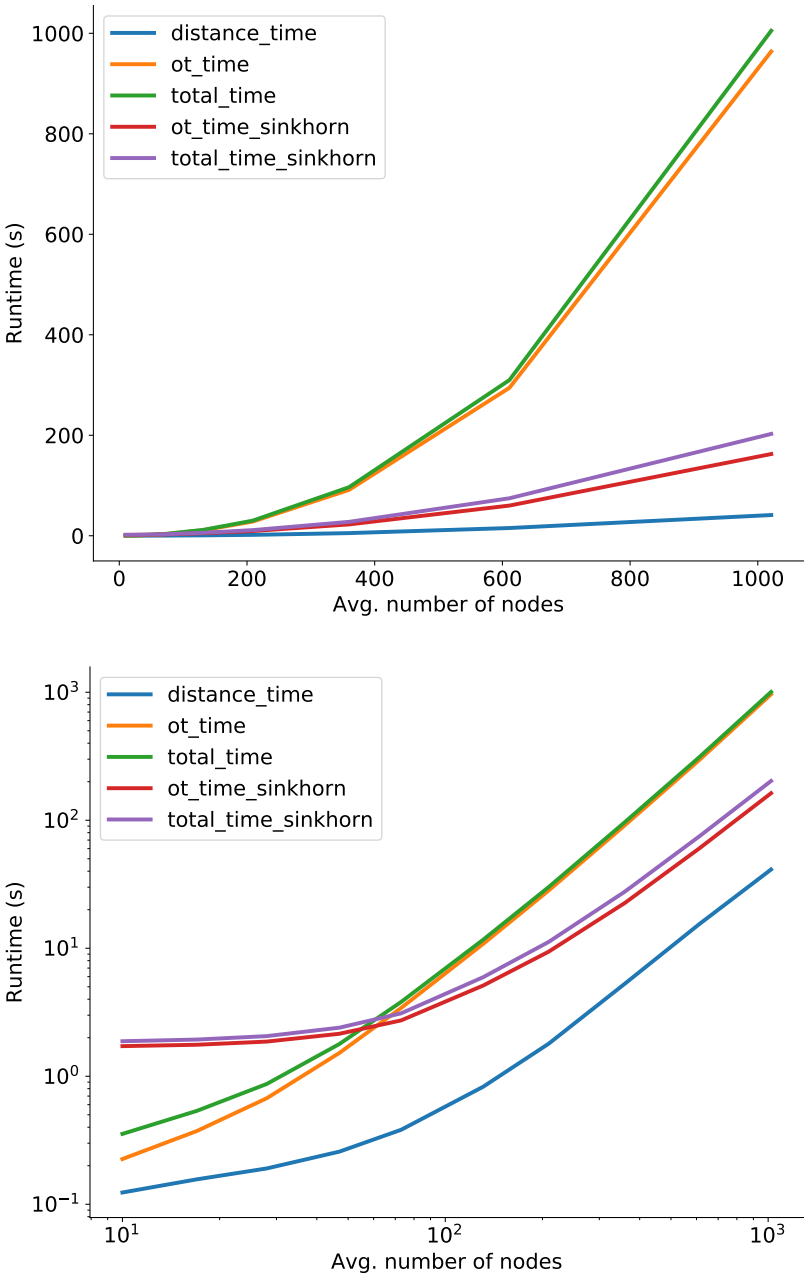


Figure 4.4: Runtime performance of the WWL Kernel computation step with a fixed number of graphs. We also report the time taken to compute the ground distance matrix as `distance_time`. Here, `total_time` is the sum of the time to compute the ground distance and the time taken to solve the optimal transport (ot) problem for the regular solver or the Sinkhorn-regularised one. The logarithmic scale on the right-side figure shows how, for a small average number of nodes, the overhead to run Sinkhorn is higher than the benefits.

4.3 Wasserstein Time series Kernels

Time series are even more widespread than graphs in biomedical applications [25, 109]. Devising resilient machine learning algorithms for such applications is therefore of high interest, making time series classification (TSC) a particularly active area of research. Many TSC approaches have been proposed, relying on very diverse methodologies. Some make use of short and predictive subsequences [260] while others are based on distance measures such as dynamic time warping (DTW). A comprehensive overview of competitive methods for TSC was assembled by Bagnall et al. [12]. Nevertheless, few methods relying on kernels have been advanced and their successes for TSC are limited. This is due to two main reasons: (i) the similarity measures used are either insensitive or hypersensitive to time shifts, and (ii) subsequence-based kernels naïvely built using the $\mathcal{R}$-Convolution framework are generally meaningless.

Hence, building on what was done for graphs in Section 4.2, we propose a meaningful kernel for time series that captures both the similarities between subsequence *distributions* in addition to their pairwise similarities. In this Section, we present the Wasserstein Time Series Kernel (WTK) and show its practical benefits.

4.3.1 Time series kernels

As already briefly mentioned, few kernel methods have been proposed for TSC. Here, we present some existing definite and indefinite kernels (as defined in Section 4.1.1). The very first kernel-based classification approaches encompassed standard SVM kernels (linear, RBF) on whole time series [205]. Then, kernels relying on cross-correlation were proposed to capture periodic patterns [243]. In parallel, methods based on DTW kernels [150] or alignment of full time series [55, 58] were published. The DTW-based kernels being indefinite in general, investigations of the impact of indefinite kernels on classification performance lead to recursive edit distance kernel for TSC [165].

Daliri [59] proposes KEMD, a kernel using the earth mover distance on the histograms of the time series data points, and evaluate it on EEG classification. While this kernel also relies on optimal transport, it fundamentally differs from the one we propose, as we specify in Section 4.3.2. Finally, Cuturi and Vert [58] define an alignment kernel through the polytope of all possible alignments (two sequences are similar if they share a wide set of efficient alignments) - extended in [55] to rely less on global information.

Nevertheless, methods that apply the $\mathcal{R}$-Convolution framework on time series can suffer from certain pitfalls. In particular, this can lead to completely meaningless similarity measures. If one considers a time-series as a structured object where its subsequences are the parts, the construction of a naïve $\mathcal{R}$-Convolution kernel

resembles the following. Let T, T' refer to two time series, and $\mathcal{S}, \mathcal{S}'$ to their respective sets of subsequences, the obtained kernel is defined as

$$\mathrm{k}(T, T') := \frac{1}{|T| \cdot |T'|} \sum_{S \in \mathcal{S}} \sum_{S' \in \mathcal{S}'} \mathrm{k}_{\text{base}}(S, S'), \tag{4.32}$$

where k_{base} represents the base kernel function. Choosing a linear kernel as k_{base} (or equivalently, the standard inner product in Euclidean space), will lead to

$$\begin{aligned} \mathrm{k}(T, T') &= \frac{1}{|T| \cdot |T'|} \sum_{S \in \mathcal{S}} \sum_{S' \in \mathcal{S}'} S^\top S' \\ &\approx \frac{1}{|T| \cdot |T'|} \left(\sum_{S \in \mathcal{S}} S^\top \right) \left(\sum_{S' \in \mathcal{S}'} S' \right) \\ &\approx \overline{T}^\top \overline{T}', \end{aligned} \tag{4.33}$$

where the last approximation is given by the observation that in the respective sums over all subsequence feature vectors, all the observations of length k, except for the *leading* $k-1$ as well as the *trailing* $k-1$ observations, will appear at all dimensions in the sum. In other terms, for several values of k, the basic $\mathcal{R}$-Convolution above degenerates into a simple comparison of the mean values of T and T'. As a consequence, the kernel becomes *meaningless*, especially in the case of z-normalised data sets, which are suggested to always be use in time series analysis [194]. A similar argumentation has been made by Keogh and Lin [117], where the authors argue that clustering time series can be inherently meaningless.

Figure 4.5 shows how this is observed in practice for some data sets from the UCR Time Series Archive (see Section 4.3.4 for details). The x-axis indicates the subsequence length w, while the y-axis depicts the mean of the kernel matrix values. For small values of the subsequence length, the mean tends to stay near zero for most data sets. As some of our experiments in Section 4.3.4 further show, even increasing the subsequence length w to a significant percentage of the original time series length does *not* result in decent predictive performance.

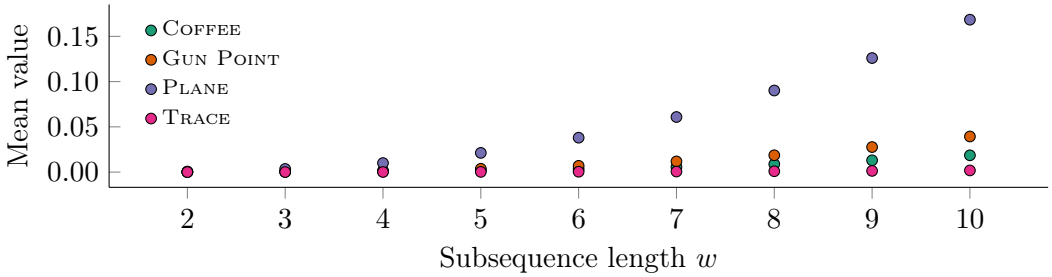


Figure 4.5: The mean value of a kernel matrix constructed for a linear kernel, using a straightforward application of the $\mathcal{R}$-Convolution framework.

To avoid this construction pitfall, here too, we leverage concepts from optimal transport (see Section 4.1.2) and define a competitive kernel for TSC.

4.3.2 A subsequence-based Wasserstein kernel

We here define and describe our novel subsequence-based Wasserstein kernel. Let $w \in \mathbb{N}_{>0}$ refer to a window width (i.e. a subsequence length). Given a set of n time series $\mathcal{T} := \{T_1, \ldots, T_n\}$ we denote their set of length-w subsequences as $\mathcal{S} := \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$. For time series with a uniform length of m, the set $\mathcal{S}_i$ therefore contains $m - w + 1$ subsequences.

Definition 11 (Wasserstein time series kernel)**.** *Let T_i and T_j be two time series, and $S_{i1}, \ldots, S_{iL}$ as well as $S_{j1}, \ldots, S_{jK}$ be their respective subsequences. Moreover, let D be a $K \times L$ matrix that contains the pairwise distances between all of the subsequences, such that*

$$D_{kl} := d(S_{ik}, S_{jl}), \tag{4.34}$$

where $d(\cdot, \cdot)$ denotes the usual Euclidean distance. Following Definition 3, we solve the optimisation problem

$$\mathrm{W}_1(T_i, T_j) := \min_{P \in \Gamma(T_i, T_j)} \langle D, P \rangle_{\mathrm{F}}, \tag{4.35}$$

which gives the optimal transport cost to transform T_i into T_j using their subsequences. Then, given $\lambda \in \mathbb{R}_{>0}$, we can define

$$WTK(T_i, T_j) := e^{-\lambda \mathrm{W}_1(T_i, T_j)}, \tag{4.36}$$

which we refer to as our Wasserstein-based subsequence kernel.

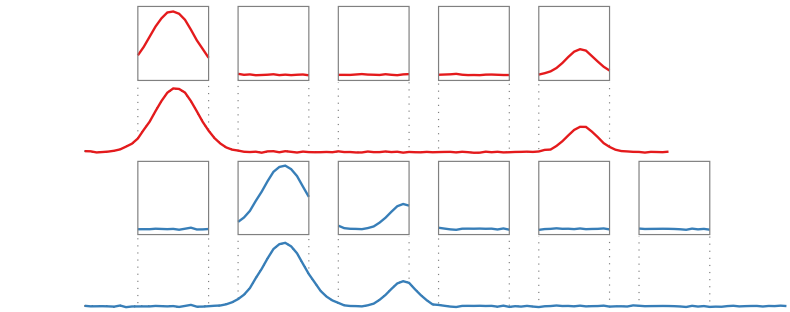
In the remainder of the thesis, since we consider that a time series T_i is represented by its set of subsequences $\mathcal{S}_i$, to simplify the notation, we will also write

$$\mathrm{W}_1(\mathcal{S}_i, \mathcal{S}_j) := \mathrm{W}_1(T_i, T_j) \tag{4.37}$$

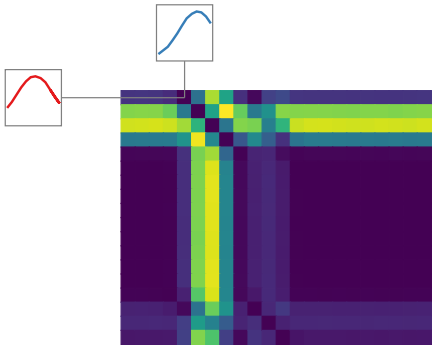
and

$$WTK(\mathcal{S}_i, \mathcal{S}_j) := WTK(T_i, T_j). \tag{4.38}$$

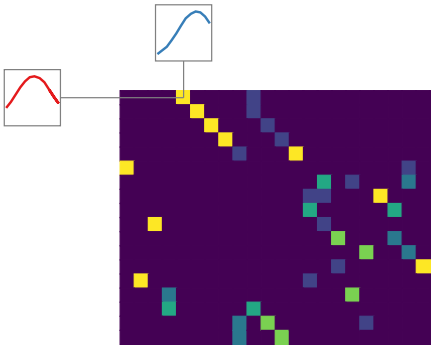
We further motivate that this kernel can be seen as an $\mathcal{R}$-convolution kernel with a single decomposition because $\mathrm{W}_1(\cdot, \cdot)$ is permutation-invariant, meaning that the order in which these subsequences are detected does *not* matter, as required in Theorem 1. As mentioned in Section 4.3.1, despite some similar theoretical background, WTK differs substantially from the Kernel Earth Mover's Distance (KEMD) method proposed by Daliri [59]. KEMD is a histogram intersection kernel [159] that treats each time series as a one-dimensional distribution of scalar values. Our approach, on the other hand, measures the distance between high-dimensional distributions of *subsequences*. It is therefore much better suited to capture long-distance similarities of subsequences and time series.



(a) Subsequence extraction



(b) Subsequence distance matrix



(c) Optimal transport plan

Figure 4.6: To measure the distance between two time series, our method proceeds in several steps. (a) First, all subsequences of the two time series are obtained using a sliding window approach (here, not all subsequences are shown due to the overlap of their windows). (b) Second, the pairwise distance matrix between all subsequences is calculated. Yellow highlights large distances, while blue shows small distances. This matrix on its own is not sufficient to assess the dissimilarity between the two time series, since it is unclear which subsequences correspond to which other. (c) Calculating the optimal transport plan makes correspondences between subsequences more readily visible. For example, the two highlighted subsequences are matched with each other in the plan. Since the two time series have different lengths, some rows of the transport plan also contain fractional matchings, making it possible to individuate fine-grained differences in the distributions of the subsequences.

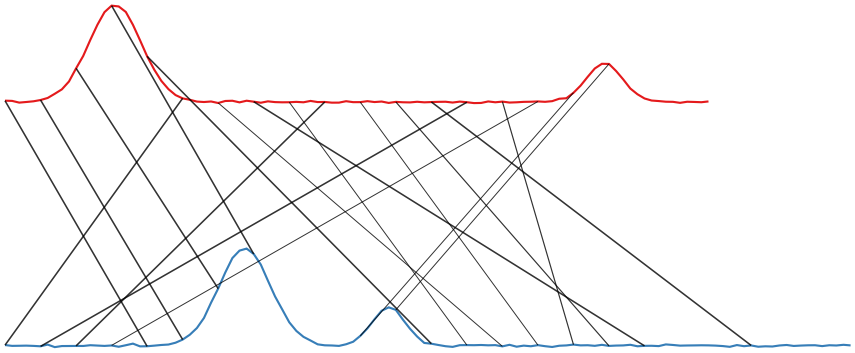


Figure 4.7: An explicit visualisation of the transport plan obtained in Figure 4.6c. Every line indicates a (partial) match between two subsequences. The lines are anchored to the beginning of the respective subsequence and their thickness reflect the transport value. Only the largest values are reported.

Intuition

WTK leverages the descriptive power of subsequences of a time series, as do other shapelet-based methods such as the Matrix Profile [261]. To provide a better understanding of the steps performed to obtain our kernel, we show a visual description of the procedure in Figure 4.6. We therefore see the length-w subsequence extraction (Figure 4.6a), followed by pairwise distance calculations (Figure 4.6b), and the final calculation of the optimal transport plan (Figure 4.6c).

The optimal transport plan P obtained after solving the optimisation problem in Equation 4.35 can be seen as a map assigning each subsequence of the first time series (columns) to at least one subsequence of the second time series (rows). The goal being to transport *all* subsequences, ultimately, every subsequence (i.e. row or column) must contain values. Figure 4.7 highlights how the transport plan values represent the mapping of the time series. The example shows that the optimisation procedure selects the lowest distances between subsequences and aligns the peaks of the time series. Moreover, since the optimisation problem accounts for sets of different cardinalities, our method can be applied to time series of varying length. This is particularly important for many TSC applications.

Finally, the Wasserstein distance is obtained by summing the element-wise multiplication values of the two matrices shown in Figure 4.6. The final distance value better captures the difference between the time series in term of the subsequence distributions rather than simply summing all the pairwise distances.

Extensions

The definition of the proposed kernel leaves room for several extension. In particular, WTK easily allows for a different ground distance measure. In fact, the Euclidean distance choice was mostly motivated by experimental practice, but the choice of a distance was shown to be crucial to obtain good predictive performance in shapelet-

Algorithm 4 Compute Wasserstein time series kernel

Input: Two time series T_1, T_2; subsequence length w; λ
Output: $k_{WTK}(T_1, T_2)$

$\mathcal{S}_1 \leftarrow$ SUBSEQUENCES(T_1, w) // Extract subsequences for T_1
$\mathcal{S}_2 \leftarrow$ SUBSEQUENCES(T_2, w) // Extract subsequences for T_2
$d_W \leftarrow \mathrm{W}_1(\mathcal{S}_1, \mathcal{S}_2)$ // Compute the Wasserstein distance
$k_{WTK}(T_1, T_2) \leftarrow e^{-\lambda d_W}$ // Apply the Laplacian kernel

based methods [160]. The only limitation is that the chosen distance needs to satisfy the axiom of a metric to ensure that the Wasserstein distance remains a metric itself. This precludes DTW, which is known not to satisfy the triangle inequality [118].

Another possible extension is to consider subsequences of multiple lengths, in order to capture similarities across different scales. One could simply combine (e.g. sum) the kernels for different lengths, therefore obtaining a kernel combining scales, but this would prevent the capture of similarities *across* scales. The challenge here resides in finding a novel way to compute distances between subsequences of different lengths, e.g. k and k'. The sliding Euclidean distance, a commonly-used distance for shapelet mining [260], is *not* a metric because it does not statisfy the "identity of indiscernibles"[3]. Additionally, the inclusion of even more subsequences leads to higher computational costs (see Section 4.3.4), which should be somehow mitigated.

As already highlighted in Section 4.2.3, obtaining positive definite kernels is usually preferable, we will now see how our defined method behaves in that regard.

Implementation & computational complexity

Algorithm 4 summarises the steps to obtain our newly defined kernel. It requires, in addition to the two time series, a subsequence length parameter $w \in \mathbb{N}_{>0}$ and the Laplacian weight $\lambda \in \mathbb{R}_{>0}$.

The complexity of WTK encompasses the following parts: (i) Subsequence extraction, (ii) Subsequence distance calculation, and (iii) Wasserstein metric calculation. We denote by n the number of time series and m the length of the time series (we assume that they are all of the same length but the derivation can easily be extended to the case where m is the maximum length). We therefore have at most $s := m - w + 1$ subsequences per time series and the extraction process is dominated by m, leading to a total complexity for step i of $\mathcal{O}(nm)$. This pre-processing step is shared with other methods, such as shapelet extraction methods [260] or matrix profile methods such as MPDIST [88].

Next, the pairwise distance computation between subsequences of two time series requires s^2 distance calculations, each with a complexity of w. Hence, in the worst

[3] This property states that $\mathrm{dist}(x, y) = 0$ if and only if $x = y$. For the sliding Euclidean distance, any subsequence S of some time series T satisfies $\mathrm{dist}(S, T) = 0$, even though $S \neq T$. The property is therefore not satisfied.

case scenario, the calculation has a complexity of $\mathcal{O}(s^2 w)$, while it is possible to reduce this significantly in the case of Euclidean distances by re-using calculations.

Finally, computing the Wasserstein distance between two time series using Equation 4.35 has a complexity of $\mathcal{O}(s^3 \log s)$ for an input matrix of maximum dimensions $s \times s$ [5]. Since m is an upper bound to the number of subsequences of fixed length s, the asymptotic runtime of all parts can thus be summarised as $\mathcal{O}(n^2 m^3 \log m)$.

This is a worst-case approximation of the runtime and here too, approximations of the Wasserstein distance relying on Sinkhorn can be used to achieve *near linear-time* complexity [5, 19, 57].

4.3.3 Theoretical considerations

In order to obtain a positive definite kernel that belongs to an RKHS, it must satisfy Equation 4.13. In a line of argument similar to the one discussed for the continuous case of the graphs in Section 4.2.3, we can only conjecture on conditions that will lead to a positive definite kernel. In fact, the Wassestein distance equipped with the euclidean distance induces a metric space that cannot provably be mapped to a space equipped with the L^2-norm. To show this, we would always need to have a matrix of Wasserstein distances that is conditionally negative definite (as mentioned in Proposition 2), which implies that it has at most *one* positive eigenvalue [15, Lemma 4.1.4, p. 163].

In practice, however, our empirical results indicate that for some data sets and specific configurations, we have *more* than one positive eigenvalue in $\mathcal{D}$. Therefore, the kernel matrix $\mathcal{K}$, whose entries are defined as $\mathcal{K}_{ij} = WTK(T_i, T_j) := e^{-\lambda \mathrm{W}_1(T_i, T_j)}$, is not positive definite. This clearly indicates that the properties of the time series influence the induced metric. We therefore have several options:

(i) We can *enforce* the eigenvalue condition by calculating $\mathcal{K}' := \mathcal{K} \cdot \mathcal{K}^\top$, where $\mathcal{K}$ refers to the $n \times n$ matrix with entries according to Equation 4.36. Letting $\boldsymbol{y} := \mathcal{K}^\top \boldsymbol{x}$ for $\boldsymbol{x} \in \mathbb{R}^n$, we then have $\boldsymbol{x}^\top \mathcal{K} \mathcal{K}^\top \boldsymbol{x} = \boldsymbol{x}^\top \mathcal{K} \boldsymbol{y} = \boldsymbol{y}^\top \boldsymbol{y} = \sum_{i=1}^n y_i \geq 0$, so $\mathcal{K}'$ is positive definite. This is also known as the *empirical kernel*. This option is the easiest from a computational perspective: it only requires an additional matrix multiplication. Nonetheless, it affects the similarity values between time series and we empirically observe that the classification accuracy is diminished by the enforcement.

(ii) We can *regularise* the matrix by subtracting all negative eigenvalues, yielding $\mathcal{K}' := \mathcal{K} - \sum_i \lambda_i \boldsymbol{v_i} \boldsymbol{v_i}^\top$, where i ranges over the indices of the negative eigenvalues and v_i denotes their corresponding unit eigenvectors. By construction, this will set negative eigenvalues to zero, leaving us with a positive definite matrix. This option is computationally more demanding as it requires a full eigendecomposition of the kernel matrix. Wu et al. [251] describe several transformations and show that the `shift` of the spectrum has negligible impact on the computational performance, while also having the lowest impact on the predictive performance.

(iii) We can *generalise* the Wasserstein distance by using a "softmin" of all possible transportation plans, which ensures that we obtain a positive definite kernel [252]. This approach was originally presented by Cuturi et al. [58]. In our case, the resulting kernel values would be given by

$$\text{SoftWTK}(S_i, S_j) := \sum_{\gamma \in \Gamma(\sigma,\mu)} \exp\left(-\lambda\left(\int_{\mathcal{M}x\mathcal{M}} \text{dist}(x,y)^p \mathrm{d}\gamma(x,y)\right)^{\frac{1}{p}}\right) \tag{4.39}$$

which is a kernel under the condition that $\text{dist}/\text{dist}+1$ is positive definite. Nonetheless, it requires the computation of the permanent of D_{kl}, which scales super-exponentially with the number of subsequences; please refer to [56] and Figure 4.12 for more information. SoftWTK is therefore infeasible for all practical purposes and we do not include it in our experiments.

(iv) We can *sidestep* the eigenvalue problem by using algorithms that are capable of handling *indefinite* matrices better [177]. This is the easiest option and, as mentioned in the introductory Section 1, a plethora of methods to learn with indefinite kernels have been proposed. Indefinite kernels are a valuable approach and training them is nowadays easier than ever.

