Feature selection by Information Imbalance optimization: Clinics, molecular modeling and ecology

PhD course in Physics and Chemistry of Biological Systems

Advisor:
Prof. Alessandro Laio

Candidate:
Romina Wild

Academic year 2023/2024

Preface

This thesis is the result of the work I have done during my doctoral studies at SISSA under the supervision of Prof. Alessandro Laio. Its main references are:

- [1] Romina Wild, Emanuela Sozio, Riccardo G. Margiotta, Fabiana Dellai, Angela Acquasanta, Fabio Del Ben, Carlo Tascini, Francesco Curcio & Alessandro Laio. **Maximally informative feature selection using Information Imbalance: Application to COVID-19 severity prediction**. *Sci Rep* **14**, 10744 (2024). https://doi.org/10.1038/s41598-024-61334-6

- [2] Romina Wild, Vittorio Del Tatto, Felix Wodaczek, Bingqing Cheng & Alessandro Laio. **Automatic feature selection and weighting using Differentiable Information Imbalance**. *Accepted for publication in Nature Communications on 04 November 2024*. https://arxiv.org/abs/2411.00851.

- [3] Aldo Glielmo, Iuri Macocco, Diego Doimo, Matteo Carli, Claudio Zeni, Romina Wild, Maria d'Errico, Alex Rodriguez & Alessandro Laio. **DADApy: Distance-based analysis of data-manifolds in Python**. *Patterns* **3**(10), 100589 (2022). https://doi.org/10.1016/j.patter.2022.100589

Abstract

In feature selection current methods are often limited by the types and dimensions of data they can handle. Supervised methods, in particular, are rigid regarding their target space, typically requiring it to be one-dimensional and of a specific type (e.g. continuous or categorical). This thesis introduces feature selection methods which mitigate these limitations using a statistic called the Information Imbalance. This method identifies a low-dimensional subset of input features that best preserves pairwise distance relations found in the target feature space by ranking nearest neighbors. First, we derive a weighted Information Imbalance approach to handle class-imbalanced medical data, along with an optimization routine capable of managing missing data. The study on COVID-19 severity prediction showcased this approach, successfully isolating a 13-feature subset from a pool of roughly 150 features. This subset outperformed traditional feature selection methods in subsequent predictions for patient severity. We then introduce an Information Imbalance variant that can handle binary and categorical data. We benchmarked this approach on Amazon Rainforest biodiversity data. By quantifying the relative information content of continuous features, like average temperature, and categorical features, like the label of the region in which data are recorded, this method identifies plausible predictors of species richness and asymmetric information even between variables which are not correlated. Finally, we introduced a differentiable variant of the Information Imbalance, implemented in the easy-to-use Python package, DADApy. Differentiable Information Imbalance (DII) optimizes relative feature weights via gradient descent, addressing combinatorial challenges of high-dimensional data. The weights correct for different units of measure and relative importance and allow for feature selection through sparsity-inducing optimization approaches. In molecular dynamics simulations, this method reduced the feature set to three collective variables effectively describing a beta-pin peptide. In another application on machine learning potentials, the input feature space was compressed, reducing run time while preserving accuracy.

Contents

Chapter 1

Introduction

Background

We live in a data-dominated world. In the last decades, the amount of data has grown exponentially: In 1990, data was measured in petabytes (1 petabyte = 1 million gigabytes), in the 2000s it grew to exabytes (1 exabyte = 1 billion gigabytes), and now in the 2020s, data volumes are measured in zettabytes (1 zettabyte = 1 trillion gigabytes) [4, 5, 6]. By now, the internet is supported by an unmeasurable amount of servers globally, on the order of hundreds of millions, and data centers consume more electricity than countries [7]. Information and computing technology is projected to account for 20 % or more of the global electricity demand [4] by 2030, over 8,000 TWh, more than the current electricity consumption of the EU, the USA and India combined [8, 9, 10]. Even if we optimize algorithms, architecture and hardware, the operational energy reduction is estimated to be only approximately 25%, not enough to offset the environmental impact from this exponential growth of data and AI [6].