We explored Options (i) and (ii) to guarantee the positive definiteness of our similarity measure. However, none of these options showed a significantly better classification performance with respect to WTK, indicating that the use of algorithm with indefinite kernels can be used without any problem for our situation. Moreover, the vast majority of the data sets in our experiments yielded a positive definite matrix $\mathcal{K}$. Therefore, we refer to WTK as a *kernel*, that sometimes only has a corresponding Kreĭn space (whose existence is guaranteed) rather than a corresponding Hilbert space. To ensure the soundness of our calculations, we here too employ a Kreĭn SVM [148] to classify the time series in the considered data sets.

4.3.4 Experimental evaluation

In this Section, we analyse the classification performance of WTK compared to other TSC methods. We are particularly interested in comparing it with (i) different subsequence-based methods, (ii) established baselines such as DTW, and (iii) state-of-the-art (SOTA) methods for time series classification. We show that it outperforms naïve kernels derived with the $\mathcal{R}$-convolution framework and that it performs on par with highly complex ensemble methods despite its simplicity.

Data sets

The standard database for the benchmarking of time series classification algorithms is the "UCR Time Series Archive" [45], a repository of 85 labelled time series that was recently increased to 128 time series [60]. Each data set consists of time series of varying lengths, though in each data set the time series length is fixed and a predefined train/test split of variable size. Please refer to http://www.timeseriesclassification.com

for additional details. We perform all our experiments on the 85 time series using the predetermined splits.

Experimental setup

Summary of the experiments. First, we compare WTK to other kernels that are based on subsequences in order to demonstrate that a straightforward $\mathcal{R}$-convolution kernel is meaningless and that the Wasserstein distance for subsequence comparison is better suited for time series classification. We then perform a direct comparison to DTW-1-NN using a "Texas Sharpshooter" plot, which shows that WTK leads to consistent predictions. Finally, we conclude with a large-scale comparison of our method against the respective state of the art for every data set.

Comparison partners. There is a multitude of time series classification approaches. Bagnall et al. [12] give a comprehensive view of alternatives. In addition to these, Wang et al. [247] established a baseline of neural network techniques, comprising a fully convolutional network as well as a residual network architecture, among others. In our experiments, in addition to the kernel and DTW-1-NN baselines, we compare against most of them: from fully convolution networks over ensemble methods such as Elastic Ensemble (EE) [144], FLAT-COTE [13], and HIVE-COTE [143] to shapelet-based classifiers such as Shapelet Transform (ST) [31] and Learned Shapelets (LS) [92]. We also evaluate against results obtained with a rotation forest [203] with 50 trees (RotF), a random forest [35] with 500 trees, a classifier based on a combination of DTW distances and SAX [141] histograms (DTW_F) [116], the SAX Vector Space Model (SAXVSM) [210], as well as as the Bag of Symbolic Fourier Approximation Symbols (BOSS) [206] method. Finally, we include several baselines such as the 1-nearest neighbour based on Euclidean Distance (E-1NN) and a Bayesian network (BN).

Training and evaluation. We evaluate the classification accuracy on the test set, selecting the parameters on the training set via a 5-fold cross validation using a Kreĭn SVM classifier [148] with the following parameter grid:

(i) $\gamma = \{10^{-5}, 10^{-4}, \ldots, 10^{3}\}$ (for the RBF kernel),

(ii) $\lambda = \{10^{-4}, 10^{-3}, \ldots, 10\}$ (for WTK),

(iii) $C = \{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ (for the SVM classifier).

We also select the length w of the subsequences using a percent grid: we select potential values of w as 10 %, 30 %, and 50 % of the original time series length.

Implementation. Our implementation uses `Python 3.7` and `POT`, the Python Optimal Transport library [77]. We make our code publicly available on GitHub.

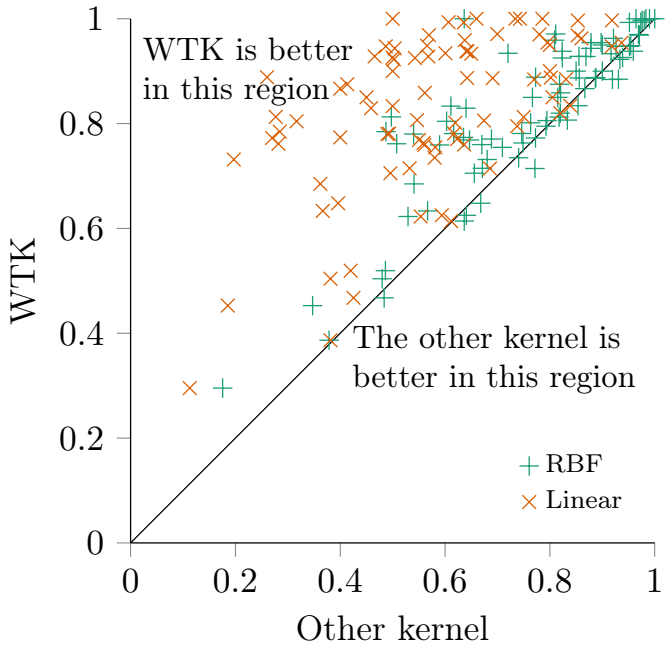


Figure 4.8: Comparison of the classification accuracy of *WTK* against the Linear and the RBF kernels for the "UCR Time Series Archive" data sets.

Results and discussion

Comparison to other kernels. We start by comparing WTK against other subsequence-based kernels: we train both a linear and an RBF kernel on subsequences of the same length. As seen in Section 4.3.1, the linear kernel degenerates into a comparison of the means of two time series, we therefore expect bad predictive performance for it. On the other hand, the RBF kernel was already used in previous TSC work [205]. We would expect the RBF kernel to perform better than the linear as it capture non-linear patterns, which are critical in TSC. Nonetheless, the RBF kernel compares each pair of subsequences independently, while WTK captures similarities across the *entire distributions* of subsequences of two time series. Figure 4.8 summarises the accuracy results for all UCR data sets. Unsurprisingly, WTK outperforms the simple linear kernel in *all* cases. This confirms the meaninglessness of a straightforward application of the $\mathcal{R}$-convolution kernel to time series outlined in Sections 4.1 and 4.3.1. WTK also appears to be superior to the RBF kernel: the performance is better on all but twelve data sets. The accuracy difference for the points below the diagonal is however negligible, with an average difference in predictive performance of only $\approx$2.2 % for those data sets. This proves that the performance of our method is not achieved by considering the subsequences per se, but by considering the *distributions* of the subsequences instead.

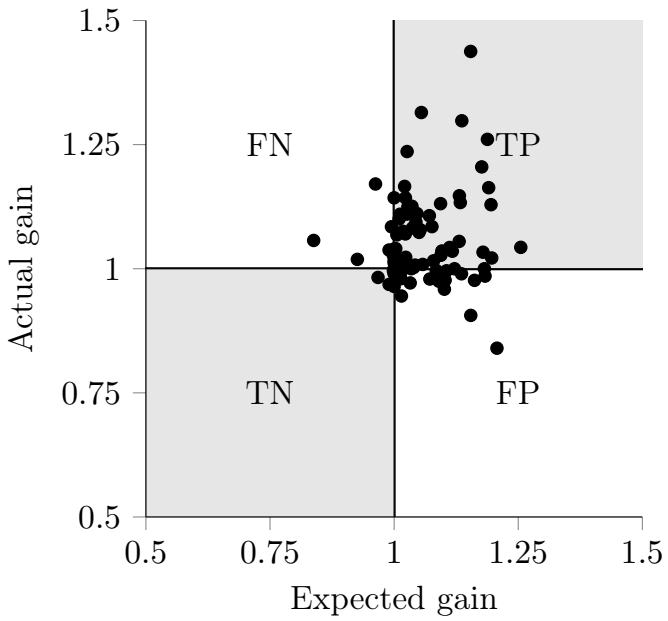


Figure 4.9: "Texas Sharpshooter" plot comparing the *expected* gains of our method WTK with the *actual* gains, relative to DTW-1-NN.

Comparison to DTW-1-NN. To follow best practices in TSC literature [60], we evaluate WTK against a strong DTW-1-NN baseline. The goal is to showcase the practical benefits of WTK by accounting for its potential performance in advance. The 'Texas Sharpshooter" plot [17] shown in Figure 4.9 presents the *expected* gain (estimated from the training data set) compared to the *actual* gain (calculated on the test data set). Most points fall into the TP or TN quadrants, implying that we correctly predicted that WTK would outperform DTW-1-NN (TP) or that it would be outperformed (TN). Points in the FN quadrant are good surprises: our method outperforms DTW-1-NN on the test set while we were expecting it would not. Points in the FP quadrant, on the other hand, are the problematic ones, however it only contains few points with minor accuracy differences. Therefore, since most points are in the upper right quadrant, the sharpshooter plot supports the claim that the proposed method is better than the DTW-1-NN baseline and highlights the fundamental consistency of the predictions of WTK.

Comparison to the state of the art Here, we compare WTK against the state of the art (SOTA) in TSC. We collected the accuracies of all published methods of the "UCR Time Series Classification Repository" [45], and two neural network baselines [247] with their classification performances from [71]. This resulted in 40 methods, nevertheless the availability of comparison partners depends on the data set (neural network baselines, for example, are not available for all of them). We then selected the best method for each data set (using the published train/test split results) and refer to it as the respective SOTA for that data set. In total, we there-

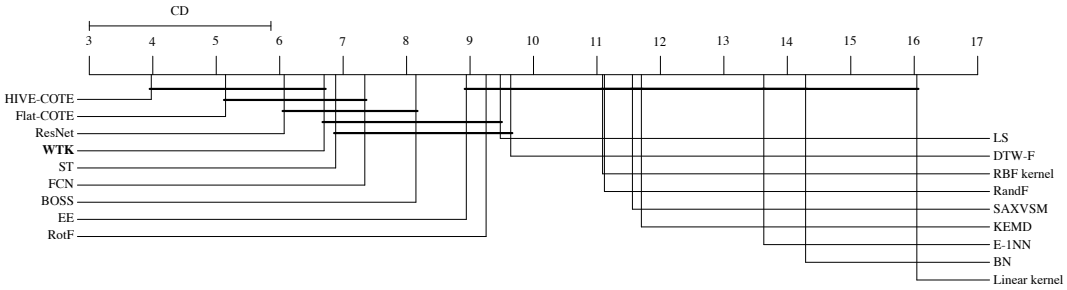


Figure 4.10: Critical difference plot comparing WTK (shown in bold) against multiple other methods. We observe that there is no statistically significant difference between the performance of our method and state-of-the-art ensemble methods.

fore compare our method against the best methods across 40 other method for each data set. WTK outperforms *all* SOTA methods on six data sets: DistalPhalanxTW, DistalPhalanxOutlineAgeGroup, MiddlePhalanxOutlineAgeGroup, Earthquakes, ECG5000, and FordB. Moreover, WTK has at least *equal* accuracy as any SOTA method on 12 data sets: BeetleFly, Coffee, ECGFiveDays, Plane, ShapeletSim, and Trace . Additionally, we are almost as good as the SOTA in many other data sets. Table 4.5 gives a better breakdown of the performance differences in comparison with HIVE-COTE, the best-performing ensemble method, and KEMD, a conceptually similar (due to its use of concepts from optimal transport) method. Each entry in the table shows the fraction of data sets for which the condition of the first column (absolute difference w.r.t. the SOTA performance) is respected. While we do not match the performance of HIVE-COTE, a heavy ensemble method, for the majority of data sets, our performance difference is less than 5 %. Moreover, we observe that the performance of KEMD is quire erratic, leading to favourable performance on some data sets while being completely outperformed on most of the others. Finally, we provide an in-depth comparison with three selected methods in Figure 4.8. We compare with HIVE-COTE (the overall best method), ResNet (the best deep neural network method), and KEMD (another method using notions of op-

Table 4.5: Absolute difference (Δ) in mean accuracy for three different methods with the respective SOTA method. Columns might not sum to 100 % due to rounding.

Δ	WTK	HIVE-COTE	KEMD
$\Delta \geq 0$	14.1 %	36.5 %	4.7 %
$0\% > \Delta \geq -5\%$	44.7 %	34.1 %	15.3 %
$-5\% > \Delta \geq -10\%$	24.7 %	18.8 %	7.1 %
$-10\% > \Delta \geq -15\%$	8.2 %	1.2 %	16.5 %
$-15\% > \Delta \geq -20\%$	4.7 %	7.1 %	9.4 %
$-20\% > \Delta$	3.5 %	2.4 %	47.1 %

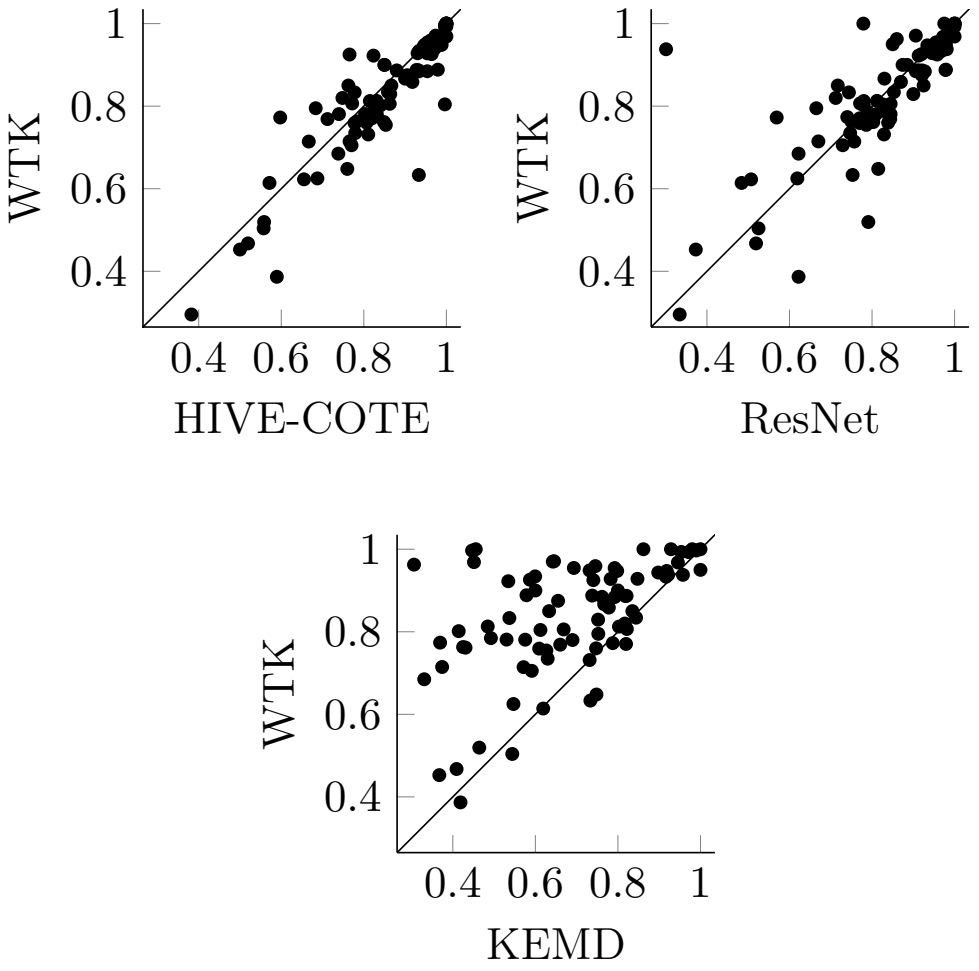


Figure 4.11: Comparison of our method WTK against selected other methods: following the critical difference plot from Figure 4.10, we chose the overall best method (HIVE-COTE), as well as the best deep neural network method (ResNet). Additionally, we also compare against KEMD because of the shared theoretical background it has with WTK. In each of the plots, every point corresponds to one data set, while the axes depict the accuracy of the respective method. We adjusted the axes to a range of $[0.4, 1.0]$ because no lower accuracies occured. In an ideal scenario, all points would be above the diagonal as this would mean that we outperform the respective comparison partner on *all* data sets.

timal transport). Our method follows the performance of ResNet closely and clearly outperforms KEMD.

Statistical analysis. To support our claims with statistical soundness, we obtained a *critical difference plot* [62], shown in Figure 4.10. It depicts the average rank of our method and the top comparison methods across all UCR datasets. The critical difference plot is obtained by running the Friedman test [82] to detect if rank distributions are significantly different across the methods. If this proves to be the case, the procedure then relies on post-hoc analyses such as the Nemenyi test [174] to determine the critical difference in average ranks that groups competing methods according to their performance and distinguishes groups of methods that offer significantly different performances. The procedure relies on multiple testing corrections to control the family-wise error rate and guarantees maximal power by limiting tests to pairwise tests with the new proposed method. The interested reader can refer to Demšar [62] for a more detailed description. For a significance level of $\alpha = 0.05$, the figure shows that there is no statistically significant difference in the performance of WTK and these best-performing classifiers [12]. This suggests good generalisation performance as our method is on par with heavily-parametrised classifiers such as neural networks or *ensembles* that comprise more than 30 methods.

Runtime considerations

Similar to what was done for WWL and graphs, we performed a brief analysis of the empirical runtime properties of WTK. Figure 4.12 confirms that the Wasserstein computation is not the driving factor in the asymptotic computational complexity since the linear kernel, the RBF kernel and WTK, all relying on differences between subsequences, perform the same asymptotically. KEMD [59] is faster since it does not extract subsequences. We also observe, as briefly mentioned in Section 4.3.2, that SoftWTK scales super-exponentially because it requires the computation of the permanent [56] of the differences between subsequences. Finally, we did not assess the impact of using Sinkhorn approximation for given length of time series but we expect it to have a similar behavior than the one observed with graphs. Namely, the speed up benefits only start to weigh in for a given number of subsequences (and therefore time series length).

Concluding remarks

We here introduced a new subsequence-based kernel for time series: the Wasserstein Time series Kernel (WTK). We showed that our method ourtperforms some of the state-of-the-art time series classification approaches while displaying favourable generalisation properties. WTK is a good way to avoid the meaninglessness of certain subsequence-based kernels applied to time series and confirms the appropriateness of Optimal Transport for machine learning applied to structured objects, as introduced in Section 4.1 and explored in Section 4.2.

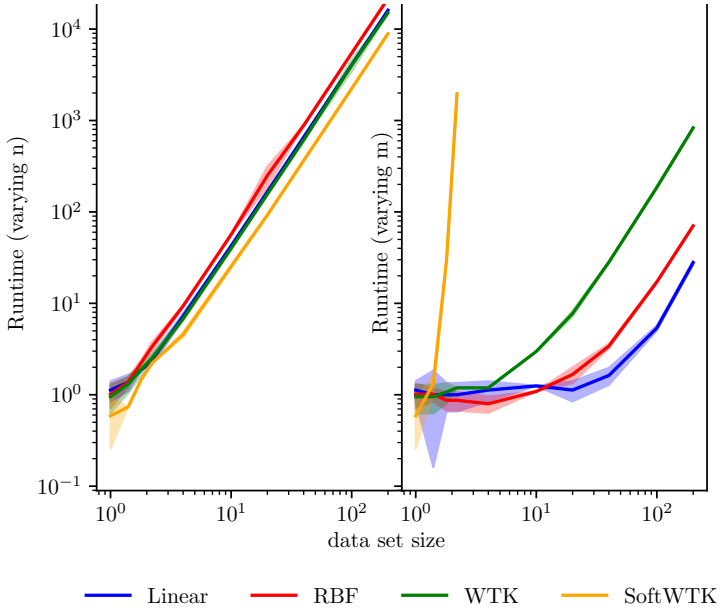


Figure 4.12: The empirical computational complexity of different subsequence kernels when scaling the dataset. n is the number of samples and m is the length of the time series. The y-axis shows running time normalised with respect to the shortest-running method.

More generally, the OT-based kernels presented in this chapter (WWL and WTK) solve the issues related to $\mathcal{R}$-Convolution mentioned in Section 4.1. We can therefore reliably use them to tackle challenges related to complex and structured objects in bioinformatics.

Part IV

Crop yield prediction

5 Deep learning for crop yield prediction

In which a new approach to accurately forecast wheat yield using several early field observations and genotypes is proposed.

One of the ultimate goals of genomics is to leverage genetic information to better understand phenotypic variations and guide decision making. In humans, this translates into improved diagnostics, patient segmentation, and therapeutic efforts [9], as we have also seen and discussed in Chapter 3. In plant research, this mostly corresponds to enhanced crop selection, as we will discuss here. In both cases, phenotype prediction from genotypes is the ultimate aspiration of many bioinformatics initiatives, albeit it has proven to be very difficult to do so well [136].

Identifying and growing crops that guarantee the highest product yield for a species is of utmost importance to guarantee appropriate and sustainable food supplies for the global population [53, 78]. Plant breeding programs benefit from increasing technological support but still rely on full growth cycle and manual yield measurement, hindering speed of development. While methods to predict yield from other measurements have been proposed, none has reached satisfying levels of performance.

In this chapter, we propose a new attention-based deep learning model that predicts wheat yield and leverages both genotype and observational data by fusing four sources of input data: multitemporal multispectral images, multitemporal thermal images, multitemporal digital elevation models, and single nucleotide polymorphism (SNP) measurements. Part of the presented content is based on the following unpublished manuscript:

- M. Togninalli, X. Wang, J. Poland, and K. Borgwardt. "Deep learning enables accurate grain yield prediction using image and genotype information". Unpublished Manuscript. 2020

The chapter is organised as follows. Section 5.1 introduces the topic of plant yield prediction and gives an overview of existing approaches. Section 5.2 outlines and motivates the method we propose to tackle the yield prediction problem. Finally, Section 5.3 summarises the outcomes and results of the experiments performed on wheat data.

5.1 Introduction

Phenotype prediction has always been a clear goal of genomic research. From the very first attempts [93, 219] to more recent approaches [136], the scientific community has tried to predict all kinds of phenotypes from genotype information alone. Nevertheless, it quickly became obvious that genotype alone cannot be used as a blueprint to explain all phenotypic differences. In fact, the initial GWAS efforts rapidly uncovered and described the so-called "missing heritability" problem [162]: even on large cohorts, the variability of individual genetic variations cannot explain *all* the variability of the heritable portion of observed phenotypes. A leading example can be found in the highly heritable *human height* trait. Visscher [242] reports that human height has an approximate empirical heritability of 80% but the 50 most associated loci together only account for 5% of the observed phenotypic variance [162, 242]. Several hypotheses of where this problem comes from have been put forward and some were validated, but a part of the heritability remains unexplained. Coming back to the previous example, Yang et al. [259] showed that accounting for *all* the common SNP variants can explain up to 45% of the observed phenotypic variance. Hence, accounting for all variants increases the explainability thanks to the weak effects of many variants. Another supposition is that genetic variants' effects are not solely additive but present an interactive nature: specific *combinations* of variants are causal for given phenotypes [163]. Finally, epigenetics is also seen as a potential source of heritability [85].

These considerations, however, are only related to the *heritable* portion of phenotypic variance. The remainder variability in phenotypic distributions is caused by environmental and developmental factors that cannot be explained by genetic factors [99]. Therefore, to fully be applicable and useful in practice, phenotype prediction also needs to account for these non-genetic factors. Most statistical genetics studies related to phenotype prediction either try to control the environmental conditions so as not to have to account for them [24] or design experiments to minimise the impact of the environment [188]. Another way to incorporate environmental variables in the predictive model is by using covariates that encompass or are linked to certain environmental factor. In humans, for example, this can be done by including covariates such as sex and age [122]. In their study, Khera et al. [122] develop risk predictors for Coronary Artery Disease and report an area under the receiver operating characteristic (ROC) curve of 0.81 when including sex and age, whereas the performance only reaches 0.64 when solely considering genetic variants [121].

Nevertheless, simply adding environment-related variables as new features in linear models can be limited, especially when considering that genotypic variations result in biological phenomena that can interact with said environment. Therefore, more complex models that capture the *interactions* between genotypes and environmental variables are better suited for phenotype prediction tasks. Moreover, the considered covariates in past studies are often still snapshots of the environment at a given point in time, but biological processes being highly dynamical, such static representations are not indicative enough. It would be better, when possible, to evaluate environ-

mental factors *throughout time* to better model the impact they have on the resulting phenotypes. Hence, while ways to mitigate or incorporate environmental factors in phenotype prediction models do exist, they are currently limited because (i) they fail to account for the interactions between environment and genetic background, and (ii) they do not consider the temporal nature of the environment.

When considering these limitations, one field of machine learning appears as an obvious candidate to tackle them. Deep learning [91] has proven to be an extremely versatile framework for both unsupervised and supervised learning with a myriad of data types. With clear successes in computer vision [102], natural language processing [237], time series forecasting [247], sequence modeling [267], and graph-based machine learning [124], deep learning has affected all areas of applied and theoretical machine learning [200]. It relies on artificial neural networks with several layers to extract features from the input that are relevant for the output and leverages *non-linearities* to capture higher degree interactions from the input features. More recently, *attention* mechanisms were shown to clearly outperform competing methods when dealing with sequential data [237]; these mechanisms allow models to learn meaningful combinations of intermediate representations. Deep learning equipped with attention mechanisms could therefore address both concerns (i) and (ii) enumerated above and efficiently capture the interactive nature of genetic variants during training.

Here we leverage attention mechanisms and convolutional neural networks to *fuse* data from several channels. We leverage a vast agricultural data set comprising measurements acquired throughout time from different sources to accurately predict wheat grain yield. We develop a deep learning model that is efficiently able to combine genetic information with rich phenotypic observations reflecting the environment and its effects *over time* on the predicted phenotype. We now present in further details the task of interest.

5.1.1 Crop yield prediction

Food security is a critical problem that recently attracted considerable interest due to the recent global population growth [78]. In order to guarantee appropriate food supplies, it is of utmost importance to identify and grow crops that guarantee the highest product yield for each species. To that end, plant breeding programs are designed to identify the crossings that guarantee the highest yield while ensuring resistance and resilience [53]. Over the years, several technological tools such as high-throughput phenotyping and genomic selection have been developed to help breeders identify the most promising candidates [227]. Nevertheless, plant scientists still rely on end-point destructive measurements such as grain yield measurement to rank and select the best candidates. Therefore, along the technological improvements above, being able to predict yield from simple aerial images during the growth stages would drastically reduce the effort needed in the selection process.

Since being able to predict crop yield is also important for food security monitoring and policy-making [180], several approaches have been proposed. Methods range

from ground-based field surveys or farmers' expert knowledge to growth models and remote sensing data [47, 245]. The most discussed source of data is remote sensing data, for they are easy to collect and access. Satellite data with various spatial, temporal, and spectral resolutions have been used over a large range of geographic areas and scales [18]. However, the coarse spatial resolution of satellite imagery motivated the development of sensor technologies embarked in low altitude Unmanned Aerial Vehicles (UAV) [50]. This led to improved spectral, spatial, and temporal resolutions and considerable cost benefits for high-throughput phenotyping [98]. In turn, many crop yield prediction models based on UAV-acquired imagery were developed for a multitude of crops [66, 90, 164].

Typical setups of UAV-based phenotyping include multispectral cameras and thermal imagery [90, 95]. Most crop yield predictors rely on manually crafted features (vegetation indices, VIs), which are believed to efficiently summarise vegetation growth related information. The Normalised Difference Vegetation Index (NDVI) [191] is a prime example of such features. Despite the efforts put in crafting these manual features, they oftentimes solely rely on two bands of the multispectral images available, discarding valuable information about the remainder of the signal. Several machine learning models have been evaluated for crop yield predictions: self-organizing maps to account for different soil types [179], Support Vector Machines [220], ensemble tree methods [103], statistical linear models [253], or artificial neural networks [139]. However, most of these approaches rely on simple summary statistics of the imagery values recorded by the UAV [112]: by relying on means and standard deviations of VIs, models discard all information related to high-order moments.

More recent approaches leveraged the potential of high-dimensional image data using deep learning techniques. You et al. [263] use convolutional neural networks (CNNs) and recurrent neural networks (RNNs) in combination with a Gaussian Process to model the spatio-temporal structure of the collected data. However, they still only used histograms of the collected images and are limited to evenly-spaced acquisition periods, i.e. the time difference between acquisition must be constant, which in practice is not a very practical assumption. Maimaitijiang et al. [158] propose a deep learning model to fuse information from multispectral and thermal images, but can only use single snapshots of information and cannot capture temporal evolution of the environmental parameters. Finally, Khaki et al. [120] propose a CNN-RNN model to combine satellite imagery with weather and soil data and models the temporal evolution across the years via a long short-term memory (LSTM) model. Nonetheless, such a model could not scale to more granular data (e.g. UAV) and is also limited to evenly-spaced acquisition periods. Moreover, *none* of the above-mentioned approaches allow the incorporation of genetic information in their predictions.

To counter the above mentioned limitations, we provide a flexible model that leverages attention mechanisms to comprehensively take into consideration four sources of information: multitemporal multispectral images, multitemporal thermal images, multitemporal digital elevation models, and SNP measurements. We suggest to leverage images captured by UAVs across time at different temporal resolutions for

plot-level yield prediction. Instead of using deep learning architectures specifically designed for time-series data such as recurrent neural networks, we approach the problem from a Multiple Instance Learning (MIL) perspective: for each sample (the plot in our cases) a set of multiple data points denoted as *instances* (images of the plot at different times and from different angles) are associated with the same label (the measured yield for that plot). We will now present the relevant background and introduce our method.

5.2 Multiple Instance Learning for phenotype prediction

This section details the necessary background on Multiple Instance Learning and deep representation learning and introduces our proposed method.

5.2.1 Background

Deep learning models

Deep learning models are complex non-linear functions that learn some target function $f*$ by updating their many parameters via backpropagation of gradients with respect to a defined loss function. Most deep learning models are called networks because they are obtained by composing together several functions (i.e. layers). The functions are *chained*: this way, the output of a function is the input of the following function. Many architecture, combining these functions in diverse ways, have been proposed to tackle all types of learning tasks: regular Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) are typical examples [133]. We here present some basic concepts from the deep learning literature. Our primer is by no means exhaustive, for a complete introduction to deep learning refer to the book by Goodfellow et al. [91].

The simplest function used in deep learning networks is the *fully connected* layer, which takes a vector $\boldsymbol{x} \in \mathbb{R}^n$ as an input, applies a linear transformation via a weight matrix $W \in \mathbb{R}^{m \times n}$ and a bias vector $\boldsymbol{b} \in \mathbb{R}^m$ followed by a non-linear *activation* function $a(\cdot)$ to output a vector $\boldsymbol{y} \in \mathbb{R}^m$:

$$\boldsymbol{y} = a(W\boldsymbol{x} + \boldsymbol{b}). \tag{5.1}$$

Chaining several fully connected layers leads to DNN, also known as fully connected networks (FCN) or multi-layer perceptrons (MLP): $f(x) = f^{(3)}(f^{(2)}(f^{(1)}(\boldsymbol{x})))$. The learning procedure then takes the final output $f(\boldsymbol{x})$ and compares it against a ground truth label to obtain a loss value $\mathcal{L}$. From there, the gradients with respect to the loss for each parameter in the network (i.e. $W^{(1)}$, $W^{(2)}$, $W^{(3)}$ in the example above) are taken leveraging the chained nature of $f(\boldsymbol{x})$ and used to update the parameters via a gradient descent approach.

Convolutional neural networks are typically composed of three types of layers: convolutional, pooling, and fully connected layers. The main idea of convolutional layers is that they share weights across the first two input dimensions and therefore reduce the number of parameters to learn. They were designed to process data with a known grid-like topology (e.g. images with their 2D grid of pixels). The layer takes as an input a matrix $X \in \mathbb{R}^{h \times w}$, applies a convolution using weight matrix $W \in \mathbb{R}^{l \times l}$ and adds a bias term $B \in \mathbb{R}^{p \times q}$ before applying a non-linear activation function $a(\cdot)$ and outputting a matrix $Y \in \mathbb{R}^{p \times q}$:

$$Y = a(W * X + B), \tag{5.2}$$

where "$*$" is the convolution operator. Intuitively, the convolution can be seen as a sliding window of size $l \times l$ that linearly combines values from x locally at every location of the matrix. This is usually followed by a pooling layers that aggregate the values of the convolutional layer's output to reduce its dimensionality. Typical pooling operations include max-pooling or mean-pooling. Here too, one can imagine this as an operation to extract local summary statistics from the input x at different locations. In addition to that, more recent CNN models also include skip-connections, which enable to skip given layers in order to mimic the biological behavior of pyramidal cells. These architectures, known as Residual Networks (ResNet) have proven to be very efficient in extracting features for image-related tasks [102].

Lastly, recurrent neural networks are oftentimes used to deal with temporal data in forecasting scenarios [104]. Nevertheless, RNNs were recently shown to be outperformed on time series tasks by using temporal convolutional networks, variants of CNNs [132]. Moreover, these techniques are devised for *evenly spaced* and *aligned* time series and offer weak performances on unaligned and irregularly spaced time series [106]. Horn et al. [106] also showed that permutation-invariant aggregation of the time-steps in a time series results in good classification performance when integrating the time-stamp of each observation in its vectorial representation, indicating that permutation-invariant functions can be applied to learn on time series.

Overall, deep learning models excel in representation learning tasks, where the goal is to learn meaningful representations of input data to be used for downstream tasks. This is possible thanks to the end-to-end training capabilities unlocked by the deep learning framework: the error signal obtained from the loss function $\mathcal{L}$ can be used to update *all* parameters used to reach the final prediction. Deep learning layers and models are therefore regularly combined to fit any learning task.

Multiple Instance Learning

Multiple Instance Learning (MIL) aims at learning a target value from a sample that is a *bag* of instances. Classically, the MIL problem was solved by combining the outputs of an instance-level model run on each of the instances for the sample [195]. Alternatively, single instances can be embedded to low-dimensional representations and fed to a set function predictor [7]. Along similar lines and more recently, researchers

adapted deep learning pipelines to the MIL problem by using neural networks to learn useful low-dimensional representations of the single instances and then leveraging attention mechanisms to aggregate the different elements' contributions [110]. More formally, for one target variable $y \in \mathbb{R}$, instead of a single associated instance $\boldsymbol{x} \in \mathbb{R}^n$ we have an associated *bag* of instances $X = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_k}\}$ that do not exhibit a particular ordering between each other.

The learning problem then aims to learn $\hat{y} = S(X)$, where S is a function that is permutation-invariant to the elements in X (i.e. the output of S is not influenced by the ordering of $\boldsymbol{x_1}, \ldots, \boldsymbol{x_k}$). Zaheer et al. [264] show that such a function needs to be decomposable in a sum of transformations as follows.

Theorem 5 (Zaheer et al. [264])**.** *A function $S(X)$ for a set of instances X having countable elements is a valid set function (i.e. permutation-invariant to the elements of X), iff it can be decomposed in the form*

$$S(X) = g(\sum_{\boldsymbol{x} \in X} f(\boldsymbol{x})) \tag{5.3}$$

where g and f are suitable transformations.

This allows us to view the MIL problem as a three-step process: (i) transform the instances with a function f, (ii) combine the transformed instances with a permutation invariant function ϕ (e.g. sum, average) (iii) transform the combined instances with a function g. In other terms, we obtain an *embedding* for each instance via f, combine them in an invariant manner with the pooling operator ϕ and rely on g to get a useful output for the learning task at hand.

This process can be translated to a deep learning setting, where both f and g are neural networks and ϕ needs to be a differentiable pooling operator. Common pooling operators in deep learning include the maximum operator and the mean operator [264]. Nevertheless, these pooling opeartors have the disadvantage of being pre-defined and non-trainable. That is why we prefer an attention-based pooling mechanism [110], which offers a higher flexibility to the data and the tackled task as well as a certain degree of *interpretability* of the pooling.

Attention mechanisms have been extensively used in natural language processing [237], image captioning [256] and graph neural networks [238]. Broadly speaking, attention mechanisms are neural networks' components in charge of quantifying the interdependence of input elements (i.e. weight the contributions of each input based on the input itself and on other inputs). This translates in finding weights $a_i = f(\boldsymbol{h_1}, \ldots, \boldsymbol{h_k}) \;\; \forall i = 0, \ldots, k$, where $H = \{\boldsymbol{h_1}, \ldots, \boldsymbol{h_k}\}$ is the set of inputs. In their simplest form, they can be a simple inner product between the inputs $\boldsymbol{h_i}$. More advanced attention mechanisms are composed by a neural network that learns the relative importance of each input element. Ilse et al. [110] proposed a MIL pooling

of the sort. Let $H = \{\boldsymbol{h_1}, \ldots, \boldsymbol{h_k}\}$ be the set of k embeddings of dimension m for a sample $X = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_k}\}$, then the pooling operation is given by:

$$\boldsymbol{z} = \sum_{i=1}^{k} a_i \boldsymbol{h_i}, \tag{5.4}$$

with

$$a_i = softmax(\boldsymbol{w}^\top \tanh(VH^\top))_i = \frac{e^{\boldsymbol{w}^\top \tanh(V\boldsymbol{h_i}^\top)}}{\sum_{j=1}^{k} e^{\boldsymbol{w}^\top \tanh(V\boldsymbol{h_j}^\top)}}, \tag{5.5}$$

where $\boldsymbol{w} \in \mathbb{R}^{l \times 1}$ and $V \in \mathbb{R}^{l \times m}$. This pooling allows for more flexibility in the way the contribution of individual instances are combined, unlocking better prediction performance. Additionally, the weights $a_1, \ldots, a_k$ can be used to gauge the relative importance of each instance of the sample and provide interpretability around the model's prediction.

5.2.2 Data fusion with attention-based MIL

Considering the problem described in Section 5.1.1, we wish to predict the wheat grain yield using multiple data sources. It is therefore necessary to combine them meaningfully. The different "views" - multispectral images, thermal images, digital elevation models, and genotypes in our case - are all linked to the same final grain yield. Moreover, we have multiple instances (i.e. images) of the same plot for the same view. With the background collected in the previous section, we can envision a MIL-based method that meets these criteria.

We base our approach on the work of Ilse et al. [110] and extend it to fuse multispectral and thermal images, temporal, spatial, and genomic information. We therefore treat each observation of a given plot as an *instance* of the object we aim to predict yield for. We rely on a DNN to encode the genotypic information and on ResNet architectures to encode the different images we have for each plots. We then combine the obtained representations in a permutation-invariant MIL setup, that allows for efficient aggregation of data from diverse sources across time. We will therefore obtain embeddings for each instance (i.e. data source) we are dealing with and we will combine them using Equations 5.4 and 5.5. Moreover, since some of our data sources consist in multiple, irregularly-sampled observations through time and since we have also seen in Section 5.2.1 that permutation-invariant aggregation can be efficiently used to learn on time series with irregularly spaced observation, we will apply the attention pooling framework along the temporal dimension too.

More practically, for each *sample*, i.e. plot of wheat, we have a final yield value and four data sources that need to be combined: multispectral images, thermal images, digital elevation models (DEM), and SNP array data. Moreover, for the first three data sources, each plot has multiple observations through time: from early images at the beginning of the growth process to images just before harvest (see Section 5.3.1 for details). We therefore rely on a deep learning architecture that (i) takes each

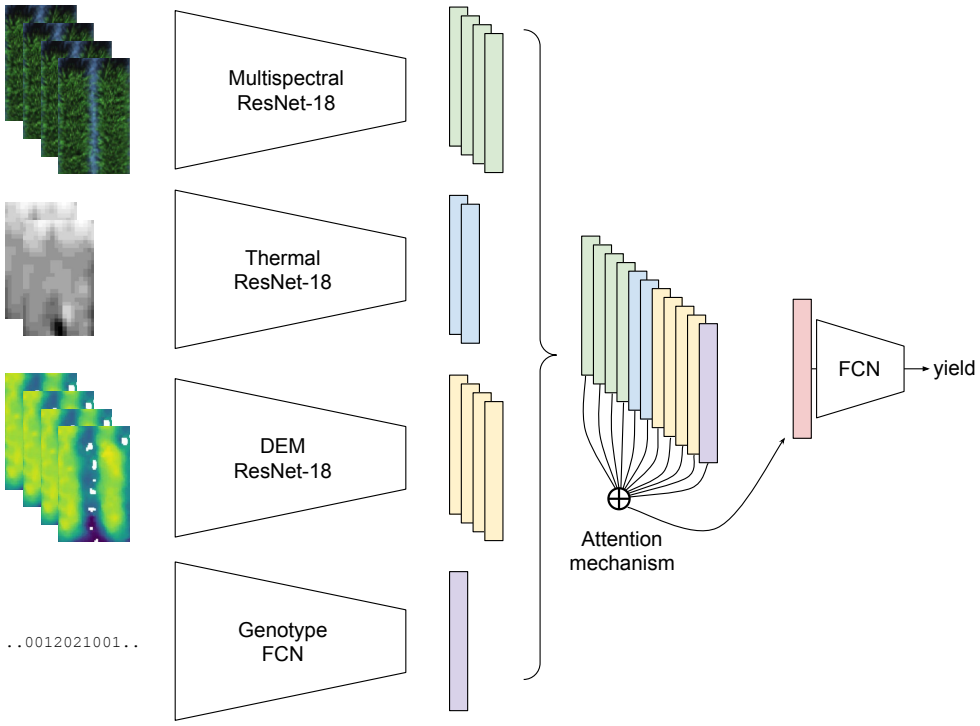


Figure 5.1: Schematic view of the Multiple Instance Learning (MIL) model used for the wheat grain yield prediction. For each sample (i.e. plot), data from four sources are combined: multitemporal *multispectral* images, multitemporal *thermal* images, multitemporal digital elevation models (*DEM*), and *genotype* data (SNP array). Each data instance is transformed into an embedding via a dedicated model (ResNet-18 for images and a simple fully-connected network - FCN - for genotype data) and the attention mechanisms combines the embeddings into a single embedding which is passed to a final fully-connected network for yield prediction.

instance of every data source and transforms it into a fixed-size embedding via a data source-specific encoder, (ii) combines the obtained embeddings into a unique vectorial representation of the sample (iii) computes the predicted grain yield for that sample. Figure 5.1 summarises the architecture in a schematic view. The model can then be trained in an *end-to-end* fashion and learn the weights for the encoding, the pooling, and the prediction networks. To do so, we use a mean squared error (MSE) loss between the measured yield y and the predicted yield $\hat{y}$: $\mathcal{L}_{MSE} = \sum_{i=1}^{N} (\hat{y}_i - y_i)^2 / N$. Moreover, since the number of instances for each plot and data source is not constant, we can easily handle samples with less or more images as well as with missing data sources.

5.2.3 Implementation

We implemented our method in a flexible manner to be able to add and remove data sources easily. We rely on the PyTorch library [183] to implement the model architecture, on PyTorch Lightning wrapper to speed up experiments [70], and make our code available on GitHub.

For image-based data channels, we rely on a small residual network architecture (ResNet-18 [102]) given the relative simplicity of the images (small size, see Section 5.3.1). For genotypic information (SNPs) we use a FCN with two layers of 1024 and 512 hidden units respectively. We then force all embedding representations to a 256-dimensional vector and combine them using the attention mechanism described in Section 5.2.2. We employ a multi-head attention mechanism with n heads, meaning that we have n different combined vectors that we then concatenate and pass through the final fully connected layer of the model for the final regression.

We also implement additional encoding mechanisms. We experiment with a temporal encoding to leverage the datestamp of each image acquired by embedding the dates as one-hot vectors and appending those to the embeddings generated by the residual convolutional networks. In a similar fashion we test channels encoding where we append a one-hot encoding of the channel to the 256-dimensional embeddings, to see if nudging the attention mechanism by indicating which data sources it is dealing with is helping.

Due to memory constraints on the computing infrastructure, we cannot use *all* images for each instance (some instances have up to 200 images). We therefore add a parameter denoted as *bag size* which indicates a maximum number of images to randomly sample for each channel at every iteration. This means that throughout training, the set of chosen images for samples with many input images constantly changes. We then tune this hyperparameter with other ones in our setup (see Experimental design in Section 5.3.2).

5.3 Experimental results

After introducing our approach, we assess its performance in practice. This section contains all the details about the performed experiments and used data set. We first present the data set, then present the performance of the model on the grain yield prediction task before investigating the attention mechanism of the model to understand what drives the model predictions.

5.3.1 Data set

The studied data set regroups multisensor data acquired by collaborators at Kansas State University. The detailed methodology can be found in Singh et al. [217], we report here the most relevant details.

Plant material and field layout. Advanced spring wheat (*Triticum aestivum L.*) breeding lines from the International Maize and Wheat Improvement Center (CIM-

MYT) breeding program were sown on November 21, 2017 during season 2017–2018. The experiment consisted of 9600 unique spring wheat entries distributed in 320 trials arranged following an α-lattice design in two blocks with plot size of 1.7×3.4 m^2. Final crop yield was measured in tons per hectare (t/ha) on a per plot basis.

Data acquisition. A DJI Matrice 100 (DJI, Shenzhen, China) UAS was used for data acquisition, it was equipped with a 5-channel multispectral RedEdge camers (MicaSense Inc., United States) with blue (475 nm), green (560 nm), red (668 nm), RedEdge (717 nm), and near infrared (840 nm) bands. Flights were conducted between 11AM and 1PM relying on the procedures developed by the Poland Lab [246] at a ground altitude of 35m. To ensure highly accurate data, the acquired images were geo-referenced and geo-rectified using 12 colored ground control points (GCPs) uniformly distributed across the field area. To collect thermal images, a FLIR VUE Pro R thermal camera (FLIR Systems, USA) was carried by the DJI Matrice 100 and flights were performed at 60m above the ground. All lines were profiled using the genotyping-by-sequencing protocol of Poland et al. [188] and sequenced on an Illumina Hi Seq2000 or HISeq2500. Single nucleotide polymorphism (SNP) markers were aligned to the reference Chinese Spring Wheat Assembly v1.0 [52]. Genotyping calls were extracted and filtered so that the percent missing data per marker was less than 40% and percent heterozygosity was less than 10%. Lines with more than 50% missing data were removed.

Table 5.1: Details of the data set after quality control and filtering.

Plants	# plots	19,161
	# trials	320
	# entries	9,596
	# genetically unique entries	8,931
Multispectral images	# images	804,546
	# unique dates	14
	Avg # images per plot per date	8.36
Thermal images	# images	1,386,679
	# unique dates	4
	Avg # images per plot per date	36.18
DEM images	# images	96,358
	# unique dates	14
	Avg # images per plot per date	1.00
Genotypes	# typed SNPs	38,361

Data processing To obtain the images for the multispectral and thermal images, the acquired pictures were orthorectified and each pixel was mapped to its exact geographical location using the GCPs via a Python pipeline available on GitHub. Finally, the plot-level image extraction was performed by cropping single-plot images out of the larger orthorectified images. To obtain digital elevation models (DEM) for

each plot, images were processed using Agisoft PhotoScan Pro (Agisoft LLC, Russia) and the protocols of the lab to extract the height estimates from the multispectral images. DEM were then treated as single-channel images by the model. Each image source (multispectral, thermal, and DEM) was further preprocessed to optimise for subsequent model learning. For each plot, all images across time and source were regrouped, their size was homogenised and they were stored in a single file. The final size for images were 128×128 pixels for multispectral and DEM images and 40×40 pixels for thermal images. For the SNP data, after filtering, a total of 38,361 SNP markers were retained and missing data were imputed with Beagle v4.1 [38]. Of the original entries, some were left out due to missing data in any of the channels. Finally, genotyped SNP values were standardised by subtracting the mean and dividing by the variance of the whole training set on every SNP. Table 5.1 summarises the main characteristic of the curated data set.

5.3.2 Phenotype prediction

The goal of the prediction task is to accurately predict final grain yield in tons per hectare (t/ha). We first detail the experimental design and then summarise the performance obtained with an increasing number of data sources.

Experimental design

For all phenotype prediction tasks, we do a 5-fold cross-validation on the plot ids, stratifying by trial. This ensures that there are no replicates that can be both in the training and in the test set. We then split the training set using the same stratification and obtain a validation set for the deep learning models. We use 80% of the data for training, 10% for validation, and 10% for testing. To guarantee comparability, the baseline models that do not require validation data can use it as training data, the test set are therefore the same for all compared methods and split.

We compare against several linear and non-linear baseline models: linear regression, lasso (L_1-regularised regression) [228], ridge regression (L_2-regularised regression) [201], elastic net (L_1 and L_2-regularised regression) [269], random forest [35], and gradient boosting [81]. Hyperparameters for these baseline models are tuned via internal cross-validation on the training set (90% of the dataset) using the dedicated scikit-learn python library [185].

The high-dimensionality of the images impedes their direct usage in the baseline models, we therefore extract moments (mean and mode) of individual channels as well as of manually crafted vegetation indices for each image. The vegetation indices considered are the normalised difference vegetation index (NDVI), the normalised difference red edge index (NDRE), and the green normalised difference vegetation index (GNDVI), and they are obtained as follows:

$$\text{NDVI} = \frac{\text{NIR} - \text{Red}}{\text{NIR} + \text{Red}}, \quad \text{NDRE} = \frac{\text{NIR} - \text{RedEdge}}{\text{NIR} + \text{RedEdge}}, \quad \text{GNDVI} = \frac{\text{NIR} - \text{Green}}{\text{NIR} + \text{Green}}, \tag{5.6}$$

where Red, RedEdge, NIR, and Green are the channels captured by the multispectral camera. Moreover, the baselines models are not capable of handling multiple instance input for the image channels: they can only handle a fixed-size input and concatenation of values across dates and instances is not possible, as the number of images per plot constantly changes. To tackle this, we average the above-mentioned values across images of a given plot. We do this both on a date-basis, where we group dates in 4 temporal groups: 1: 18.01.2018 - 31.01.2018, 2: 01.02.2018 - 02.03.2018, 3: 03.03.2018 - 10.03.2018, 4: 11.03.2018 - 21.03.2018. We then train the baseline model using individual date group values or using all the values combined. Each sample therefore has 16 features (2 x 5 channels and 2 x 3 VIs) for a given date group and 64 in the case of training with *all* dates.