While humanity urgently needs to reduce the growing hunger of data processing, this enormous ecological footprint comes paired with advancements. For one, we can now better than ever quantify the evolution of our planet, as done *e.g.* by the International Panel on Climate Change (IPCC), which analyzes and summarizes incredible amounts data into their IPCC reports [11]. Data is also used to understand diseases [12], mitigate pandemics [13, 14], and improve technologies and scientific methods [15]. On the other end of the spectrum, what might be termed 'dark data science' exploits vast amounts of information for purposes like consumer surveillance [16], online marketing [17], algorithmic political manipulation [18], and digital inequality [19]. As data grows, so do its applications, with each fueling the other's expansion.

The data sets grow in "length" and "width", meaning in the number of sam-

ples and in the number of features. This growth is due to better data collection capabilities, advanced machine learning techniques and more storage possibilities. The "wide" data sets, with many features, are termed "high-dimensional", and have increasingly become a focus of research. The meaning of high-dimensional varies between fields: In molecular dynamics (MD) simulations we have between tens to 1000s potential collective variables (CVs) [20, 3]. Features from genetic sequencing [21, 22] or the parameters in neural networks such as language models [23] are commonly in the 10.000s and larger. Ecological databases can include tens or hundreds of biotic or abiotic features, or even 10.000s of features *e.g.* for species abundance data of the Amazon rain forest [24, 25].

The width of these data sources leads to practical problems: Non-interpretability and overfitting of models are the most obvious ones. Very often, most of the features defining a data point are redundant, irrelevant, or affected by noise. In these cases one can employ **feature selection**, discarding all except a small subset of relevant features to improve model performance and increase interpretability. Feature selection is everywhere. It takes many forms and shapes and may be implicit or explicit. The graphic in the overview on feature selection gives an idea of the field.

Executive summary

Chapters 2 - 6 of this thesis are structured as follows:

- First, an overview over feature selection (chapter 2) is given and our main working tool, the Information Imbalance, is introduced (chapter 3).
- Then we discuss the problems we addressed to use the Information Imbalance in class-imbalanced classification tasks in a clinical data set (chapter 4), and to categorical feature spaces in an ecological data set (chapter 5).
- Furthermore, we present the main technical contribution of this thesis, a "Differentiable Information Imbalance" (DII) which is optimizable with gradient descent and allows finding optimal feature sets and the features' relative weights (chapter 6).
- Finally, we draw conclusions and emphazise the main findings (chapter 7).

Information Imbalance

In this thesis we develop a class of feature selection methods based on the **Information Imbalance**, and describe attempts to improve performance in different

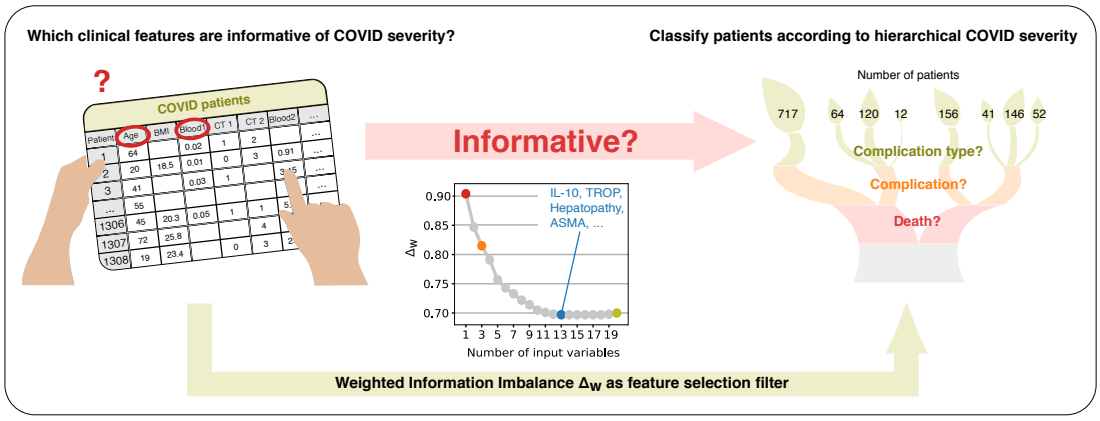
settings of feature selection. Working on improving feature selection methods is, in our opinion, timely and important, since many available methods are narrow in their applicability, as described in chapter 2. For example, many feature selection methods are designed for regression or classification, assuming already an underlying relationship of specific type between input and output (embedded methods), often limited to a single prediction target and certain data types. Wrapper methods, which use the downstream model to iteratively evaluate the performance of feature subsets, are inefficient, since their paradigm, as we will see, leads to a combinatorial explosion in the number of tests. The more universal filter methods (which do not assume an underlying model), are often limited to one-dimensional, sometimes discrete, ground truths (label / target) data, and no flexible multi-target filter methods are available in user-friendly software packages.