For the MIL model, we fine-tune the hyperparameters via a random search of 20 runs on a split using the validation's Pearson's correlation coefficient to select the best set of parameters. The tuned hyperparameters are:

- Learning rate: the initial learning rate, chosen among $\{10^{-5}, 10^{-4}, 10^{-3}\}$
- Learning rate scheduling: this parameter allows us to have scheduled changes in the learning rate throughout training. This has proven to improve training, we try to have no scheduling, a plateau scheme which reduces the learning rate once learning stagnates or a cyclic cosine annealing scheme [151] chosen among $\{none, plateau, cosine\}$
- Batch size: the amount of samples processed in parallel, chosen among $\{8, 16\}$
- Bag size: the maximum number of processed images for a given channel, chosen among $\{8, 16, 32\}$
- Number of attention heads: the number of attention mechanisms used in parallel, chosen among $\{1, 4, 8\}$
- Temporal encoding: whether temporal encoding as described in Section 5.2.3 is performed, chosen among $\{True, False\}$
- Channel encoding: whether channel encoding as described in Section 5.2.3 is performed, chosen among $\{True, False\}$

We evaluate the regression performance by evaluating mean absolute error (MAE), mean squared error (MSE), Pearson's correlation coefficient, and the coefficient of determination R^2. We report the Pearson's correlation score on plots and refer to other values in tables.

All experiments were run on a dedicated cluster running Ubuntu 14.04.5 LTS, with 16 CPUs (Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz) each with 8 cores and 24 threads, 128 GB of RAM, and 8 GPUs.

Prediction with multispectral images

In a first step, we wish to assess the performance of the model with respect to baselines relying on traditional, manually crafted vegetation indices (VIs). To do so,

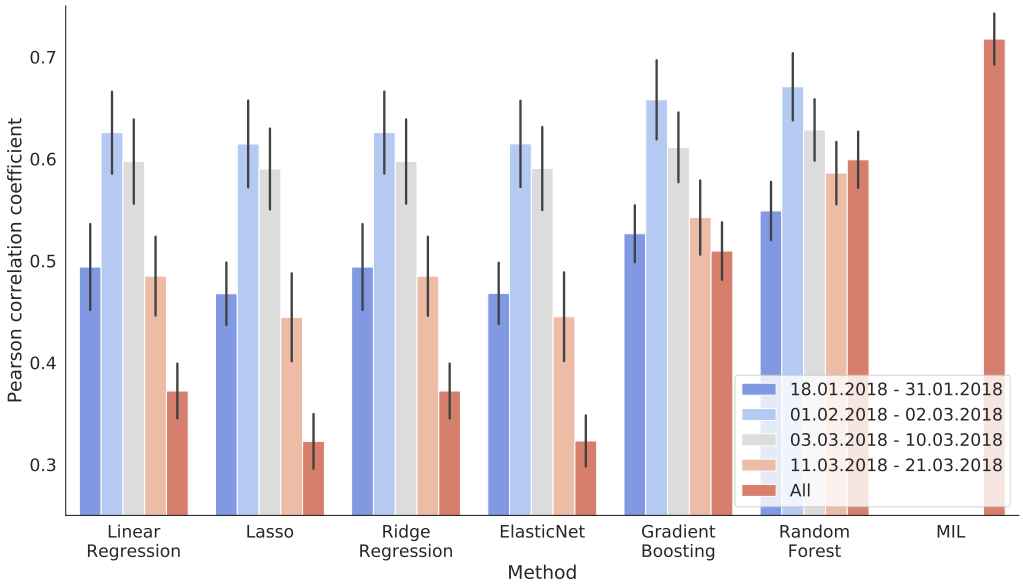


Figure 5.2: Pearson's correlation coefficient for linear and non-linear baselines on wheat yield prediction using multispectral images as well as for a MIL approach. Each baseline relies on VIs values aggregated on a time window (different colors) or on a combination of the aggregated values ("All" bar). Error bars indicate standard deviation across splits.

we only consider multispectral images as input. As highlighted above, the considered baseline cannot use multiple images as input, therefore we need to use summary statistics of the data at hand. Moreover, values are aggregated in groups of dates to capture the relevant properties at different growth stages. Figure 5.2 reports the performance of the baselines with respect to our MIL approach.

Moreover, a detailed breakdown of the performance of the MIL model and the baseline models relying on data from 01.02.2018 to 02.03.2018 can be found in Table 5.2. The MIL approach clearly outperforms the baselines. This can be attributed to two phenomena:

(i) First, the deep learning framework enables the use of the entire images instead of simple moments of their pixel value distributions. It can therefore better capture *non-linear* relationship between channels and neighboring pixels. This is also corroborated by the better performance of gradient boosting and random forest, two non-linear methods, with respect to the linear baselines.

(ii) Second, the multiple instance learning framework combined with the attention mechanism allows the efficient capture of inter-relationships between images of the same plot across observation points and time: with an average of 42 multispectral images per plot, a lot of information can be lost with simplistic aggregations such as averaging.

Table 5.2: Detailed results on wheat yield prediction for baselines and MIL model on multispectral images alone (mean ± std).

Method	MAE	MSE	R^2	Pearson Coeff.
Linear Regression	0.416 ± 0.016	0.293 ± 0.024	0.387 ± 0.054	0.626 ± 0.045
Lasso	0.422 ± 0.016	0.301 ± 0.025	0.371 ± 0.053	0.615 ± 0.048
Ridge Regression	0.416 ± 0.016	0.293 ± 0.024	0.387 ± 0.054	0.626 ± 0.045
ElasticNet	0.422 ± 0.016	0.301 ± 0.025	0.370 ± 0.052	0.615 ± 0.047
Gradient Boosting	0.402 ± 0.015	0.273 ± 0.022	0.428 ± 0.052	0.658 ± 0.043
Random Forest	0.395 ± 0.015	0.264 ± 0.022	0.447 ± 0.047	0.671 ± 0.037
MIL	$\mathbf{0.372 \pm 0.012}$	$\mathbf{0.237 \pm 0.019}$	$\mathbf{0.507 \pm 0.045}$	$\mathbf{0.717 \pm 0.028}$

We can therefore conclude that MIL can be efficiently used to combine multiple observations of images *across time*.

Wasserstein kernels on multispectral images. Since we are interested in two linked sources of data (i.e. multispectral images and genotypes), we wish to investigate the similarities between them. Images can also be considered as structured objects, we therefore apply the Wasserstein kernel framework presented in Chapter 4 to obtain similarity measures between each plot and compare these similarities with the underlying genotypic relationships. To do so, we transform every plot-related set of multispectral images in a high-dimensional histogram and compute the Wasserstein distance between them. We then apply the Laplacian kernel to obtain a similarity measure. A hierarchical clustering model uses these similarities to group the $19,161$ unique plots in $8,931$ clusters, which is the number of unique genotypes in the data set, resulting in $4,030$ clusters composed of a single plot and $4,901$ clusters with more than one plot. Out of those, only 323 ($\sim 6.6\%$) contain multiple instances of the same genetic replicate. This indicates that the variability captured in the multispectral images is different from the one given by the genotypic information and combining these sources would be beneficial for prediction. We will therefore now look at the performance of our model once genotype data is included. It is worth noting that we also apply our kernel to predict phenotype from multispectral images alone by using Support Vector Regression but do not reach satisfactory performance (Pearson's correlation coefficient of 0.562 ± 0.038). This is due to the simplification assumptions made during the histogram creation: instead of simple histograms, one should prefer images *signatures* [204], which cluster the pixel values of each image and get more meaningful representations of the images. However, this could not be performed on our data set, considered the large amount of images ($804,546$ multispectral images).

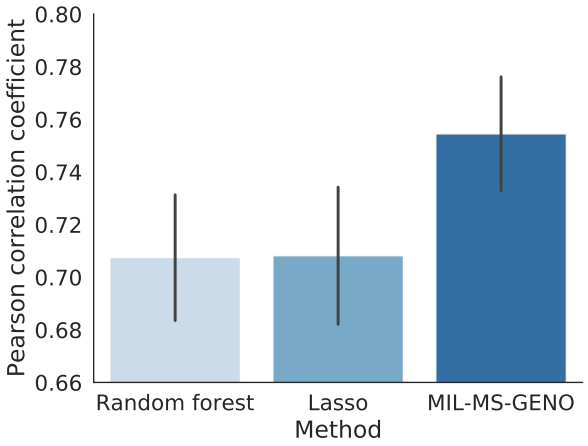


Figure 5.3: Pearson's correlation coefficient on wheat yield prediction for approaches combining genotype data and multispectral images. Error bars indicate standard deviation across splits.

Prediction with multispectral images and genotypes

For the baseline model comparison, since the best performance for images alone was achieved with the second date group (01.02.2018 - 02.03.2018), we only keep the summary statistics for the multispectral images for those dates and concatenated them with the normalised genotype values. Moreover, due to the very high dimensionality of the SNP array data (38,361 features) we only consider the Lasso regression and the random forest baselines. Lasso's L_1 regularisation acts as a feature selector and is the go-to model in linear phenotype predictors from genotypic data and we keep random forest as a non-linear baseline.

For the MIL model, on the other hand, we used the attention-based aggregation to combine the embeddings from MIL images across all dates together with the one obtained from the genotype FCN. Figure 5.3 and Table 5.3 present the results of this comparison. Here again, the MIL approach outperforms the linear and non-linear baseline by a considerable margin. We will now integrate the data from all 4 channels and assess the impact on performance before investigating the attention mechanism.

Furthermore, these results, when compared to the performance obtained by genotype-only based models (Pearson's correlation coefficients of 0.559 ± 0.050 for Lasso, 0.335 ± 0.046 for a simple FCN), also confirm that phenotype prediction can be greatly improved when incorporating covariates that model the environmental effects on the samples.

Prediction from all channels

Combining information from all channels is expected to improve the predictive performance of the algorithm: both thermal images and digital elevation models have

Table 5.3: Detailed results on wheat yield prediction for baselines and MIL model on multispectral images and SNP array data (mean ± std).

Method	MAE	MSE	R^2	Pearson Coeff.
Lasso	0.371 ± 0.008	0.240 ± 0.015	0.498 ± 0.040	0.708 ± 0.029
Random Forest	0.372 ± 0.011	0.241 ± 0.016	0.496 ± 0.035	0.707 ± 0.027
MIL	$\mathbf{0.345 \pm 0.008}$	$\mathbf{0.210 \pm 0.017}$	$\mathbf{0.563 \pm 0.038}$	$\mathbf{0.754 \pm 0.024}$

been shown to be partially predictive of the final yield of the plots [217]. This is confirmed in Figure 5.4 and Table 5.4, where the performance of the model slightly improves with the additional data provided across channels. The largest impact seems to be provided by adding genotypic information and the gain seems to saturate: the addition of DEM and thermal images improves the performance only slightly. DEM images are obtained from the multispectral images directly, it seems therefore plausible that part of their information content is already captured by the model when looking at the multispectral images. Nonetheless, the addition of image channels diminishes the variability of the performance, potentially indicating that the different sources corroborate each other and increase the model's certainty.

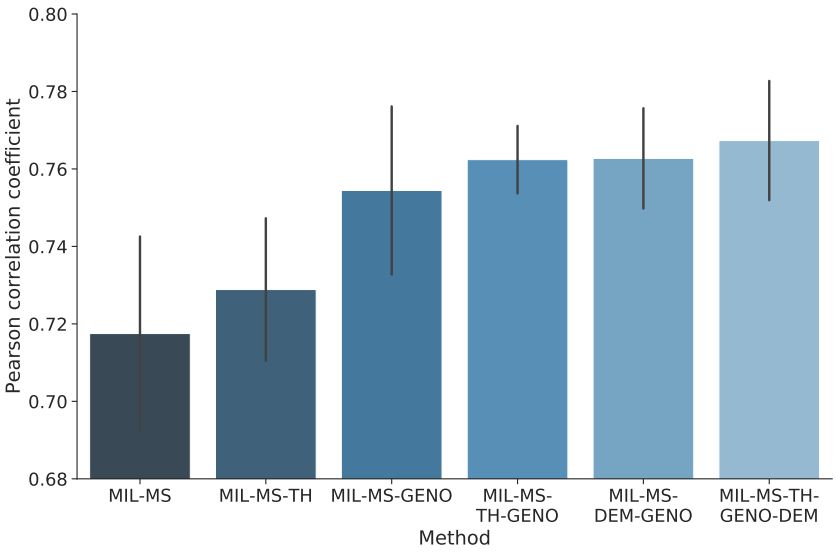


Figure 5.4: Pearson's correlation coefficient on wheat yield prediction for different MIL models and channel combinations. Error bars indicate standard deviation across splits. MS: Multispectral, TH: thermal, DEM: Digital elevation models, GENO: SNP data.

We can therefore conclude that the MIL approach combining four input channels is able to accurately predict the wheat grain yield of wheat crops. While comparison with other studies is difficult given the peculiarity of different crops and setups, we

Table 5.4: Detailed results on wheat yield prediction for baselines and MIL model with varying input channels (mean ± std). MS: Multispectral, TH: thermal, DEM: Digital elevation models, GENO: SNP data.

Method	MAE	MSE	R^2	Pearson Coeff.
MIL-MS	0.372 ± 0.012	0.237 ± 0.019	0.507 ± 0.045	0.717 ± 0.028
MIL-MS-TH	0.364 ± 0.007	0.228 ± 0.011	0.526 ± 0.028	0.729 ± 0.021
MIL-MS-GENO	$\mathbf{0.345 \pm 0.008}$	0.210 ± 0.017	0.563 ± 0.038	0.754 ± 0.024
MIL-MS-GENO-TH	0.352 ± 0.010	0.213 ± 0.005	0.578 ± 0.019	0.762 ± 0.115
MIL-MS-GENO-DEM	0.348 ± 0.008	0.214 ± 0.003	0.579 ± 0.027	0.763 ± 0.185
MIL-MS-GENO-TH-DEM	0.347 ± 0.002	$\mathbf{0.209 \pm 0.004}$	$\mathbf{0.586 \pm 0.028}$	$\mathbf{0.767 \pm 0.019}$

can observe that the performance we report is higher compared to the ones in other studies [158, 245, 263].

5.3.3 Feature importance analysis

The attention mechanism contributes effectively to the performance of the model, but it can also be very helpful to investigate the effects of the different input data. In fact, if we consider the attention values as the weights of a weighted average across the input elements, they represent the relative contributions of the input instances to the final prediction. Therefore, we can look at these values to better understand (i) which data sources are more relevant, and (ii) which temporal windows contain the most informative (i.e. predictive) images. In the selected MIL models, we rely on 8 attention heads, which behave differently. We therefore need to investigate the attention value across all of them. In the following experiments, we use samples from the *test* set and pass them through the trained model, which had never seen these samples before.

Channel contributions

We begin by looking at the attention distribution across data channels. Since each channel has a different and variable number of entries for each sample, we take the average attention value across channels and normalise these into a percentage score to get comparable results. Figure 5.5 shows the contribution of individual data sources for the 8 different heads of the MIL-MS-GENO-TH-DEM model for *one sample*. Different heads indeed give a different importance to each channel. The multispectral images are the ones that constantly get more attention, reaching almost 40% of the attention in head 4. On the opposite side, thermal images overall receive less attention, indicating that their contribution is not so key for this particular sample and task.

To verify if these observation generalise across more samples, we take 100 test instances and plot the attention that each of their channel receive for heads 1 (where

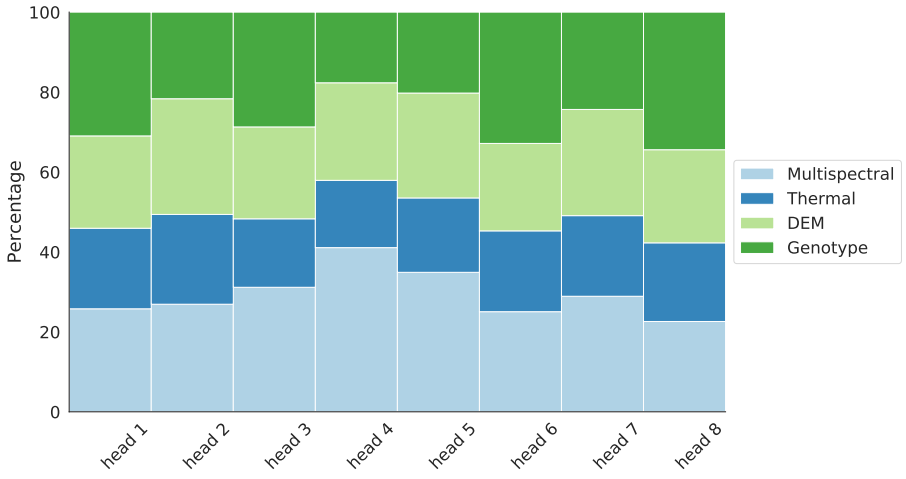


Figure 5.5: Attention distribution in percentage across data channels for one sample and eight heads. Each attention head learns to focus on different representations of the plot and their combined output lead to improved performance.

genotype data is deemed more important) and head 4 (where multispectral images dominate while genotype and thermal data are not considered). Figures 5.6 and 5.7 show that the attention distribution is roughly maintained throughout samples and that head 1 focuses more on genotypic data while head 4 gives a major importance to multispectral images. This indicates that the attention heads indeed tend to specialise in a given data channel and will distinctively extract relevant features for those.

Temporal contributions

Since each input instance from the 3 image channels are also linked to a date, we can investigate what attention the model gives to different channels at given temporal scales. To do so, we combine the attention values across channels and take the mean for a given date. We repeat the operations for multiple samples (here again, 100) and average the obtained "attention time series" to get an indicative distribution of the attention for a given head across time. The resulting plots can be found in Figures 5.10 and 5.11, where the temporal distribution of the attention is presented for heads 1 and 4.

When looking at these plots for all attention heads, one can observe a general trend for multispectral and DEM attentions. The attention mechanisms deem *earlier* multispectral images as more important while it considers *later* DEM images as more relevant. This intuitively makes sense: at early stage of growth, little information about the plant height is contained in the DEM images. Similarly, relevant information about the soil properties, only visible in earlier images, can be accounted

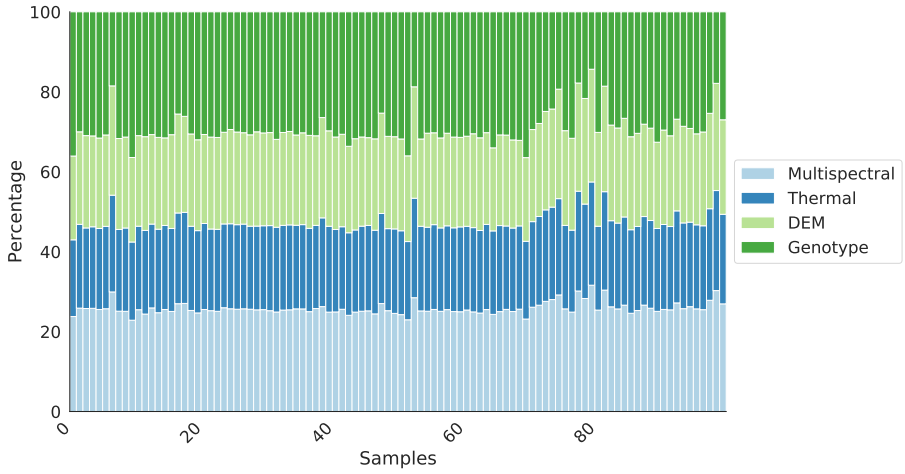


Figure 5.6: Attention distribution of attention head 1 in percentage across data channels for 100 samples.

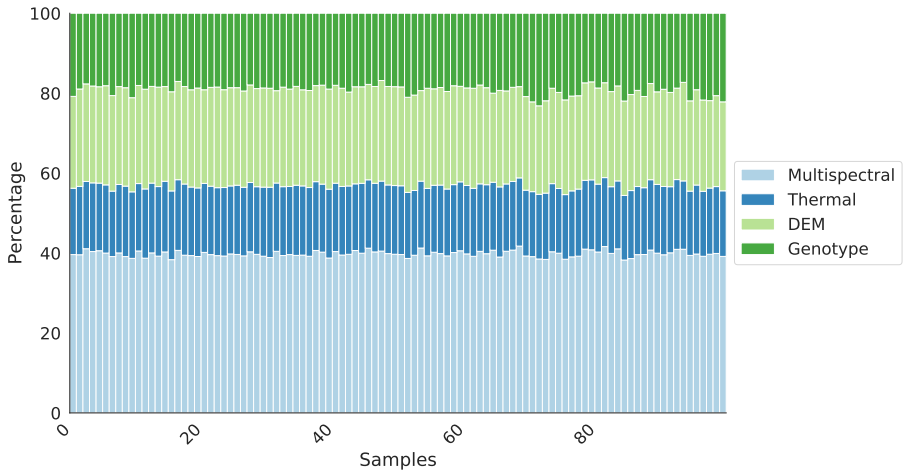


Figure 5.7: Attention distribution of attention head 4 in percentage across data channels for 100 samples.

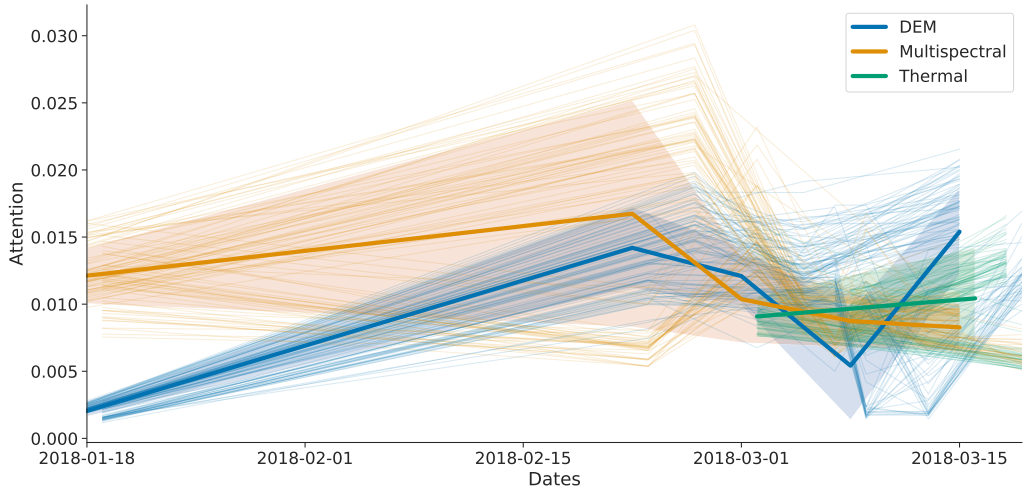


Figure 5.8: Temporal distribution of attention yielded by attention head 1 across data channels for 100 samples.

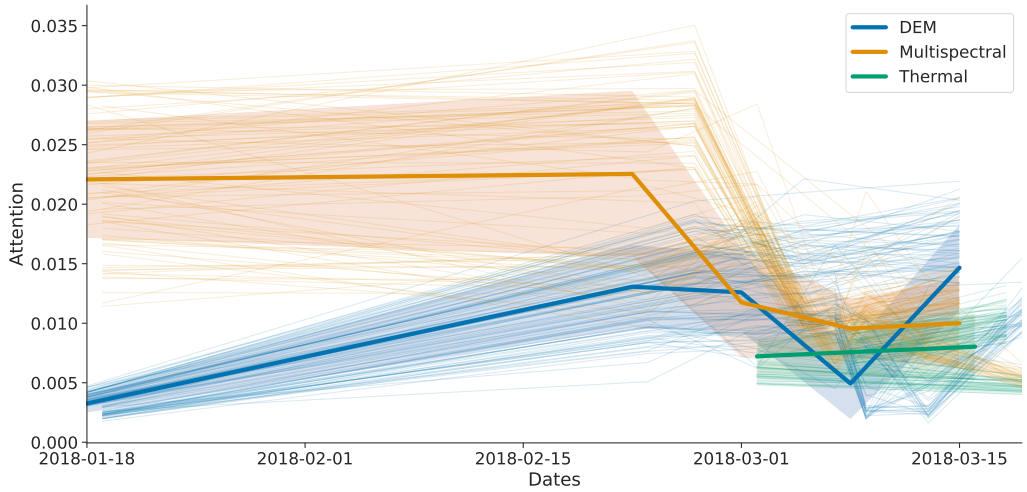


Figure 5.9: Temporal distribution of attention yielded by attention head 4 across data channels for 100 samples.

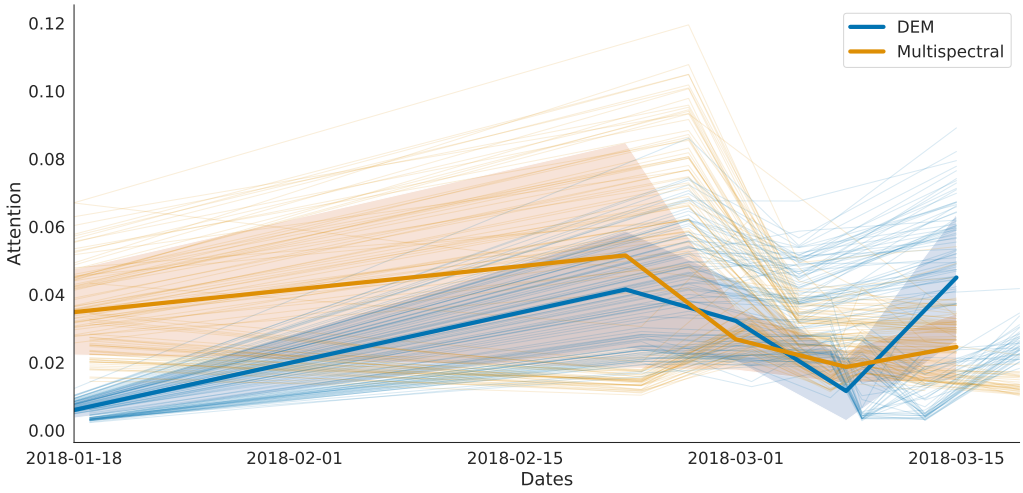


Figure 5.10: Temporal distribution of attention yielded by attention head 1 across data channels without thermal images for 100 samples.

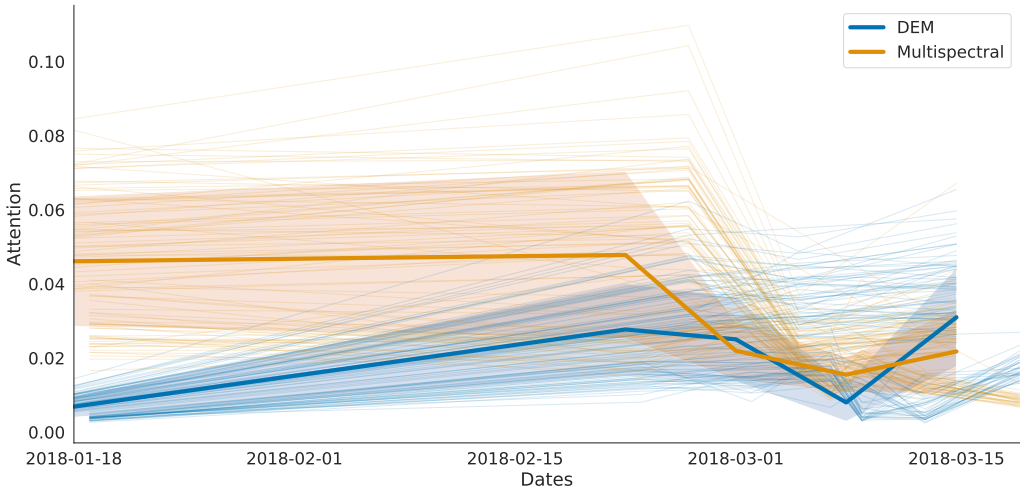


Figure 5.11: Temporal distribution of attention yielded by attention head 4 across data channels without thermal images for 100 samples.

for in the multispectral images. Moreover, there seems to be a higher cumulative attention load towards the middle of the growth phase (e.g. 01.02.2018 - 02.03.2018), which coincide with the period for which linear and non-linear baselines gave better results when using VIs from multispectral images (see Figure 5.2). Thermal images, on the other hand, do not seem to have considerably different importance across time. This is also due to the fact that all samples only had thermal observations at only 2 distinct dates and that these were quite close in time.

To ensure that thermal images are not confounding the relative importance at later stages of the other channels, we remove them at inference time. Figures 5.10 and 5.11 show the distribution of the relative attention obtained by the model when thermal images are discarded (and hence not accounted in the computation of the attention). Similar behavior of the attention distribution can be observed across the other channels, with a slightly smaller difference between multispectral and DEM importance in the intermediate phase. This confirms the findings mentioned above.

5.3.4 Concluding remarks

We here present a flexible, accurate, and interpretable method to predict wheat crop yield. Our approach can be applied to widely diverse data sets and can incorporate endless data channels with instances sampled at irregular time intervals. Additionally, we successfully combine genotypic information with environmental variables by fusing SNP array data with images that also capture environmental conditions such as soil properties and the effects of these conditions across time. We confirm that environmental variables are widely important in phenotype prediction tasks and show that they can easily be incorporated into an end-to-end deep learning model.

6 Conclusions and Outlook

> In which we summarise the findings of the different parts of this thesis, draw conclusions, and present perspective for future research directions.

At the beginning of this thesis, we identified two major challenges for the computational analysis of biological data sets, namely:
- the lack of coherence across the ever-increasing amount of data sets and generated results;
- the lack of efficient ways to account for all the complexity encountered in biological phenomena.

Throughout the remainder of the work, we presented solutions and approaches to solve or mitigate these problems. In this chapter, we summarise the findings collected across the thesis and propose a new outlook for each of the explored directions. We then conclude by giving general directions for future work.

Online resources for *Arabidopsis thaliana*

In Chapter 2, we presented AraPheno and the AraGWAS Catalog. AraPheno is a comprehensive repository for *A. thaliana* phenotypes where everyone can submit new measured phenotypes while the AraGWAS Catalog is the first manually curated and standardised database to collect the results of GWAS for *A. thaliana* on the AraPheno phenotypes. The ultimate goal of these resources is to provide the *Arabidopsis thaliana* community with a homogenised view of all existing associations between the plant's genotype and its multiple phenotypes.

The public availability of high quality genotypes and phenotypes in *A. thaliana* offers the unique opportunity to systematically re-compute and analyse GWAS results using a best-practice pipeline. The Catalog further enables researchers to analyse and compare standardised results of GWAS on different or related traits. These horizontal GWAS analyses unlock unique opportunities to detect seemingly unrelated functions of a gene caused by pleiotropic effects or discover traits with a shared genetic basis. The catalog offers a sophisticated and fast search API to query the database and to extract information for specific associations, genes or traits. Interactive visualizations empower the user to easily maneuver the data and uncover interesting patterns.

For AraPheno, we plan to integrate more phenotypes and meta-information, similar to what we did for the RNA-Seq data. For example, we would like to add climatic

data for all *A. thaliana* accessions allowing users to link environmental variables with different genotypes or phenotypes. Currently, we only provided a link to the AraCLIM portal but plan an easy and smooth integration of such data in future releases. For the AraGWAS Catalog, a permutation-based standardised GWAS pipeline was used to compute univariate associations using a linear mixed model [114] on binary and continuous traits. Nevertheless, recent advances in machine learning enabled the use of generalised linear mixed models for dichotomous traits [161], which we plan to include into future releases of the Catalog. In addition to results of univariate GWAS, we would also like to include SNP-trait associations from multi-locus GWAS [11, 193, 209] and multi-trait GWAS [43, 129]. The platforms are currently focused on data from *A. thaliana*, but we plan to develop a flexible application programming interface (API), such that anyone could build their custom platforms for their individual species.

Our vision would be that anyone generating population-scale phenotypes in A. thaliana will upload their phenotypic data to AraPheno. This will not only make AraPheno and the AraGWAS Catalog more valuable resources, but will also set a precedent for the publication of these data. Furthermore, in the long term, this will facilitate investigations into more complex research questions. Recent work in phenome-wide association studies indicate the promises of such cross-phenotype approaches [41].

Imputation of GWAS summary statistics

The stark need for comparability across studies has pushed the development of many bioinformatics methods relying exclusively on GWAS summary statistics. Nevertheless, the often incomplete overlap of typed SNPs between studies limits the performance of said methods. Imputation of missing values has therefore become a key procedure and several approaches to do so have been proposed. However, as we highlighted in our analysis, some of the available methods suffer from usability issues.

First, not all methods can account for studies with mixed-ethnicity cohorts. Given the constantly increasing size of the participants pools in modern GWAS, it is becoming a necessity to have versatile methods that can easily take in consideration mixed-ethnicity panels. DISTMIX [135], an existing method that can handle this type of cohorts, however relies entirely on allele frequencies for accurate imputation results. While these data might be accessible in certain cases, their exchange has been severely reduced after they were proven to be an effective mean to identify participants in a GWAS [105]. Moreover, as privacy concerns steadily grow, access to sensitive information will only become harder in the future. A method like DISTMIX is therefore severely disadvantaged in the case of missing allele frequencies. Similarly, a method that requires covariates on the original data, like DISSCO [257], simply cannot be executed when these are not present. In fact, given its requirements, it can be argued that the usability of DISSCO is circumscribed to a small niche of users who have access to the original genotype data in a study (and its co-

variates) but prefer to impute summary statistics on the missing SNPs rather than perform the more accurate (yet more computationally intensive) task of imputing the missing genotypes with IMPUTE2, MaCH or others. Lastly, ImpG-Summary [182] offers excellent performance on certain well-defined data sets, but lacks the flexibility necessary to impute missing values in slightly more complex studies. In our experiments, the GWAS study on insomnia shows that an easily adaptable method gives much better imputation performance, even for a self-reported homogeneous cohort.

That is why, in Chapter 3, we introduced Ardiss, a fast, accurate and adaptable method to impute missing Z-scores while inferring the underlying population composition without relying on any extra information such as allele frequencies or covariates of the original study population. Our method matches all use-case scenarios better than other available solutions. Our motivation to develop Ardiss was to simplify the task of imputing summary statistics by providing a unique and robust solution that encompasses all imagined scenarios (cf. Figure 3.1), while at the same time, providing superior imputation accuracy and better runtimes. The results of the performed experiments prove both these claims. Finally, Ardiss is a key element to integrate Z-scores from different studies and will contribute, together with the ever increasing body of publicly available results from association studies in many organisms, to pushing scientists that use GWAS results to ask questions that go beyond the SNP-trait association.

Wasserstein kernels for structured objects

Complex phenomena, such as the ones encountered in biology, sometimes require the use of sophisticated techniques. Advanced data types such as *structured objects* like graphs offer great modeling opportunities for machine learning methods. Similarity measures - or kernels, for structured objects are usually constructed using the $\mathcal{R}$-Convolution framework [101]. Nevertheless, kernels obtained via this framework suffer from pitfalls. For example, classical $\mathcal{R}$-Convolution kernels for graphs rely on simplifying aggregation strategies that discard valuable information about the distribution of nodes. Moreover, in time series, naïve $\mathcal{R}$-Convolution kernels can result in meaningless similarity measures that simply compare the mean value of time series.

In Chapter 4, we highlight these limitations and propose a new family of kernels for structured objects based on the Wasserstein distance, an optimal transport measure. In particular, we present a new family of graph kernels, the Wasserstein Weisfeiler–Lehman (WWL) graph kernels. We theoretically prove some properties of the obtained kernels and our experiments show that WWL outperforms the state of the art for graph classification in the scenario of continuous node attributes, while matching the state of the art in the categorical setting. Similarly, we introduce the Wasserstein Time series Kernel (WTK), a subsequence-based similarity measure that efficiently distinguishes between time series. To prove the benefits of our proposed method, we performed a large-scale evaluation on the several time series benchmark data sets. Extensive analysis scenarios indicate that our method outperforms

some of the state-of-the-art time series classification approaches while also displaying favourable generalisation properties.

As a line of research for future work, we see great potential in the runtime improvement, thus, enabling applications of our method on regimes with larger data sets. Preliminary experiments already confirm the benefits of using Sinkhorn regularisation when the average number of nodes in the graph or length of the time series increases. In parallel, it would be beneficial to derive approximations of the explicit feature representations in the RKKS for both WWL and WTK, as this would also provide a consistent speedup. We further envision that major theoretical contributions could be made by defining theoretical bounds to ensure the positive definiteness of the WWL and WTK kernels when using the Wasserstein distance combined with the euclidean distance. Finally, optimisation objectives based on optimal transport could be employed to develop new algorithms in deep learning with graph neural networks [67, 124] or temporal convolution networks [132]. On a more general level, the proposed methods provide a solid foundation of the use of optimal transport theory for kernel methods and highlight the large potential of optimal transport for machine learning.

Crop yield prediction using deep learning-based data fusion

Phenotype prediction has been a clear focus of genomics for a long time. Past attempts highlighted the problem of "missing heritability": *all* genotypic variants need to be considered in the prediction task and so do their high-order interactions. Moreover, it has also been clear that environmental and developmental factors play a key role in observed phenotypic variability. Nonetheless, very few machine learning approaches account for these aspects. With the advent of deep representation learning, new data fusion possibilities have been unlocked and place deep learning as an obvious candidate for predicting phenotypes by fusing all factors meaningfully.

In Chapter 5 we show how to integrate genotypic data with multitemporal multispectral images, multitemporal thermal images, and multitemporal digital elevation models for accurate prediction of wheat crop yield. By relying on Multiple Instance Learning and attention-based aggregation of learned representations, we show that new state-of-the-art prediction performance can be achieved. Moreover, the attention mechanisms prove to be a rich source of interpretability for the model prediction, giving indications on which data sources and temporal windows are deemed more important.

Our experiments confirm the potential for enhanced phenotype prediction while relying on multiple data sources in addition to genotypic information. Deep learning, despite its seemingly fathomless behaviors, offers great opportunities to capture all the complexity of biological phenomena and gives hints on how to best understand them by providing some interpretability over the prediction mechanism. Naturally, this represents a first preliminary study and much remains to be investigated to understand deep learning suitability in the area of complex phenotype prediction.

We therefore presented a series of approaches to deal with two incumbent problems of data-driven machine learning method development in the life sciences: the incoherence of data sets and the complex nature of biological phenomena. Our recomputation of GWAS results for *Arabidopsis thaliana* as well as the development of a summary statistics imputation method are essential steps in the direction of large-scale comparative analyses of GWAS results. For instance, GWAS results can help elucidate the presence of genomic regions associated with multiple phenotypes, an effect referred to as *pleiotropy* [218]. Investigation of pleiotropic effects has already been performed to some extent but has always been limited to a predefined set of traits or genomic regions or had to rely on raw genotypic data [89, 187]. Moreover, biological pleiotropy can be mapped onto gene networks [44]. The "omnigenic" model proposed by Boyle et al. [34] suggests that pleiotropy is ubiquitous in the human genome since gene regulatory networks are sufficiently interconnected for many non-related genes ("peripheral genes") to affect the few phenotype-related genes ("core genes") in almost every phenotype and to give rise to a phenomenon the authors call network pleiotropy. However, there is currently no described method to confirm said widespread pleiotropy.

More generally, combining GWAS summary statistics with the abundant biological network data could shed some light on the underlying mechanisms of biology. *Causal inference* [184] concepts have already been used with GWAS summary statistics to identify causal relationships between phenotypes via mendelian randomization experiments [61]. But extensions of these methods to also account for known biological networks have yet to be thoroughly explored. Integration of knowledge obtained from different sources can be extremely beneficial and would help to better reflect the complexity of biological phenomena.

As seen in Chapters 4 and 5, machine learning approaches offer promising properties to model complex objects and behaviours. Deep representation learning can be extremely powerful in extracting relevant features and modeling non-linear relationships between data. This power is exemplified by the strong predictive performance deep learning offers. Nevertheless, further research is needed to understand the real extent of the learning abilities of these models. In particular, "nudging" mechanisms offer interesting possibilities to fathom trained deep learning models and to extract more insights on the biological mechanisms. Ma et al. [153], for example, used a cell-inspired deep learning architecture to better understand interactions between cellular subsystems during cellular growth. Similar approaches could be envisioned at the organism level: by combining existing knowledge, genotypic information, and auxiliary phenotypic observations, one could devise deep learning models that account for known effects and model the remainder of interactions between the genetic, environmental and developmental factors. Being able to then inspect individual components of these complex architectures would enable new findings on mechanisms at different biological scales. Additionally, nudging can also be used to other ends. Constructed deep learning architectures could leverage auxiliary phenotypic data at training time

to obtain better genotype-based predictors at test time. Relying on environmental factors during training could nudge the model into extracting specific high-order interactions between genotypic data that would then be retained at test time, when the model runs on genotype data alone. The learned representation could therefore indirectly contribute to explain additional parts of the missing heritability. All in all, the vast progress machine learning has experienced in recent times has paved the way for new exciting ways to interrogate living systems and generate new biological insights: new discoveries are therefore poised to intensify over the coming years.

Acronyms

ARD	Automatic Relevance Determination
GP	Gaussian Process
GWAS	Genome-Wide Association Study
GWD	Graph Wasserstein Distance
MIL	Multiple Instance Learning
OT	Optimal Transport
SNP	Single Nucleotide Polymorphism
TSC	Time Series Classification
UAV	Unmanned Aerial Vehicles
WL	Weisfeiler–Lehman
WTK	Wasserstein Time series Kernel
WWL	Wasserstein Weisfeiler–Lehman

List of Figures

2.1 AraPheno phenotype view, containing details related to the Iron Concentration in leaves. Users can easily cite the phenotype using the DOI or download the reported values with the download button on the top right. 19

2.2 AraPheno accession view, containing details related to the Ör-1 accession, collected in Sweden. Users can easily download the details related to the accession with the download button on the top right. 20

2.3 AraGWAS Catalog detailed study view, containing details about the GWA Study on M216T665 phenotype. Users can easily download the details related to the filtered associations with the download button on the bottom right. (A) Brief description about study related information with links to the phenotype and publication. (B) Summary statistics about SNP type, impact, annotation and MAF. (C) Sorted list of associated markers. (D) Filters to narrow down the list of associated hits. 24

2.4 AraGWAS Catalog association view, containing details about the Chr4_1269036 accession of the GWA Study on M216T665 phenotype. Users can easily visualise the distribution of the phenotype for different allelic groups. 25

2.5 AraGWAS Catalog GWAS HitMap, containing a snapshot overview of all associated hits reported in the Catalog. Each column is a chromosome while each row represents a study of the catalog. The color (yellow to red) indicates of the strength of the association. 26

2.6 AraGWAS Catalog Gene view, showing details of associations around specific genes. Detailed gene descriptions are available when hovering with the cursor over a certain gene. 27

3.1 Decision flowchart for the choice of the best-suited association summary statistics imputation method. No alternative method fits all scenarios, while Ardiss covers all of them without needing additional information about the original study. Accounting for covariates is not necessary if the covariates were taken into account during the original study (see Section 3.3.3). 33

3.2 Stacked contributions of individuals from the two major super populations of interest (African-descent, AFR, and European-descent, EUR) obtained by Ardiss for sets of different ethnic compositions of the original study cohort. The theoretical composition is represented in red along the diagonal. 44

3.3 Relative contribution of individual samples as detected by the weights obtained by ARD for the 100% AA | 0% NHW mix of population. Some residual weights for non-African populations are picked up. Boxplots are obtained by taking the weight output by Ardiss, i.e. one per sample from the reference panel, and grouping them by their super-population code. Super-population codes are reported in Table 3.2. . 45

3.4 Relative contribution of individual samples as detected by the weights obtained by ARD for the 100% AA | 0% NHW mix of population. Some residual weights for non-African populations are picked up. Boxplots are obtained by taking the weight output by Ardiss, i.e. one per sample from the reference panel, and grouping them by their population code. Population codes are reported in Table 3.2. 46

3.5 Weights obtained by Ardiss (x axis) and by DISTMIX using the allele frequencies of the original study (y axis) for a selection of populations. The color code indicates the population to which the weight belongs and the different points are obtained from the various sets of ethnicity mixture. Population codes are reported in Table 3.2. 47

3.6 Pearson's correlation coefficients obtained during full genome imputation across different mixtures of ethnicity sets using Ardiss and comparison partners. ImpG-Summary is run using all the samples in the reference panel and DISTMIX computes the optimal weights from the allele frequencies. The shaded area represents the standard deviation interval across the 10-fold validation. 48

3.7 Relative improvement of Ardiss over ImpG-Summary for different randomised mixed-ethnicity cohorts. Ardiss outperforms ImpG-Summary in all mixture scenarios, with both methods being equally accurate in cases of very heterogeneous cohorts (with practically 50% of AA and NHW). The shaded area represents the standard deviation interval across the 10-fold validation. 49

3.8 Relative improvement of Ardiss over DISTMIX for different randomised mixed-ethnicity cohorts. Ardiss outperforms DISTMIX in all mixture scenarios across the whole genome. The shaded area represents the standard deviation interval across the 10-fold validation. . 50

3.9 Pearson's correlation coefficients obtained during imputation across different mixtures of ethnicity sets using ARDISS and other available methods on chromosome 12. IMPG-SUMMARY was run using all the samples in the reference panel, with only the European samples (ImpG-Summary-EUR) and with only the African samples (ImpG-Summary-AFR). DISTMIX computed the optimal weights from the allele frequencies and was run with manual weight setting (for which we provided it with the original fractions of ASW and CEU). 51

3.10 Pearson's correlation coefficients obtained during ARDISS imputation on chromosome 12 of the 0%AA | 100%NHW admixed cohort for different window sizes. The performance initially increases for larger windows but deteriorates for very large values. The reason is that larger windows lead to less successful automatic relevance determination, since large windows encompass more LD regions and dilute the signal. The imputation step is also affected as larger windows negatively impact imputation by adding considerable noise from low-LD SNPs. 52

3.11 Breakdown of the run times for sequential imputation of summary statistics across chromosomes 18 to 22. 54

3.12 Scaling of ARDISS runtime with increasing number of samples in the reference panel. ARDISS scales linearly for an increasing number of samples as both the weight learning step and the imputation step only rely on inner products of genotypes for the computation of the covariance matrix. The experiments were run on our server, under the conditions specified in Section 3.3.2, for the 0% AA | 100% NHW study over chromosomes 18 to 22. 55

3.13 Breakdown of the runtime for imputation over chromosomes 18-22 using different window sizes. While the complexity of ARDISS is quadratic in the window size, very small window sizes also have a longer runtime in practice. This is due to the larger number of iterations the optimiser needs to converge for small window sizes. The experiments were run on our server, under the conditions specified in Section 3.3.2, for the 0% AA | 100% NHW study. 56

4.1 Schematic summary of the graph Wasserstein distance. First, f generates node embeddings for two input graphs G and G'. Then, the Wasserstein distance between the embedding distributions is computed. 68

4.2 Classification accuracies on graphs with continuous node or edge attributes. Comparison of vertices histogram baseline (VH-C), RBF Weisgeiler–Lehman (RBF-WL), hash graph kernel (HGK-WL, HGK-SP), GraphHopper kernel (GH), and Wasserstein Weisfeiler–Lehman (WWL, ours). 81

4.3 Relative distance between (Erdős–Rényi) graph G and the relative permuted and perturbed variant G' with respect to a third independent graph G'' for an increasing noise level for both the Weisfeiler–Lehman (WL) and the Wasserstein Weisfeiler–Lehman (WWL) distances. . . 83

4.4 Runtime performance of the WWL Kernel computation step with a fixed number of graphs. We also report the time taken to compute the ground distance matrix as `distance_time`. Here, `total_time` is the sum of the time to compute the ground distance and the time taken to solve the optimal transport (ot) problem for the regular solver or the Sinkhorn-regularised one. The logarithmic scale on the right-side figure shows how, for a small average number of nodes, the overhead to run Sinkhorn is higher than the benefits. 85

4.5 The mean value of a kernel matrix constructed for a linear kernel, using a straightforward application of the $\mathcal{R}$-Convolution framework. 87

4.6 To measure the distance between two time series, our method proceeds in several steps. (a) First, all subsequences of the two time series are obtained using a sliding window approach (here, not all subsequences are shown due to the overlap of their windows). (b) Second, the pairwise distance matrix between all subsequences is calculated. Yellow highlights large distances, while blue shows small distances. This matrix on its own is not sufficient to assess the dissimilarity between the two time series, since it is unclear which subsequences correspond to which other. (c) Calculating the optimal transport plan makes correspondences between subsequences more readily visible. For example, the two highlighted subsequences are matched with each other in the plan. Since the two time series have different lengths, some rows of the transport plan also contain fractional matchings, making it possible to individuate fine-grained differences in the distributions of the subsequences. 89