We try to address the problem from a different angle, where we allow as much flexibility as possible in the data type and dimensionalities of input and ground truth data sets. Our different philosophy for feature selection is inspired by a simple principle, the one underlying the Information Imbalance (chapter 3). Qualitatively, we try reproduce the neighborhood relationship using the smallest possible number of variables, which may or may not include the target space variables. We retain only the most informative features, such that they produce nearest neighbor relationships which are very similar to the ones in the target space. In essence, the algorithm searches for the best low-dimensional neighborhood clone of the target space. Both, the target space and the input space can have a wide range of dimensions, from extremely high-dimensional to low-dimensional (or even one-dimensional).

Even though we aim to develop an universal, one-fits-all feature selection algorithm, we had to adapt to various use cases. Throughout this thesis, different variants of the Information Imbalance statistic are developed, to solve various issues of feature selection with different data sources.

Unbalanced class prediction with categorical variables and missing data in medicine

In chapter 4 we investigate feature selection based on a variant of Information Imbalance to predict COVID-19 severity using a clinical dataset with significant missing data and categorical and binary features. The ground truth data in this case consisted in 14 binary features which encoded the patient fate. Together with the medical professionals, they were organized into a severity tree, resulting in a tree-specific target distance space and eight unbalanced severity classes of patient fate, which we aimed to predict. Information Imbalance was extended to include class weights to compensate for the unbalance in class populations.



Out of an original set of ∼150 features across 1300 patients, measurable upon hospitalization of any patient, an optimal 13-feature subset was identified, yielding high predictive accuracy for disease outcomes. That combination of features suggested on the one hand a systemic inflammation and autoimmunity, signaled by neutrophils and autoantibodies, and on the other hand an immune paralysis and anti-inflammatory effort. However, due to missing values, these features were only jointly available for 102 patients. To address this, patient-specific optimal n-plets were developed, which allowed prediction of disease severity even in patients without full feature sets. Although this approach reduced predictive performance slightly, it still achieved a meaningful accuracy for predicting severe outcomes.

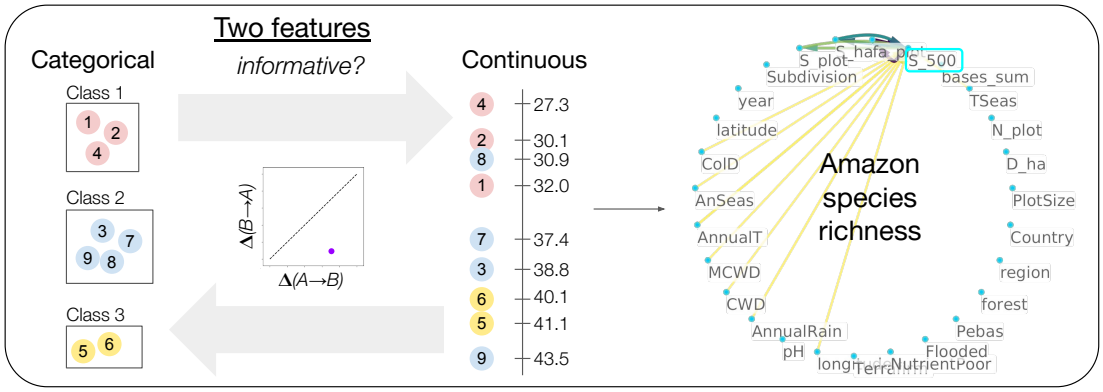
In this chapter we also introduce a metric to assess the intrinsic importance of each feature independent of its availability. This identified several important but underrepresented biomarkers—such as IL-6, direct bilirubin (BILD), and glycated hemoglobin (HBA1CM)—that are crucial for COVID-19 severity prediction but were frequently missing in the dataset. We advocate for enhanced data collection especially of those of the intrinsically important features which are severely undersampled, to improve future predictive models.