4.7 An explicit visualisation of the transport plan obtained in Figure 4.6c. Every line indicates a (partial) match between two subsequences. The lines are anchored to the beginning of the respective subsequence and their thickness reflect the transport value. Only the largest values are reported. 90

4.8 Comparison of the classification accuracy of WTK against the Linear and the RBF kernels for the "UCR Time Series Archive" data sets. . 95

4.9 "Texas Sharpshooter" plot comparing the *expected* gains of our method WTK with the *actual* gains, relative to DTW-1-NN. 96

4.10 Critical difference plot comparing WTK (shown in bold) against multiple other methods. We observe that there is no statistically significant difference between the performance of our method and state-of-the-art ensemble methods. 97

4.11 Comparison of our method WTK against selected other methods: following the critical difference plot from Figure 4.10, we chose the overall best method (HIVE-COTE), as well as the best deep neural network method (ResNet). Additionally, we also compare against KEMD because of the shared theoretical background it has with WTK. In each of the plots, every point corresponds to one data set, while the axes depict the accuracy of the respective method. We adjusted the axes to a range of $[0.4, 1.0]$ because no lower accuracies occured. In an ideal scenario, all points would be above the diagonal as this would mean that we outperform the respective comparison partner on *all* data sets. 98

4.12 The empirical computational complexity of different subsequence kernels when scaling the dataset. n is the number of samples and m is the length of the time series. The y-axis shows running time normalised with respect to the shortest-running method. 100

5.1 Schematic view of the Multiple Instance Learning (MIL) model used for the wheat grain yield prediction. For each sample (i.e. plot), data from four sources are combined: multitemporal *multispectral* images, multitemporal *thermal* images, multitemporal digital elevation models (*DEM*), and *genotype* data (SNP array). Each data instance is transformed into an embedding via a dedicated model (ResNet-18 for images and a simple fully-connected network - FCN - for genotype data) and the attention mechanisms combines the embeddings into a single embedding which is passed to a final fully-connected network for yield prediction. 111

5.2 Pearson's correlation coefficient for linear and non-linear baselines on wheat yield prediction using multispectral images as well as for a MIL approach. Each baseline relies on VIs values aggregated on a time window (different colors) or on a combination of the aggregated values ("All" bar). Error bars indicate standard deviation across splits. . . 116

5.3 Pearson's correlation coefficient on wheat yield prediction for approaches combining genotype data and multispectral images. Error bars indicate standard deviation across splits. 118

5.4 Pearson's correlation coefficient on wheat yield prediction for different MIL models and channel combinations. Error bars indicate standard deviation across splits. MS: Multispectral, TH: thermal, DEM: Digital elevation models, GENO: SNP data. 119

5.5 Attention distribution in percentage across data channels for one sample and eight heads. Each attention head learns to focus on different representations of the plot and their combined output lead to improved performance. 121

5.6 Attention distribution of attention head 1 in percentage across data channels for 100 samples. 122

5.7 Attention distribution of attention head 4 in percentage across data channels for 100 samples. 122
5.8 Temporal distribution of attention yielded by attention head 1 across data channels for 100 samples. 123
5.9 Temporal distribution of attention yielded by attention head 4 across data channels for 100 samples. 123
5.10 Temporal distribution of attention yielded by attention head 1 across data channels without thermal images for 100 samples. 124
5.11 Temporal distribution of attention yielded by attention head 4 across data channels without thermal images for 100 samples. 124

List of Tables

2.1 AraPheno content and summary statistics as of January 2020. 18
2.2 AraGWAS Catalog content and summary statistics as of January 2020. Numbers of associated hits are filtered by minor allele count (MAC) > 5. Sig. is an abbreviation of Significant. 23
2.3 AraGWAS Catalog Top Genes according to the number of significant hits as of January 2020. The number of associated loci per gene are based on permutation-based thresholds and minor allele count (MAC) > 5. 28

3.1 Sample size details of the COPDGene cohort. The column "Case" refers to individuals who were diagnosed with COPD. The number of SNPs in the intersection of both populations is $615,906$ and we take this as the starting point of our analysis. 41
3.2 Details of the samples in the 1000 Genomes Project that are used in our analyses as reference panel. The four superpopulations are: AFR (African), AMR (ad-mixed American), EAS (East Asian), EUR (European). 43
3.3 Example percentage of recovered top 100 SNPs after imputation with Ardiss on chromosome 12 for the 10% AA | 90% NHW cohort. . . . 53
3.4 Imputation performance of Ardiss and ImpG-Summary on the publicly available summary statistics for the Insomnia Complaints GWAS. 53
3.5 Full results for the imputation performance of ARDISS, ImpG-Summary and DISTMIX for different ethnicity mixtures on Chromosome 12. AA: African American, NHW: Non-Hispanic White. . . 57

4.1 Details of the experimental data sets. 79
4.2 Classification accuracies on graphs with categorical node labels. Comparison of Weisfeiler–Lehman kernel (WL), optimal assignment kernel (WL-OA), and Wasserstein Weisfeiler–Lehman (WWL, ours). 80
4.3 Classification accuracies on graphs with continuous node or edge attributes. Comparison of hash graph kernel (HGK-WL, HGK-SP), GraphHopper kernel (GH), and Wasserstein Weisfeiler–Lehman (WWL, ours). 80
4.4 Classification accuracies on synthetic graphs with continuous node attributes. Comparison of hash graph kernel (HGK-WL, HGK-SP), GraphHopper kernel (GH), and Wasserstein Weisfeiler–Lehman (WWL, ours). 82

4.5 Absolute difference (Δ) in mean accuracy for three different methods with the respective SOTA method. Columns might not sum to 100 % due to rounding. 97

5.1 Details of the data set after quality control and filtering. 113
5.2 Detailed results on wheat yield prediction for baselines and MIL model on multispectral images alone (mean ± std). 117
5.3 Detailed results on wheat yield prediction for baselines and MIL model on multispectral images and SNP array data (mean ± std). 119
5.4 Detailed results on wheat yield prediction for baselines and MIL model with varying input channels (mean ± std). MS: Multispectral, TH: thermal, DEM: Digital elevation models, GENO: SNP data. 120

Bibliography

1. 1000 Genomes Project Consortium et al. "An integrated map of genetic variation from 1,092 human genomes". *Nature* 491:7422, 2012, pp. 56–65.
2. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.
3. B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey. "Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning". *Nature biotechnology* 33:8, 2015, pp. 831–838.
4. C. Alonso-Blanco, J. Andrade, C. Becker, F. Bemm, J. Bergelson, K. Borgwardt, J. Cao, E. Chae, T. M. M. Dezwaan, W. Ding, J. R. R. Ecker, M. Exposito-Alonso, A. Farlow, J. Fitz, X. Gan, D. G. G. Grimm, A. M. M. Hancock, S. R. R. Henz, S. Holm, M. Horton, M. Jarsulic, R. A. A. Kerstetter, A. Korte, P. Korte, C. Lanz, C. R. Lee, D. Meng, T. P. P. Michael, R. Mott, N. W. W. Muliyati, T. Nägele, M. Nagler, V. Nizhynska, M. Nordborg, P. Y. Y. Novikova, F. X. Picó, A. Platzer, F. A. A. Rabanal, A. Rodriguez, B. A. A. Rowan, P. A. A. Salomé, K. J. J. Schmid, R. J. J. Schmitz, Ü. Seren, F. G. G. Sperone, M. Sudkamp, H. Svardal, M. M. M. Tanzer, D. Todd, S. L. L. Volchenboum, C. Wang, G. Wang, X. Wang, W. Weckwerth, D. Weigel, and X. Zhou. "1,135 Genomes Reveal the Global Pattern of Polymorphism in Arabidopsis thaliana". *Cell* 166:2, 2016, pp. 481–491. DOI: 10.1016/j.cell.2016.05.063.
5. J. Altschuler, J. Weed, and P. Rigollet. "Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration". In: *Advances in Neural Information Processing Systems 30*. 2017, pp. 1964–1974.
6. D. Altshuler, M. J. Daly, and E. S. Lander. "Genetic mapping in human disease". *science* 322:5903, 2008, pp. 881–888.
7. S. Andrews, I. Tsochantaridis, and T. Hofmann. "Support vector machines for multiple-instance learning". In: *Advances in neural information processing systems*. 2003, pp. 577–584.
8. M. Arjovsky, S. Chintala, and L. Bottou. "Wasserstein GAN". *arXiv preprint arXiv:1701.07875*, 2017.

9. E. A. Ashley. "Towards precision medicine". *Nature Reviews Genetics* 17:9, 2016, p. 507.
10. S. Atwell, Y. S. Huang, B. J. Vilhjálmsson, G. Willems, M. Horton, Y. Li, D. Meng, A. Platt, A. M. Tarone, T. T. Hu, R. Jiang, N. W. Muliyati, X. Zhang, M. A. Amer, I. Baxter, B. Brachi, J. Chory, C. Dean, M. Debieu, J. De Meaux, J. R. Ecker, N. Faure, J. M. Kniskern, J. D. Jones, T. Michael, A. Nemri, F. Roux, D. E. Salt, C. Tang, M. Todesco, M. B. Traw, D. Weigel, P. Marjoram, J. O. Borevitz, J. Bergelson, and M. Nordborg. "Genome-wide association study of 107 phenotypes in Arabidopsis thaliana inbred lines". *Nature* 465:7298, 2010, pp. 627–631. DOI: 10.1038/nature08800.
11. C. A. Azencott, D. Grimm, M. Sugiyama, Y. Kawahara, and K. Borgwardt. "Efficient network-guided multi-locus association mapping with graph cuts". In: *Bioinformatics*. Vol. 29. 13. 2013. DOI: 10.1093/bioinformatics/btt238.
12. A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh. "The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances". *Data Mining and Knowledge Discovery* 31:3, 2017, pp. 606–660.
13. A. Bagnall, J. Lines, J. Hills, and A. Bostrom. "Time-Series Classification with COTE:The Collective of Transformation-Based Ensembles". *IEEE Transactions on Knowledge and Data Engineering* 27:9, 2015, pp. 2522–2535.
14. M.-F. Balcan, A. Blum, and N. Srebro. "A theory of learning with similarity functions". *Machine Learning* 72:1-2, 2008, pp. 89–112.
15. R. B. Bapat and T. E. S. Raghavan. *Nonnegative matrices and applications*. Cambridge University Press, Cambridge, UK, 1997.
16. L. Barboza, S. Effgen, C. Alonso-Blanco, R. Kooke, J. J. Keurentjes, M. Koornneef, and R. Alcázar. "Arabidopsis semidwarfs evolved from independent mutations in GA20ox1, ortholog to green revolution dwarf alleles in rice and barley". *Proceedings of the National Academy of Sciences of the United States of America* 110:39, 2013, pp. 15818–15823. DOI: 10.1073/pnas.1314979110.
17. G. E. Batista, X. Wang, and E. J. Keogh. "A complexity-invariant distance measure for time series". In: *SDM*. 2011, pp. 699–710.
18. M. Battude, A. Al Bitar, D. Morin, J. Cros, M. Huc, C. M. Sicre, V. Le Dantec, and V. Demarez. "Estimating maize biomass and yield over large areas using high spatial and temporal resolution Sentinel-2 like remote sensing data". *Remote Sensing of Environment* 184, 2016, pp. 668–681.
19. J.-D. Benamou, G. Carlier, M. Cuturi, L. Nenna, and G. Peyré. "Iterative Bregman Projections for Regularized Transportation Problems". *SIAM Journal on Scientific Computing* 37:2, 2015, A1111–A1138.
20. Y. Benjamini and Y. Hochberg. "Controlling the false discovery rate: a practical and powerful approach to multiple testing". *Journal of the Royal statistical society: series B (Methodological)* 57:1, 1995, pp. 289–300.

21. C. Berg, J. P. R. Christensen, and P. Ressel. *Harmonic analysis on semigroups. Theory of positive definite and related functions*. Springer, Heidelberg, Germany, 1984.

22. R. Bernardo. "Bandwagons I, too, have known". *Theoretical and applied genetics* 129:12, 2016, pp. 2323–2332.

23. R. Bernardo. "Molecular markers and selection for complex traits in plants: learning from the last 20 years". *Crop science* 48:5, 2008, pp. 1649–1664.

24. J. S. Bloom, I. M. Ehrenreich, W. T. Loo, T.-L. V. Lite, and L. Kruglyak. "Finding the sources of missing heritability in a yeast cross". *Nature* 494:7436, 2013, pp. 234–237.

25. C. Bock, T. Gumbsch, M. Moor, B. Rieck, D. Roqueiro, and K. Borgwardt. "Association mapping in biomedical time series via statistically significant shapelet mining". *Bioinformatics* 34:13, 2018, pp. i438–i446.

26. C. Bock, M. Togninalli, E. Ghisu, T. Gumbsch, B. Rieck, and K. Borgwardt. "A Wasserstein Subsequence Kernel for Time Series". In: *19th IEEE International Conference on Data Mining (ICDM 2019)*. 2019.

27. A. M. Bolger, H. Poorter, K. Dumschott, M. E. Bolger, D. Arend, S. Osorio, H. Gundlach, K. F. Mayer, M. Lange, U. Scholz, and B. Usadel. *Computational aspects underlying genome to phenome analysis in plants*. 2019. DOI: 10.1111/tpj.14179.

28. C. E. Bonferroni, C Bonferroni, and C. Bonferroni. "Teoria statistica delle classi e calcolo delle probabilita'.", 1936.

29. K. M. Borgwardt and H.-P. Kriegel. "Shortest-path kernels on graphs". In: *Proceedings of the Fifth IEEE International Conference on Data Mining*. 2005, pp. 74–81.

30. K. M. Borgwardt, C. S. Ong, S. Schönauer, S. Vishwanathan, A. J. Smola, and H.-P. Kriegel. "Protein function prediction via graph kernels". *Bioinformatics* 21, 2005, pp. i47–i56.

31. A. Bostrom and A. Bagnall. "Binary Shapelet Transform for Multiclass Time Series Classification". In: *International Conference on Big Data Analytics and Knowledge Discovery*. Springer. 2015, pp. 257–269.

32. N. Bouain, S. B. Satbhai, A. Korte, C. Saenchai, G. Desbrosses, P. Berthomieu, W. Busch, and H. Rouached. "Natural allelic variation of the AZI1 gene controls root growth under zinc-limiting condition". *PLoS Genetics* 14:4, 2018. DOI: 10.1371/journal.pgen.1007304.

33. R. Bouckaert, T. G. Vaughan, J. Barido-Sottani, S. Duchêne, M. Fourment, A. Gavryushkina, J. Heled, G. Jones, D. Kühnert, N. De Maio, et al. "BEAST 2.5: An advanced software platform for Bayesian evolutionary analysis". *PLoS computational biology* 15:4, 2019, e1006650.

34. E. A. Boyle, Y. I. Li, and J. K. Pritchard. "An Expanded View of Complex Traits: From Polygenic to Omnigenic". *Cell* 169:7, 2017, pp. 1177–1186. DOI: 10.1016/j.cell.2017.05.038. URL: http://dx.doi.org/10.1016/j.cell.2017.05.038.

35. L. Breiman. "Random Forests". *Machine Learning* 45:1, 2001, pp. 5–32. ISSN: 1573-0565. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324.

36. M. R. Bridson and A. Häfliger. *Metric spaces of non-positive curvature*. Springer, Heidelberg, Germany, 2013.

37. B. L. Browning and S. R. Browning. "A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals". *American Journal of Human Genetics* 84:2, 2008, pp. 210–223. DOI: 10.1016/j.ajhg.2009.01.005.

38. B. L. Browning and S. R. Browning. "Genotype imputation with millions of reference samples". *The American Journal of Human Genetics* 98:1, 2016, pp. 116–126.

39. S. R. Browning and B. L. Browning. "Rapid and accurate haplotype phasing and missing-data inference for whole-genome association studies by use of localized haplotype clustering". *American Journal of Human Genetics* 81:5, 2007, pp. 1084–1097. DOI: 10.1086/521987.

40. J. Burgueño, G. de los Campos, K. Weigel, and J. Crossa. "Genomic prediction of breeding values when modeling genotype× environment interaction using pedigree and dense molecular markers". *Crop Science* 52:2, 2012, pp. 707–719.

41. W. S. Bush, M. T. Oetjens, and D. C. Crawford. "Unravelling the human genome–phenome relationship using phenome-wide association studies". *Nature Reviews Genetics* 17:3, 2016, p. 129.

42. M. C. Campbell and S. A. Tishkoff. "African genetic diversity: implications for human demographic history, modern human origins, and complex disease mapping". *Annu Rev Genomics Hum Genet* 9, 2008, pp. 403–433.

43. F. P. Casale, B. Rakitsch, C. Lippert, and O. Stegle. "Efficient set tests for the genetic analysis of correlated traits". *Nature Methods* 12:8, 2015, pp. 755–758. DOI: 10.1038/nmeth.3439.

44. S. Chavali, F. Barrenas, K. Kanduri, and M. Benson. "Network properties of human disease genes with pleiotropic effects." *BMC systems biology* 4, 2010, p. 78. DOI: 10.1186/1752-0509-4-78. URL: http://www.ncbi.nlm.nih.gov/pubmed/20525321http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2892460.

45. Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista. *The UCR Time Series Classification Archive*. 2015. URL: http://www.cs.ucr.edu/~eamonn/time_series_data.

46. C.-Y. Cheng, V. Krishnakumar, A. P. Chan, F. Thibaud-Nissen, S. Schobel, and C. D. Town. "Araport11: a complete reannotation of the Arabidopsis thaliana reference genome". *The Plant Journal* 89:4, 2017, pp. 789–804. DOI: 10.1111/tpj.13415. URL: http://doi.wiley.com/10.1111/tpj.13415.

47. A. Chlingaryan, S. Sukkarieh, and B. Whelan. "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review". *Computers and electronics in agriculture* 151, 2018, pp. 61–69.

48. M. H. Cho, M.-L. N. McDonald, X. Zhou, M. Mattheisen, P. J. Castaldi, C. P. Hersh, D. L. DeMeo, J. S. Sylvia, J. Ziniti, N. M. Laird, C. Lange, A. A. Litonjua, D. Sparrow, R. Casaburi, R. G. Barr, E. A. Regan, B. J. Make, J. E. Hokanson, S. Lutz, T. M. Dudenkov, H. Farzadegan, J. B. Hetmanski, R. Tal-Singer, D. A. Lomas, P. Bakke, A. Gulsvik, J. D. Crapo, E. K. Silverman, and T. H. Beaty. "Risk loci for chronic obstructive pulmonary disease: a genome-wide association study and meta-analysis". *The Lancet Respiratory Medicine* 2:3, 2014, pp. 214–225. DOI: http://dx.doi.org/10.1016/S2213-2600(14)70002-5.

49. P. Cingolani, A. Platts, L. L. Wang, M. Coon, T. Nguyen, L. Wang, S. J. Land, X. Lu, and D. M. Ruden. "A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3". *Fly* 6:2, 2012, pp. 80–92. DOI: 10.4161/fly.19695.

50. I. Colomina and P. Molina. "Unmanned aerial systems for photogrammetry and remote sensing: A review". *ISPRS Journal of photogrammetry and remote sensing* 92, 2014, pp. 79–97.

51. . G. Consortium. "1,135 genomes reveal the global pattern of polymorphism in Arabidopsis thaliana". *Cell* 166, pp. 481–491.

52. I. W. G. S. Consortium et al. "A chromosome-based draft sequence of the hexaploid bread wheat (Triticum aestivum) genome". *Science* 345:6194, 2014, p. 1251788.

53. J. Crossa, P. Pérez-Rodríguez, J. Cuevas, O. Montesinos-López, D. Jarquín, G. de los Campos, J. Burgueño, J. M. González-Camacho, S. Pérez-Elizalde, Y. Beyene, et al. "Genomic selection in plant breeding: methods, models, and perspectives". *Trends in plant science* 22:11, 2017, pp. 961–975.

54. J. Cuevas, J. Crossa, V. Soberanis, S. Pérez-Elizalde, P. Pérez-Rodríguez, G. d. l. Campos, O. Montesinos-López, and J. Burgueño. "Genomic prediction of genotype× environment interaction kernel regression models". *The plant genome* 9:3, 2016.

55. M. Cuturi. "Fast global alignment kernels". In: *28th International Conference on Machine Learning (ICML)*. 2011, pp. 929–936.

56. M. Cuturi. "Permanents, transportation polytopes and positive definite kernels on histograms". In: *IJCAI*. 2007, pp. 732–737.

57. M. Cuturi. "Sinkhorn distances: Lightspeed computation of optimal transport". In: *Advances in Neural Information Processing Systems 26*. 2013, pp. 2292–2300.

58. M. Cuturi, J.-P. Vert, Ø. Birkenes, and T. Matsui. "A kernel for time series based on global alignments". In: *ICASSP*. Vol. 2. 2007, pp. 413–416.

59. M. R. Daliri. "Kernel earth mover's distance for EEG classification". *Clinical EEG and Neuroscience* 44:3, 2013, pp. 182–187.

60. H. A. Dau, A. J. Bagnall, K. Kamgar, C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, and E. J. Keogh. "The UCR Time Series Archive". *arXiv e-prints* abs/1810.07758, 2018.

61. G. Davey Smith and S. Ebrahim. "'Mendelian randomization': can genetic epidemiology contribute to understanding environmental determinants of disease?" *International journal of epidemiology* 32:1, 2003, pp. 1–22.

62. J. Demšar. "Statistical comparisons of classifiers over multiple data sets". *Journal of Machine Learning Research* 7, 2006, pp. 1–30.

63. B Devlin, K. Roeder, and L. Wasserman. "Genomic control, a new approach to genetic-based association studies". *Theoretical population biology* 60:3, 2001, pp. 155–166.

64. B. Devlin and K. Roeder. "Genomic control for association studies". *Biometrics* 55:4, 1999, pp. 997–1004.

65. H. Dittberner, A. Korte, T. Mettler-Altmann, A. Weber, G. Monroe, and J. de Meaux. "Natural variation in stomata size contributes to the local adaptation of water-use efficiency in Arabidopsis thaliana". *Molecular Ecology* 27:20, 2018, pp. 4052–4065. DOI: 10.1111/mec.2018.27.issue-20.

66. M. Du and N. Noguchi. "Monitoring of wheat growth status and mapping of wheat yield's within-field spatial variations using color images acquired from UAV-camera system". *Remote sensing* 9:3, 2017, p. 289.

67. D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams. "Convolutional networks on graphs for learning molecular fingerprints". In: *Advances in Neural Information Processing Systems 28*. 2015, pp. 2224–2232.

68. Y. Erlich and A. Narayanan. "Routes for breaching and protecting genetic privacy". *Nature Reviews Genetics* 15:6, 2014, pp. 409–421.

69. M. Exposito-Alonso, M. Exposito-Alonso, R. Gómez Rodríguez, C. Barragán, G. Capovilla, E. Chae, J. Devos, E. S. Dogan, C. Friedemann, C. Gross, P. Lang, D. Lundberg, V. Middendorf, J. Kageyama, T. Karasov, S. Kersten, S. Petersen, L. Rabbani, J. Regalado, L. Reinelt, B. Rowan, D. K. Seymour, E. Symeonidi, R. Schwab, D. T. N. Tran, K. Venkataramani, A. L. Van de Weyer, F. Vasseur, G. Wang, R. Wedegärtner, F. Weiss, R. Wu, W. Xi, M. Zaidem, W. Zhu, F. García-Arenal, H. A. Burbano, O. Bossdorf, D. Weigel, H. A. Burbano,

O. Bossdorf, R. Nielsen, and D. Weigel. "Natural selection on the Arabidopsis thaliana genome in present and future climates". *Nature* 573:7772, 2019, pp. 126–129. DOI: 10.1038/s41586-019-1520-9.

70. W. Falcon et al. *PyTorch Lightning*. https://github.com/PytorchLightning/pytorch-lightning. 2019.

71. H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller. "Deep learning for time series classification: a review". *Data Mining and Knowledge Discovery*, 2019, pp. 1–47.

72. A. Feragen, N. Kasenburg, J. Petersen, M. de Bruijne, and K. Borgwardt. "Scalable kernels for graphs with continuous attributes". In: *Advances in Neural Information Processing Systems 26*. 2013, pp. 216–224.

73. A. Feragen, F. Lauze, and S. Hauberg. "Geodesic exponential kernels: When curvature and linearity conflict". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3032–3042.

74. Á. Ferrero-Serrano and S. M. Assmann. "Phenotypic and genome-wide association with the local environment of Arabidopsis". *Nature Ecology and Evolution* 3:2, 2019, pp. 274–285. DOI: 10.1038/s41559-018-0754-5.

75. A. Figalli and C. Villani. "Optimal transport and curvature". In: *Nonlinear PDE's and Applications*. Springer, Heidelberg, Germany, 2011, pp. 171–217.

76. R. A. Fisher et al. "On the "Probable Error" of a Coefficient of Correlation Deduced from a Small Sample.", 1921.

77. R. Flamary and N. Courty. *POT Python Optimal Transport library*. 2017. URL: https://github.com/rflamary/POT.

78. Food and A. O. of the United Nations. *The State of Food Security and Nutrition in the World 2019*. 2019, p. 237. DOI: https://doi.org/https://doi.org/10.18356/63e608ce-en. URL: https://www.un-ilibrary.org/content/publication/63e608ce-en.

79. T. Freilinger, V. Anttila, B. De Vries, R. Malik, M. Kallela, G. M. Terwindt, P. Pozo-Rosich, B. Winsvold, D. R. Nyholt, W. P. Van Oosterhout, V. Artto, U. Todt, E. Hämäläinen, J. Fernández-Morales, M. A. Louter, M. A. Kaunisto, J. Schoenen, O. Raitakari, T. Lehtimäki, M. Vila-Pueyo, H. Göbel, E. Wichmann, C. Sintas, A. G. Uitterlinden, A. Hofman, F. Rivadeneira, A. Heinze, E. Tronvik, C. M. Van Duijn, J. Kaprio, B. Cormand, M. Wessman, R. R. Frants, T. Meitinger, B. Müller-Myhsok, J. A. Zwart, M. Färkkilä, A. MacAya, M. D. Ferrari, C. Kubisch, A. Palotie, M. Dichgans, and A. M. Van Den Maagdenberg. "Genome-wide association analysis identifies susceptibility loci for migraine without aura". *Nature Genetics* 44:7, 2012, pp. 777–782. DOI: 10.1038/ng.2307.

80. J. A. Freudenthal, M. J. Ankenbrand, D. G. Grimm, and A. Korte. "GWASFlow: A GPU accelerated framework for efficient permutation based genome-wide association studies". *bioRxiv*, 2019, p. 783100. DOI: 10.1101/783100. URL: http://biorxiv.org/content/early/2019/09/27/783100.abstract.

81. J. H. Friedman. "Stochastic gradient boosting". *Computational statistics & data analysis* 38:4, 2002, pp. 367–378.

82. M. Friedman. "The use of ranks to avoid the assumption of normality implicit in the analysis of variance". *Journal of the American Statistical Association* 32:200, 1937, pp. 675–701. DOI: 10.1080/01621459.1937.10503522.

83. C. Frogner, C. Zhang, H. Mobahi, M. Araya, and T. A. Poggio. "Learning with a Wasserstein loss". In: *Advances in Neural Information Processing Systems 28*. 2015, pp. 2053–2061.

84. H. Fröhlich, J. K. Wegner, F. Sieker, and A. Zell. "Optimal Assignment Kernels for Attributed Molecular Graphs". In: *Proceedings of the 22nd International Conference on Machine Learning*. 2005, pp. 225–232.

85. R. E. Furrow, F. B. Christiansen, and M. W. Feldman. "Environment-sensitive epigenetics and the heritability of complex diseases". *Genetics* 189:4, 2011, pp. 1377–1387.

86. A. Gardner, C. A. Duncan, J. Kanno, and R. R. Selmic. "On the Definiteness of Earth Mover's Distance and Its Relation to Set Intersection". *IEEE Transactions on Cybernetics*, 2017.

87. A. Gaulton, A. Hersey, M. Nowotka, A. P. Bento, J. Chambers, D. Mendez, P. Mutowo, F. Atkinson, L. J. Bellis, E. Cibrián-Uhalte, M. Davies, N. Dedman, A. Karlsson, M. P. Magariños, J. P. Overington, G. Papadatos, I. Smit, and A. R. Leach. "The ChEMBL database in 2017". *Nucleic Acids Research* 45:D1, 2016, pp. D945–D954. DOI: 10.1093/nar/gkw1074.

88. S. Gharghabi, S. Imani, A. Bagnall, A. Darvishzadeh, and E. Keogh. "Matrix Profile XII: MPdist: A Novel Time Series Distance Measure to Allow Data Mining in More Challenging Scenarios". In: *IEEE International Conference on Data Mining (ICDM)*. 2018, pp. 965–970.

89. C. Giambartolomei, J. Zhenli Liu, W. Zhang, M. Hauberg, H. Shi, J. Boocock, J. Pickrell, A. E. Jaffe, C. Consortium, B. Pasaniuc, et al. "A Bayesian framework for multiple trait colocalization from summary association statistics". *Bioinformatics* 34:15, 2018, pp. 2538–2545.

90. Y. Gong, B. Duan, S. Fang, R. Zhu, X. Wu, Y. Ma, and Y. Peng. "Remote estimation of rapeseed yield with unmanned aerial vehicle (UAV) imaging and spectral mixture analysis". *Plant methods* 14:1, 2018, p. 70.

91. I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

92. J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme. "Learning time-series shapelets". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2014, pp. 392–401.

93. P. Guldberg, F. Rey, J. Zschocke, V. Romano, B. François, L. Michiels, K. Ullrich, G. F. Hoffmann, P. Burgard, H. Schmidt, et al. "A European multicenter study of phenylalanine hydroxylase deficiency: classification of 105 mutations and a general system for genotype-based prediction of metabolic phenotype". *The American Journal of Human Genetics* 63:1, 1998, pp. 71–79.

94. A. C. Gumpinger, D. Roqueiro, D. G. Grimm, and K. Borgwardt. "Methods and tools in genome-wide association studies". In: *Methods in Molecular Biology*. Vol. 1819. Springer, NY, 2018, pp. 93–136. DOI: 10.1007/978-1-4939-8618-7{_}5.

95. J. Guo, G. Tian, Y. Zhou, M. Wang, N. Ling, Q. Shen, and S. Guo. "Evaluation of the grain yield and nitrogen nutrient status of wheat (Triticum aestivum L.) using thermal imaging". *Field Crops Research* 196, 2016, pp. 463–472.

96. B. Haasdonk. "Feature space interpretation of SVMs with indefinite kernels". *IEEE TPAMI* 27:4, 2005, pp. 482–492.

97. B. Haasdonk and C. Bahlmann. "Learning with Distance Substitution Kernels". In: *DAGM-Symposium*. 2004.

98. A. Haghighattalab, L. G. Pérez, S. Mondal, D. Singh, D. Schinstock, J. Rutkoski, I. Ortiz-Monasterio, R. P. Singh, D. Goodin, and J. Poland. "Application of unmanned aerial systems for high throughput phenotyping of large wheat breeding nurseries". *Plant Methods* 12:1, 2016, p. 35.

99. J. Hallmayer, S. Cleveland, A. Torres, J. Phillips, B. Cohen, T. Torigoe, J. Miller, A. Fedele, J. Collins, K. Smith, et al. "Genetic heritability and shared environmental factors among twin pairs with autism". *Archives of general psychiatry* 68:11, 2011, pp. 1095–1102.

100. A. R. Hammerschlag, S. Stringer, C. A. De Leeuw, S. Sniekers, E. Taskesen, K. Watanabe, T. F. Blanken, K. Dekker, B. H. Te Lindert, R. Wassing, I. Jonsdottir, G. Thorleifsson, H. Stefansson, T. Gislason, K. Berger, B. Schormair, J. Wellmann, J. Winkelmann, K. Stefansson, K. Oexle, E. J. Van Someren, and D. Posthuma. "Genome-wide association analysis of insomnia complaints identifies risk genes and genetic overlap with psychiatric and metabolic traits". *Nature Genetics* 49:11, 2017, pp. 1584–1592. DOI: 10.1038/ng.3888.

101. D. Haussler. *Convolution kernels on discrete structures*. Technical report. Department of Computer Science, University of California, 1999.

102. K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

103. S. Heremans, Q. Dong, B. Zhang, L. Bydekerke, and J. Van Orshoven. "Potential of ensemble tree methods for early-season prediction of winter wheat yield from short time series of remotely sensed normalized difference vegetation index and in situ meteorological data". *Journal of Applied Remote Sensing* 9:1, 2015, p. 097095.

104. S. Hochreiter and J. Schmidhuber. "Long short-term memory". *Neural computation* 9:8, 1997, pp. 1735–1780.

105. N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig. "Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays". *PLoS genetics* 4:8, 2008, e1000167.

106. M. Horn, M. Moor, C. Bock, B. Rieck, and K. Borgwardt. "Set Functions for Time Series". *arXiv preprint arXiv:1909.12064*, 2019.

107. M. W. Horton, A. M. Hancock, Y. S. Huang, C. Toomajian, S. Atwell, A. Auton, N. W. Muliyati, A. Platt, F. G. Sperone, B. J. Vilhjálmsson, M. Nordborg, J. O. Borevitz, and J. Bergelson. "Genome-wide patterns of genetic variation in worldwide Arabidopsis thaliana accessions from the RegMap panel". *Nature Genetics* 44:2, 2012, pp. 212–216. DOI: 10.1038/ng.1042.

108. B. N. Howie, P. Donnelly, and J. Marchini. "A flexible and accurate genotype imputation method for the next generation of genome-wide association studies". *PLoS Genetics* 5:6, 2009. DOI: 10.1371/journal.pgen.1000529.

109. S. L. Hyland, M. Faltys, M. Hüser, X. Lyu, T. Gumbsch, C. Esteban, C. Bock, M. Horn, M. Moor, B. Rieck, et al. "Early prediction of circulatory failure in the intensive care unit using machine learning". *Nature Medicine*, 2020, pp. 1–10.

110. M. Ilse, J. Tomczak, and M. Welling. "Attention-based Deep Multiple Instance Learning". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, Stockholmsmässan, Stockholm Sweden, 2018, pp. 2127–2136. URL: http://proceedings.mlr.press/v80/ilse18a.html.

111. U. Johanson, J. West, C. Lister, S. Michaels, R. Amasino, and C. Dean. "Molecular analysis of FRIGIDA, a major determinant of natural variation in Arabidopsis flowering time". *Science* 290:5490, 2000, pp. 344–347. DOI: 10.1126/science.290.5490.344.

112. D. M. Johnson. "An assessment of pre-and within-season remotely sensed variables for forecasting corn and soybean yields in the United States". *Remote Sensing of Environment* 141, 2014, pp. 116–128.

113. M. M. Julkowska, I. T. Koevoets, S. Mol, H. Hoefsloot, R. Feron, M. A. Tester, J. J. Keurentjes, A. Korte, M. A. Haring, G. J. De Boer, and C. Testerink. "Genetic components of root architecture remodeling in response to salt stress". *Plant Cell* 29:12, 2017, pp. 3198–3213. DOI: 10.1105/tpc.16.00680.

114. H. M. Kang, J. H. Sul, S. K. Service, N. A. Zaitlen, S. Y. Kong, N. B. Freimer, C. Sabatti, and E. Eskin. "Variance component model to account for sample structure in genome-wide association studies". *Nature Genetics* 42:4, 2010, pp. 348–354. DOI: 10.1038/ng.548.

115. H. Kashima, K. Tsuda, and A. Inokuchi. "Marginalized Kernels between Labeled Graphs". In: *Proceedings of the 20th International Conference on Machine Learning*. 2003, pp. 321–328.

116. R. J. Kate. "Using dynamic time warping distances as features for improved time series classification". *Data Mining and Knowledge Discovery* 30:2, 2016, pp. 283–312.

117. E. Keogh and J. Lin. "Clustering of time-series subsequences is meaningless: implications for previous and future research". *Knowledge and Information Systems* 8:2, 2005, pp. 154–177.

118. E. Keogh and C. A. Ratanamahatana. "Exact indexing of dynamic time warping". *Knowledge and Information Systems* 7:3, 2005, pp. 358–386.

119. K. Kersting, N. M. Kriege, C. Morris, P. Mutzel, and M. Neumann. *Benchmark Data Sets for Graph Kernels*. 2016. URL: http://graphkernels.cs.tu-dortmund.de.

120. S. Khaki, L. Wang, and S. V. Archontoulis. "A CNN-RNN framework for crop yield prediction". *Frontiers in Plant Science* 10, 2020, p. 1750.

121. A. V. Khera, M. Chaffin, K. Aragam, C. A. Emdin, D. Klarin, M. Haas, C. Roselli, P. Natarajan, and S. Kathiresan. "Genome-wide polygenic score to identify a monogenic risk-equivalent for coronary disease". *bioRxiv*, 2017, p. 218388.

122. A. V. Khera, M. Chaffin, K. G. Aragam, M. E. Haas, C. Roselli, S. H. Choi, P. Natarajan, E. S. Lander, S. A. Lubitz, P. T. Ellinor, et al. "Genome-wide polygenic scores for common diseases identify individuals with risk equivalent to monogenic mutations". *Nature genetics* 50:9, 2018, pp. 1219–1224.

123. J. Kim, J. H. Kim, J. I. Lyu, H. R. Woo, and P. O. Lim. *New insights into the regulation of leaf senescence in Arabidopsis*. 2018. DOI: 10.1093/jxb/erx287.

124. T. N. Kipf and M. Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *5th International Conference on Learning Representations*. 2017.

125. A. Kirby, H. M. Kang, C. M. Wade, C. Cotsapas, E. Kostem, B. Han, N. Furlotte, E. Y. Kang, M. Rivas, M. A. Bogue, K. A. Frazer, F. M. Johnson, E. J. Beilharz, D. R. Cox, E. Eskin, and M. J. Daly. "Fine mapping in 94 inbred mouse strains using a high-density haplotype resource". *Genetics* 185:3, 2010, pp. 1081–1095. DOI: 10.1534/genetics.110.115014.

126. J. Klicpera, A. Bojchevski, and S. Günnemann. "Combining Neural Networks with Personalized PageRank for Classification on Graphs". In: *7th International Conference on Learning Representations*. 2019.

127. S. Kolouri, Y. Zou, and G. K. Rohde. "Sliced Wasserstein Kernels for Probability Distributions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 5258–5267.

128. M. Koornneef and D. Meinke. "The development of Arabidopsis as a model plant". *Plant Journal* 61:6, 2010, pp. 909–921. DOI: 10.1111/j.1365-313X.2009.04086.x.

129. A. Korte, B. J. Vilhjálmsson, V. Segura, A. Platt, Q. Long, and M. Nordborg. "A mixed-model approach for genome-wide association studies of correlated traits in structured populations". *Nature Genetics* 44:9, 2012, pp. 1066–1071. DOI: 10.1038/ng.2376.

130. N. Kriege and P. Mutzel. "Subgraph Matching Kernels for Attributed Graphs". In: *Proceedings of the 29th International Conference on Machine Learning*. 2012, pp. 1015–1022.

131. N. M. Kriege, P.-L. Giscard, and R. C. Wilson. "On Valid Optimal Assignment Kernels and Applications to Graph Classification". In: *Advances in Neural Information Processing Systems 29*. 2016, pp. 1623–1631.

132. C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager. "Temporal convolutional networks for action segmentation and detection". In: *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 156–165.

133. Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning". *nature* 521:7553, 2015, pp. 436–444.

134. D. Lee, T. B. Bigdeli, B. P. Riley, A. H. Fanous, and S. A. Bacanu. "DIST: Direct imputation of summary statistics for unmeasured SNPs". *Bioinformatics* 29:22, 2013, pp. 2925–2927. DOI: 10.1093/bioinformatics/btt500.

135. D. Lee, T. B. Bigdeli, V. S. Williamson, V. I. Vladimirov, B. P. Riley, A. H. Fanous, and S. A. Bacanu. "DISTMIX: Direct imputation of summary statistics for unmeasured SNPs from mixed ethnicity cohorts". *Bioinformatics* 31:19, 2015, pp. 3099–3104. DOI: 10.1093/bioinformatics/btv348.

136. L. Lello, S. G. Avery, L. Tellier, A. I. Vazquez, G. de los Campos, and S. D. Hsu. "Accurate genomic prediction of human height". *Genetics* 210:2, 2018, pp. 477–497.

137. R. Leslie, C. J. O'Donnell, and A. D. Johnson. "GRASP: analysis of genotype–phenotype results from 1390 genome-wide association studies and corresponding open access database". *Bioinformatics* 30:12, 2014, pp. i185–i194. DOI: 10.1093/bioinformatics/btu273.

138. Y. Li, C. J. Willer, J. Ding, P. Scheet, and G. R. Abecasis. "MaCH: Using sequence and genotype data to estimate haplotypes and unobserved genotypes". *Genetic Epidemiology* 34:8, 2010, pp. 816–834. DOI: 10.1002/gepi.20533.

139. L. Liang, L. Di, L. Zhang, M. Deng, Z. Qin, S. Zhao, and H. Lin. "Estimation of crop LAI using hyperspectral vegetation indices and a hybrid inversion method". *Remote Sensing of Environment* 165, 2015, pp. 123–134.

140. H.-T. Lin and C.-J. Lin. *A Study on Sigmoid Kernels for SVM and the Training of non-PSD Kernels by SMO-type Methods*. Technical report. National Taiwan University, 2003.

141. J. Lin, E. Keogh, L. Wei, and S. Lonardi. "Experiencing SAX: a novel symbolic representation of time series". *Data Mining and Knowledge Discovery* 15:2, 2007, pp. 107–144. ISSN: 1573-756X. DOI: 10.1007/s10618-007-0064-z. URL: https://doi.org/10.1007/s10618-007-0064-z.

142. T. Lin, G. Zhu, J. Zhang, X. Xu, Q. Yu, Z. Zheng, Z. Zhang, Y. Lun, S. Li, X. Wang, Z. Huang, J. Li, C. Zhang, T. Wang, Y. Zhang, A. Wang, Y. Zhang, K. Lin, C. Li, G. Xiong, Y. Xue, A. Mazzucato, M. Causse, Z. Fei, J. J. Giovannoni, R. T. Chetelat, D. Zamir, T. Städler, J. Li, Z. Ye, Y. Du, and S. Huang. "Genomic analyses provide insights into the history of tomato breeding". *Nature Genetics* 46:11, 2014, pp. 1220–1226. DOI: 10.1038/ng.3117.

143. J. Lines, S. Taylor, and A. Bagnall. "HIVE-COTE: The Hierarchical Vote Collective of Transformation-Based Ensembles for Time Series Classification". In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. 2016, pp. 1041–1046. DOI: 10.1109/ICDM.2016.0133.

144. J. Lines and A. Bagnall. "Time series classification with ensembles of elastic distance measures". *Data Mining and Knowledge Discovery* 29:3, 2015, pp. 565–592.

145. C. Lippert, J. Listgarten, Y. Liu, C. M. Kadie, R. I. Davidson, and D. Heckerman. "FaST linear mixed models for genome-wide association studies". *Nature Methods* 8:10, 2011, pp. 833–835. DOI: 10.1038/nmeth.1681.

146. F. Llinares-López, D. G. Grimm, D. A. Bodenham, U. Gieraths, M. Sugiyama, B. Rowan, and K. Borgwardt. "Genome-wide detection of intervals of genetic heterogeneity associated with complex traits". In: *Bioinformatics*. Vol. 31. 12. Oxford University Press, 2015, pp. i240–i249. DOI: 10.1093/bioinformatics/btv263.

147. F. Llinares-Lopez, M. Sugiyama, L. Papaxanthos, and K. Borgwardt. "Fast and Memory-Efficient Significant Pattern Mining via Permutation Testing". *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 725–734. DOI: 10.1145/2783258.2783363.

148. G. Loosli, S. Canu, and C. S. Ong. "Learning SVM in Kreĭn spaces". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38:6, 2015, pp. 1204–1216.

149. M. Lopez-Cruz, J. Crossa, D. Bonnett, S. Dreisigacker, J. Poland, J.-L. Jannink, R. P. Singh, E. Autrique, and G. de los Campos. "Increased prediction accuracy in wheat breeding trials using a marker× environment interaction genomic selection model". *G3: Genes, Genomes, Genetics* 5:4, 2015, pp. 569–582.

150. A. Lorincz, L. Attila Jeni, Z. Szabo, J. F. Cohn, and T. Kanade. "Emotional Expression Classification Using Time-Series Kernels". In: *IEEE CVPR Workshops*. 2013, pp. 889–895.

151. I. Loshchilov and F. Hutter. "SGDR: Stochastic gradient descent with warm restarts". In: *International Conference on Learning Representations*. 2017.

152. R. Luss and A. d'Aspremont. "Support vector machine classification with indefinite kernels". *Mathematical Programming Computation* 1:2, 2009, pp. 97–118.

153. J. Ma, M. K. Yu, S. Fong, K. Ono, E. Sage, B. Demchak, R. Sharan, and T. Ideker. "Using deep learning to model the hierarchical structure and function of a cell". *Nature methods* 15:4, 2018, p. 290.

154. J. MacArthur, E. Bowler, M. Cerezo, L. Gil, P. Hall, E. Hastings, H. Junkins, A. McMahon, A. Milano, J. Morales, Z. MayPendlington, D. Welter, T. Burdett, L. Hindorff, P. Flicek, F. Cunningham, and H. Parkinson. "The new NHGRI-EBI Catalog of published genome-wide association studies (GWAS Catalog)". *Nucleic Acids Research* 45:D1, 2017, pp. D896–D901. DOI: 10.1093/nar/gkw1133.

155. D. J. C. MacKay. "Bayesian non-linear modelling for the energy prediction competition". *ASHRAE Transactions* 100, 1994, pp. 1053–1062.

156. T. F. MacKay, S. Richards, E. A. Stone, A. Barbadilla, J. F. Ayroles, D. Zhu, S. Casillas, Y. Han, M. M. Magwire, J. M. Cridland, M. F. Richardson, R. R. Anholt, M. Barrón, C. Bess, K. P. Blankenburg, M. A. Carbone, D. Castellano, L. Chaboub, L. Duncan, Z. Harris, M. Javaid, J. C. Jayaseelan, S. N. Jhangiani, K. W. Jordan, F. Lara, F. Lawrence, S. L. Lee, P. Librado, R. S. Linheiro, R. F. Lyman, A. J. MacKey, M. Munidasa, D. M. Muzny, L. Nazareth, I. Newsham, L. Perales, L. L. Pu, C. Qu, M. Ràmia, J. G. Reid, S. M. Rollmann, J. Rozas, N. Saada, L. Turlapati, K. C. Worley, Y. Q. Wu, A. Yamamoto, Y. Zhu, C. M. Bergman, K. R. Thornton, D. Mittelman, and R. A. Gibbs. "The Drosophila melanogaster Genetic Reference Panel". *Nature* 482:7384, 2012, pp. 173–178. DOI: 10.1038/nature10811.

157. L. N. Magner. *A history of the life sciences, revised and expanded*. CRC Press, 2002.

158. M. Maimaitijiang, V. Sagan, P. Sidike, S. Hartling, F. Esposito, and F. B. Fritschi. "Soybean yield prediction from UAV using multimodal data fusion and deep learning". *Remote Sensing of Environment* 237, 2020, p. 111599.

159. S. Maji, A. C. Berg, and J. Malik. "Classification using intersection kernel support vector machines is efficient". In: *IEEE CVPR*. 2008, pp. 1–8.

160. A. Mallasto, J. Frellsen, W. Boomsma, and A. Feragen. "(q,p)-Wasserstein GANs: Comparing Ground Metrics for Wasserstein GANs". *arXiv e-prints*, arXiv:1902.03642, 2019, arXiv:1902.03642. arXiv: 1902.03642 [cs.LG].

161. S. Mandt, F. Wenzel, S. Nakajima, J. Cunningham, C. Lippert, and M. Kloft. "Sparse probit linear mixed model". *Machine Learning* 106:9-10, 2017, pp. 1621–1642. DOI: 10.1007/s10994-017-5652-6.

162. T. A. Manolio, F. S. Collins, N. J. Cox, D. B. Goldstein, L. A. Hindorff, D. J. Hunter, M. I. McCarthy, E. M. Ramos, L. R. Cardon, A. Chakravarti, J. H. Cho, A. E. Guttmacher, A. Kong, L. Kruglyak, E. Mardis, C. N. Rotimi, M. Slatkin, D. Valle, A. S. Whittemore, M. Boehnke, A. G. Clark, E. E. Eichler, G. Gibson, J. L. Haines, T. F. C. Mackay, S. A. McCarroll, and P. M. Visscher. "Finding the missing heritability of complex diseases". *Nature* 461:7265, 2009, pp. 747–753. DOI: 10.1038/nature08494. URL: http://www.nature.com/doifinder/10.1038/nature08494.

163. J. Marchini, P. Donnelly, and L. R. Cardon. "Genome-wide strategies for detecting multiple loci that influence complex diseases". *Nature genetics* 37:4, 2005, pp. 413–417.

164. Á. Maresma, M. Ariza, E. Martínez, J. Lloveras, and J. A. Martínez-Casasnovas. "Analysis of vegetation indices to determine nitrogen application and yield prediction in maize (Zea mays L.) from a standard UAV service". *Remote Sensing* 8:12, 2016, p. 973.

165. P.-F. Marteau and S. Gibet. "On Recursive Edit Distance Kernels With Application to Time Series Classification". *IEEE Transactions on Neural Networks and Learning Systems* 26:6, 2015, pp. 1121–1133.

166. A. G. d. G. Matthews, M. van der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. León-Villagrá, Z. Ghahramani, and J. Hensman. "GPflow: A Gaussian process library using TensorFlow". *Journal of Machine Learning Research* 18:40, 2017, pp. 1–6. URL: http://jmlr.org/papers/v18/16-537.html.

167. M. I. McCarthy, G. R. Abecasis, L. R. Cardon, D. B. Goldstein, J. Little, J. P. Ioannidis, and J. N. Hirschhorn. "Genome-wide association studies for complex traits: consensus, uncertainty and challenges". *Nature reviews genetics* 9:5, 2008, pp. 356–369.

168. M. Meijón, S. B. Satbhai, T. Tsuchimatsu, and W. Busch. "Genome-wide association study using cellular traits identifies a new regulator of root development in Arabidopsis". *Nature Genetics* 46:1, 2014, pp. 77–81. DOI: 10.1038/ng.2824.

169. Members of the Multinational Arabidopsis Steering Committee. URL: https://www.nsf.gov/pubs/2002/bio0202/model.htm.

170. J. Menche, A. Sharma, M. Kitsak, S. D. Ghiassian, M. Vidal, J. Loscalzo, and A.-L. Barabási. "Uncovering disease-disease relationships through the incomplete interactome". *Science* 347:6224, 2015, p. 1257601.

171. J. G. Monroe, T. Powell, N. Price, J. L. Mullen, A. Howard, K. Evans, J. T. Lovell, and J. K. McKay. "Drought adaptation in arabidopsis thaliana by extensive genetic loss-of-function". *eLife* 7, 2018. DOI: 10.7554/eLife.41038.

172. A. Montesinos-López, O. A. Montesinos-López, J. Cuevas, W. A. Mata-López, J. Burgueño, S. Mondal, J. Huerta, R. Singh, E. Autrique, L. González-Pérez, et al. "Genomic Bayesian functional regression models with interactions for predicting wheat grain yield using hyper-spectral image data". *Plant Methods* 13:1, 2017, p. 62.

173. C. Morris, N. M. Kriege, K. Kersting, and P. Mutzel. "Faster kernels for graphs with continuous attributes via hashing". In: *Proceedings of the 16th IEEE International Conference on Data Mining*. 2016, pp. 1095–1100.

174. P. Nemenyi. "Distribution-free multiple comparisons (PhD Dissertation, Princeton University, 1963)". *Dissertation Abstracts International* 25:2, 1963, p. 1233.

175. M. Neumann, R. Garnett, C. Bauckhage, and K. Kersting. "Propagation kernels: efficient graph kernels from propagated information". *Machine Learning* 102:2, 2016, pp. 209–245.

176. Y. Ni, D. Aghamirzaie, H. Elmarakeby, E. Collakova, S. Li, R. Grene, and L. S. Heath. "A machine learning approach to predict gene regulatory networks in seed development in Arabidopsis". *Frontiers in Plant Science* 7:DECEMBER2016, 2016. DOI: 10.3389/fpls.2016.01936.

177. D. Oglic and T. Gärtner. "Learning in reproducing kernel Krein spaces". In: *Proceedings of the 35th International Conference on Machine Learning*. 2018, pp. 3859–3867.

178. C. S. Ong, X. Mary, S. Canu, and A. J. Smola. "Learning with non-positive kernels". In: *Proceedings of the 21st International Conference on Machine Learning*. 2004.

179. X. E. Pantazi, D. Moshou, T. Alexandridis, R. L. Whetton, and A. M. Mouazen. "Wheat yield prediction using machine learning and advanced sensing techniques". *Computers and Electronics in Agriculture* 121, 2016, pp. 57–65.

180. M. Parry, C. Rosenzweig, and M. Livermore. "Climate change, global food supply and risk of hunger". *Philosophical Transactions of the Royal Society B: Biological Sciences* 360:1463, 2005, pp. 2125–2138.

181. B. Pasaniuc and A. L. Price. "Dissecting the genetics of complex traits using summary association statistics". *Nature Reviews Genetics* 18:2, 2016, pp. 117–127. DOI: 10.1038/nrg.2016.142. URL: http://www.nature.com/doifinder/10.1038/nrg.2016.142.

182. B. Pasaniuc, N. Zaitlen, H. Shi, G. Bhatia, A. Gusev, J. Pickrell, J. Hirschhorn, D. P. Strachan, N. Patterson, and A. L. Price. "Fast and accurate imputation of summary statistics enhances evidence of functional enrichment". *Bioinformatics (Oxford, England)* 30:20, 2014, pp. 2906–2914. DOI: 10.1093/bioinformatics/btu416.

183. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. 2019, pp. 8024–8035.

184. J. Pearl et al. "Causal inference in statistics: An overview". *Statistics surveys* 3, 2009, pp. 96–146.

185. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. "Scikit-learn: Machine learning in Python". *Journal of machine learning research* 12:Oct, 2011, pp. 2825–2830.

186. G. Peyré, M. Cuturi, et al. "Computational optimal transport". *Foundations and Trends® in Machine Learning* 11:5-6, 2019, pp. 355–607.

187. J. K. Pickrell, T. Berisa, J. Z. Liu, L. Ségurel, J. Y. Tung, and D. A. Hinds. "Detection and interpretation of shared genetic influences on 42 human traits". *Nature Genetics* 48:7, 2016, pp. 709–717. DOI: 10.1038/ng.3570. URL: http://www.nature.com/doifinder/10.1038/ng.3570.

188. J. Poland, J. Endelman, J. Dawson, J. Rutkoski, S. Wu, Y. Manes, S. Dreisigacker, J. Crossa, H. Sánchez-Villeda, M. Sorrells, et al. "Genomic selection in wheat breeding using genotyping-by-sequencing". *The Plant Genome* 5:3, 2012, pp. 103–113.

189. A. L. Price, N. J. Patterson, R. M. Plenge, M. E. Weinblatt, N. A. Shadick, and D. Reich. "Principal components analysis corrects for stratification in genome-wide association studies". *Nature genetics* 38:8, 2006, pp. 904–909.

190. A. L. Price, N. A. Zaitlen, D. Reich, and N. Patterson. "New approaches to population stratification in genome-wide association studies". *Nature Reviews Genetics* 11:7, 2010, pp. 459–463. DOI: 10.1038/nrg2813.

191. N. Quarmby, M. Milnes, T. Hindle, and N Silleos. "The use of multi-temporal NDVI measurements from AVHRR data for crop yield estimation and prediction". *International Journal of Remote Sensing* 14:2, 1993, pp. 199–210.

192. J. Rabin, G. Peyré, J. Delon, and M. Bernot. "Wasserstein barycenter and its application to texture mixing". In: *International Conference on Scale Space and Variational Methods in Computer Vision*. 2011, pp. 435–446.

193. B. Rakitsch, C. Lippert, O. Stegle, and K. Borgwardt. "A Lasso multi-marker mixed model for association mapping with population structure correction". *Bioinformatics* 29:2, 2013, pp. 206–214. DOI: 10.1093/bioinformatics/bts669.

194. T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, and E. Keogh. "Searching and mining trillions of time series subsequences under dynamic time warping". In: *KDD*. 2012, pp. 262–270.

195. J. Ramon and L. De Raedt. "Multi instance neural networks". In: *Proceedings of the ICML-2000 workshop on attribute-value and relational learning*. 2000, pp. 53–60.

196. C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, USA, 2006, p. 248. ISBN: 0-262-18253-X.

197. G. P. Redei. "Arabidopsis as a Genetic Tool". *Annual Review of Genetics* 9:1, 1975, pp. 111–127. DOI: 10.1146/annurev.ge.09.120175.000551. URL: http://www.annualreviews.org/doi/10.1146/annurev.ge.09.120175.000551.

198. E. A. Regan et al. "Genetic epidemiology of COPD (COPDGene) study design". *COPD: Journal of Chronic Obstructive Pulmonary Disease* 7:1, 2011, pp. 32–43.

199. B. Rieck, C. Bock, and K. Borgwardt. "A Persistent Weisfeiler–Lehman Procedure for Graph Classification". In: *Proceedings of the 36th International Conference on Machine Learning*. 2019, pp. 5448–5458.

200. B. Rieck, M. Togninalli, C. Bock, M. Moor, M. Horn, T. Gumbsch, and K. Borgwardt. "Neural Persistence: A Complexity Measure for Deep Neural Networks Using Algebraic Topology". In: *International Conference on Learning Representations*. 2019. URL: https://openreview.net/forum?id=ByxkijC5FQ.

201. R. M. Rifkin and R. A. Lippert. "Notes on regularized least squares", 2007.

202. S. Ripatti, E. Tikkanen, M. Orho-Melander, A. S. Havulinna, K. Silander, A. Sharma, C. Guiducci, M. Perola, A. Jula, J. Sinisalo, et al. "A multilocus genetic risk score for coronary heart disease: case-control and prospective cohort analyses". *The Lancet* 376:9750, 2010, pp. 1393–1400.

203. J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso. "Rotation Forest: A New Classifier Ensemble Method". *IEEE transactions on pattern analysis and machine intelligence* 28:10, 2006, pp. 1619–1630.

204. Y. Rubner, C. Tomasi, and L. J. Guibas. "The Earth Mover's Distance as a metric for image retrieval". *International Journal of Computer Vision* 40:2, 2000, pp. 99–121.

205. S. Rüping. *SVM kernels for time series analysis*. Technical report 43. Technical University of Dortmund, 2001.

206. P. Schäfer. "The BOSS is concerned with time series classification in the presence of noise". *Data Mining and Knowledge Discovery* 29:6, 2015, pp. 1505–1530.

207. B. Schölkopf. "The kernel trick for distances". In: *Advances in Neural Information Processing Systems 13*. 2001, pp. 301–307.

208. B. Schölkopf and A. J. Smola. *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT Press, 2002.

209. V. Segura, B. J. Vilhjálmsson, A. Platt, A. Korte, Ü. Seren, Q. Long, and M. Nordborg. "An efficient multi-locus mixed-model approach for genome-wide association studies in structured populations". *Nature Genetics* 44:7, 2012, pp. 825–830. DOI: 10.1038/ng.2314.

210. P. Senin and S. Malinchik. "SAX-VSM: Interpretable Time Series Classification Using SAX and Vector Space Model". In: *2013 IEEE 13th international conference on data mining*. IEEE. 2013, pp. 1175–1180.

211. A. W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Žídek, A. W. Nelson, A. Bridgland, et al. "Improved protein structure prediction using potentials from deep learning". *Nature*, 2020, pp. 1–5.

212. Ü. Seren, D. Grimm, J. Fitz, D. Weigel, M. Nordborg, K. Borgwardt, and A. Korte. "AraPheno: a public database for Arabidopsis thaliana phenotypes". *Nucleic Acids Research* 45:D1, 2017, pp. D1054–D1059. DOI: 10.1093/nar/gkw986. URL: https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkw986.

213. B. Servin and M. Stephens. "Imputation-based analysis of association studies: Candidate regions and quantitative traits". *PLoS Genetics* 3:7, 2007, pp. 1296–1308. DOI: 10.1371/journal.pgen.0030114.

214. N. Shervashidze and K. Borgwardt. "Fast subtree kernels on graphs". In: *Advances in Neural Information Processing Systems 22*. 2009, pp. 1660–1668.

215. N. Shervashidze, P. Schweitzer, E. J. v. Leeuwen, K. Mehlhorn, and K. M. Borgwardt. "Weisfeiler-Lehman graph kernels". *Journal of Machine Learning Research* 12, 2011, pp. 2539–2561.

216. O. Shin-Ichi. "Barycenters in Alexandrov spaces of curvature bounded below". *Advances in Geometry* 14:4, 2012, pp. 571–587.

217. D. Singh, X. Wang, U. Kumar, L. Gao, M. Noor, M. Imtiaz, R. P. Singh, and J. Poland. "High-throughput phenotyping enabled genetic dissection of crop lodging in wheat". *Frontiers in plant science* 10, 2019, p. 394.

218. S. Sivakumaran, F. Agakov, E. Theodoratou, J. G. Prendergast, L. Zgaga, T. Manolio, I. Rudan, P. McKeigue, J. F. Wilson, and H. Campbell. "Abundant pleiotropy in human complex diseases and traits". *The American Journal of Human Genetics* 89:5, 2011, pp. 607–618.

219. J. Snape. "Predicting the frequencies of transgressive segregants for yield and yield components in wheat". *Theoretical and Applied Genetics* 62:2, 1982, pp. 127–134.

220. M. Stas, J. Van Orshoven, Q. Dong, S. Heremans, and B. Zhang. "A comparison of machine learning algorithms for regional wheat yield prediction using NDVI time series of SPOT-VGT". In: *2016 Fifth International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*. IEEE. 2016, pp. 1–5.

221. S. S. Stevens. "On the theory of scales of measurement". *Science* 103:2684, 1946, pp. 677–680.

222. C. Sudlow, J. Gallacher, N. Allen, V. Beral, P. Burton, J. Danesh, P. Downey, P. Elliott, J. Green, M. Landray, B. Liu, P. Matthews, G. Ong, J. Pell, A. Silman, A. Young, T. Sprosen, T. Peakman, and R. Collins. "UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age". *PLoS Medicine* 12:3, 2015. DOI: 10.1371/journal.pmed.1001779.

223. M. Sugiyama, C. A. Azencott, D. Grimm, Y. Kawahara, and K. Borgwardt. "Multi-task feature selection on multiple networks via maximum flows". In: *SIAM International Conference on Data Mining 2014, SDM 2014*. Vol. 1. Society for Industrial and Applied Mathematics, 2014, pp. 199–207. ISBN: 9781510811515. DOI: 10.1137/1.9781611973440.23.

224. M. Sugiyama, M. E. Ghisu, F. Llinares-López, and K. Borgwardt. "graphkernels: R and Python packages for graph comparison". *Bioinformatics* 34:3, 2018, pp. 530–532.

225. C. Suo, T. Toulopoulou, E. Bramon, M. Walshe, M. Picchioni, R. Murray, and J. Ott. "Analysis of multiple phenotypes in genome-wide genetic mapping studies". *BMC Bioinformatics* 14, 2013. DOI: 10.1186/1471-2105-14-151.

226. *TAIR - About Arabidopsis*. URL: https://www.arabidopsis.org/portals/education/aboutarabidopsis.jsp#hist.

227. M. Tester and P. Langridge. "Breeding technologies to increase crop production in a changing world". *Science* 327:5967, 2010, pp. 818–822.

228. R. Tibshirani. "Regression shrinkage and selection via the lasso". *Journal of the Royal Statistical Society: Series B (Methodological)* 58:1, 1996, pp. 267–288.

229. M. Togninalli, E. Ghisu, F. Llinares-López, B. Rieck, and K. Borgwardt. "Wasserstein Weisfeiler-Lehman Graph Kernels". In: *Advances in Neural Information Processing Systems*. 2019, pp. 6436–6446.

230. M. Togninalli, D. Roqueiro, I. COPDGene, and K. M. Borgwardt. "Accurate and adaptive imputation of summary statistics in mixed-ethnicity cohorts". *Bioinformatics* 34:17, 2018, pp. i687–i696.

231. M. Togninalli, Ü. Seren, J. A. Freudenthal, J. G. Monroe, D. Meng, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "AraPheno and the AraGWAS Catalog 2020: a major database update including RNA-Seq and knockout mutation data for Arabidopsis thaliana". *Nucleic Acids Research* 48:D1, 2020, pp. D1063–D1068. DOI: 10.1093/nar/gkz925. URL: https://academic.oup.com/nar/article/48/D1/D1063/5603218.

232. M. Togninalli, Ü. Seren, J. A. Freudenthal, J. G. Monroe, D. Meng, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "AraPheno and the AraGWAS Catalog 2020: a major database update including RNA-Seq and knockout mutation data for Arabidopsis thaliana". *Nucleic acids research*

48:D1, 2020, pp. D1063–D1068. DOI: 10.1093/nar/gkz925. URL: https://academic.oup.com/nar/article/48/D1/D1063/5603218.

233. M. Togninalli, Ü. Seren, D. Meng, J. Fitz, M. Nordborg, D. Weigel, K. Borgwardt, A. Korte, and D. G. Grimm. "The AraGWAS Catalog: A curated and standardized Arabidopsis thaliana GWAS catalog". *Nucleic Acids Research* 46:D1, 2018, pp. D1150–D1156. DOI: 10.1093/nar/gkx954.

234. M. Togninalli, X. Wang, J. Poland, and K. Borgwardt. "Deep learning enables accurate grain yield prediction using image and genotype information". Unpublished Manuscript. 2020.

235. K. Turner, Y. Mileyko, S. Mukherjee, and J. Harer. "Fréchet Means for Distributions of Persistence Diagrams". *Discrete & Computational Geometry* 52, 2014, pp. 44–70.

236. N. M. (US) and National Center for Biotechnology Information. *GenBank [Online]*. Accessed: 2020-04-26. 2020.

237. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

238. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. "Graph Attention Networks". In: *International Conference on Learning Representations*. 2018. URL: https://openreview.net/forum?id=rJXMpikCZ.

239. J.-P. Vert. "The optimal assignment kernel is not positive definite". *arXiv preprint arXiv:0801.4061*, 2008.

240. C. Villani. *Optimal transport: Old and new*. Vol. 338. Comprehensive Studies in Mathematics. Springer, Heidelberg, Germany, 2008.

241. S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt. "Graph kernels". *Journal of Machine Learning Research* 11, 2010, pp. 1201–1242.

242. P. M. Visscher. "Sizing up human height variation". *Nature genetics* 40:5, 2008, pp. 489–490.

243. G. Wachman, R. Khardon, P. Protopapas, and C. R. Alcock. "Kernels for Periodic Time Series Arising in Astronomy". In: *Machine Learning and Knowledge Discovery in Databases*. Springer, Heidelberg, Germany, 2009, pp. 489–505.

244. M. Wainberg, N. Sinnott-Armstrong, N. Mancuso, A. N. Barbeira, D. A. Knowles, D. Golan, R. Ermel, A. Ruusalepp, T. Quertermous, K. Hao, J. L. Björkegren, H. K. Im, B. Pasaniuc, M. A. Rivas, and A. Kundaje. "Opportunities and challenges for transcriptome-wide association studies". *Nature Genetics* 51:4, 2019, pp. 592–599. DOI: 10.1038/s41588-019-0385-z.

245. L. Wang, Y. Tian, X. Yao, Y. Zhu, and W. Cao. "Predicting grain yield and protein content in wheat by fusing multi-sensor and multi-temporal remote-sensing images". *Field Crops Research* 164, 2014, pp. 178–188.

246. X. Wang, D. Singh, S. Marla, G. Morris, and J. Poland. "Field-based high-throughput phenotyping of plant height in sorghum using different sensing technologies". *Plant Methods* 14:1, 2018, p. 53.

247. Z. Wang, W. Ya, and T. Oates. "Time series classification from scratch with deep neural networks: A strong baseline". In: *International Joint Conference on Neural Networks (IJCNN)*. 2017, pp. 1578–1585.

248. D. Welter, J. MacArthur, J. Morales, T. Burdett, P. Hall, H. Junkins, A. Klemm, P. Flicek, T. Manolio, L. Hindorff, and H. Parkinson. "The NHGRI GWAS Catalog, a curated resource of SNP-trait associations". *Nucleic Acids Research* 42:D1, 2014, pp. 1001–1006. DOI: 10.1093/nar/gkt1229.

249. X. Wen and M. Stephens. "Using linear predictors to impute allele frequencies from summary or pooled genotype data". *Annals of Applied Statistics* 4:3, 2010, pp. 1158–1182. DOI: 10.1214/10-AOAS338.

250. J. E. Wigginton, D. J. Cutler, and G. R. Abecasis. "A note on exact tests of Hardy-Weinberg equilibrium". *The American Journal of Human Genetics* 76:5, 2005, pp. 887–893.

251. G. Wu, E. Y. Chang, and Z. Zhang. "An analysis of transformation on non-positive semidefinite similarity matrix for kernel machines". In: *ICML*. 2005.

252. L. Wu, I. En-Hsu Yen, F. Xu, P. Ravikumar, and M. Witbrock. "D2KE: From Distance to Kernel and Embedding". *arXiv e-prints*, arXiv:1802.04956, 2018, arXiv:1802.04956. arXiv: 1802.04956 [stat.ML].

253. Q. Wu, Y. Jin, Y. Bao, Q. Hai, R. Yan, B. Chen, H. Zhang, B. Zhang, Z. Li, X. Li, et al. "Comparison of two inversion methods for leaf area index using HJ-1 satellite data in a temperate meadow steppe". *International Journal of Remote Sensing* 36:19-20, 2015, pp. 5192–5207.

254. Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande. "MoleculeNet: a benchmark for molecular machine learning". *Chemical science* 9:2, 2018, pp. 513–530.

255. H. Xu, D. Luo, H. Zha, and L. C. Duke. "Gromov–Wasserstein Learning for Graph Matching and Node Embedding". In: *Proceedings of the 36th International Conference on Machine Learning*. 2019, pp. 6932–6941.

256. K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. "Show, attend and tell: Neural image caption generation with visual attention". In: *International conference on machine learning*. 2015, pp. 2048–2057.

257. Z. Xu, Q. Duan, S. Yan, W. Chen, M. Li, E. Lange, and Y. Li. "DISSCO: Direct imputation of summary statistics allowing covariates". *Bioinformatics* 31:15, 2015, pp. 2434–2442. DOI: 10.1093/bioinformatics/btv168.

258. P. Yanardag and S. Vishwanathan. "Deep graph kernels". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2015, pp. 1365–1374.

259. J. Yang, B. Benyamin, B. P. McEvoy, S. Gordon, A. K. Henders, D. R. Nyholt, P. A. Madden, A. C. Heath, N. G. Martin, G. W. Montgomery, et al. "Common SNPs explain a large proportion of the heritability for human height". *Nature genetics* 42:7, 2010, p. 565.

260. L. Ye and E. Keogh. "Time Series Shapelets: A New Primitive for Data Mining". In: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, New York, NY, USA, 2009, pp. 947–956.

261. C.-C. M. Yeh, Y. Zhu, L. Ulanova, N. Begum, Y. Ding, H. A. Dau, D. F. Silva, A. Mueen, and E. Keogh. "Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets". In: *IEEE International Conference on Data Mining (ICDM)*. 2016, pp. 1317–1322.

262. Y. Ying, C. Campbell, and M. Girolami. "Analysis of SVM with Indefinite Kernels". In: *Advances in Neural Information Processing Systems (NIPS)*. Vol. 22. Curran Associates, Inc., 2009, pp. 2205–2213.

263. J. You, X. Li, M. Low, D. Lobell, and S. Ermon. "Deep gaussian process for crop yield prediction based on remote sensing data". In: *Thirty-First AAAI Conference on Artificial Intelligence*. 2017.

264. M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. "Deep sets". In: *Advances in neural information processing systems*. 2017, pp. 3391–3401.

265. X. Zhang, P. Pérez-Rodríguez, J. Burgueño, M. Olsen, E. Buckler, G. Atlin, B. M. Prasanna, M. Vargas, F. San Vicente, and J. Crossa. "Rapid cycling genomic selection in a multiparental tropical maize population". *G3: Genes, Genomes, Genetics* 7:7, 2017, pp. 2315–2326.

266. K. Zhao, C. W. Tung, G. C. Eizenga, M. H. Wright, M. L. Ali, A. H. Price, G. J. Norton, M. R. Islam, A. Reynolds, J. Mezey, A. M. McClung, C. D. Bustamante, and S. R. McCouch. "Genome-wide association mapping reveals a rich genetic architecture of complex traits in Oryza sativa". *Nature Communications* 2:1, 2011. DOI: 10.1038/ncomms1467.

267. J. Zhou and O. G. Troyanskaya. "Predicting effects of noncoding variants with deep learning–based sequence model". *Nature methods* 12:10, 2015, pp. 931–934.

268. J. J. Zhou, M. H. Cho, C. Lange, S. Lutz, E. K. Silverman, and N. M. Laird. "Integrating multiple correlated phenotypes for genetic association analysis by maximizing heritability". *Human Heredity* 79:2, 2015, pp. 93–104. DOI: 10.1159/000381641.

269. H. Zou and T. Hastie. “Regularization and variable selection via the elastic net”. *Journal of the royal statistical society: series B (statistical methodology)* 67:2, 2005, pp. 301–320.