The study found that Weighted Information Imbalance feature selection, combined with k-NN or support vector classifiers, outperformed traditional feature selection methods in identifying minority class patients at high risk for severe outcomes. Notably, this approach did not require data imputation, making it adaptable to real-world clinical datasets with incomplete information. The recommended feature sampling schemes could improve patient triaging and resource allocation in clinical settings. The optimal 13 features provide, together with a classifier, a way to assess patient fate a-priori, especially on the coarsest level where only mild *vs.* dangerous disease progression is predicted.

The database included numerous categorical variables, resulting in degenerate values and distances. In this specific research project we addressed this issue by

(a) creating a specific severity tree distance as the target for the output space and (b) adding small random values to the degenerate values in the input space. Constructing case-specific distances is not always feasible, and the chapter 5 provides a more systematic approach to handling categorical variables. To manage the combinatorial explosion of enumerating all potential feature tuples, in this first project we applied a beam search heuristic; an automated approach to this challenge is proposed in chapter 6.

Treating categorical variables: Biodiversity data in ecology



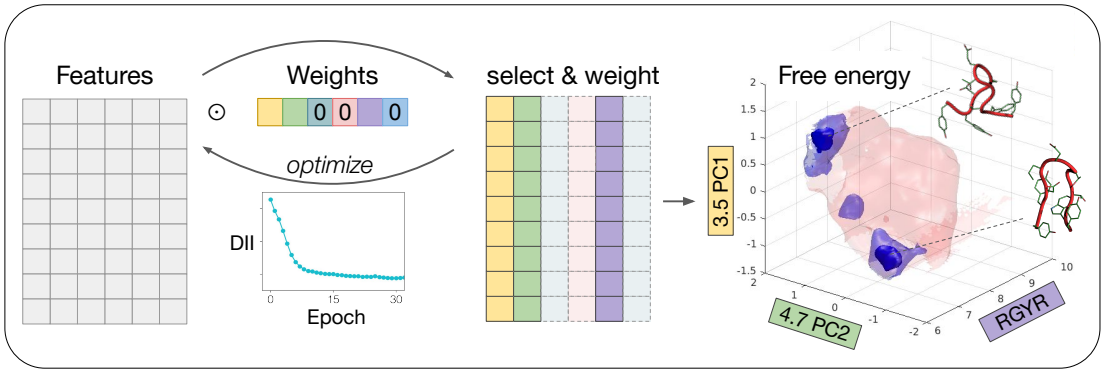
Subsequently, chapter 5 addresses the challenges of analyzing Information Imbalance in categorical non-ordinal data, where the traditional method is not suitable to capture relationships effectively. We propose two solutions, to use continuous features to predict categorical features and vice versa. Both only use the distance information from the continuous space, while considering instead only the classes in the categorical feature.

The approach is tested for investigating species diversity and richness in the Amazon Rainforest using 27 features suspected to related to Amazon biodiversity. Assessing biodiversity is a challenging task, particularly at biogeographic scales.

The analysis revealed asymmetric information flows, where climatic variables moderately predict species richness, while species richness does not provide significant information about environmental conditions. Notably, there is nearly perfect information symmetry between Fisher's alpha diversity index and species richness in 500 trees, suggesting their interchangeable use in ecological studies. Additionally, collecting intensity shows predictive value for species richness, indicating ongoing sampling efforts are crucial for capturing the richness of Amazonian biodiversity. The findings underscore the need for improved sampling strategies and further research on the interplay between community composition and ecosystem functions for effective conservation.

An open question remains the treatment of categorical but ordinal data. In the future the analysis should also be extended to other ecological data sets and the multi-variate case, to test whether there are larger feature sets which improve predictability.

An optimizable Information Imbalance for high dimensional data



In the final technical chapter, chapter 6, we develop an optimizable version of the Information Imbalance, the Differentiable Information Imbalance (DII), where feature weights are determined via gradient descent. This statistic is an advancement in feature selection and weighting methodologies, solving the issues of combinatorial explosion during enumeration of feature tuples, as well as finding an effective weighting between the features. By employing the Differentiable Information Imbalance as a loss function, the relative feature weights of the inputs are optimized, simultaneously performing unit alignment and relative importance scaling, while preserving interpretability. Furthermore, this method can generate sparse solutions: This is particularly advantageous in high-dimensional spaces where traditional methods often struggle. We will show that in the examples we considered, the DII can determine the minimal required set size of informative features while preserving the essential structure of the data.

In practical applications, DII demonstrated robustness and consistency, particularly in analyzing molecular dynamics simulations of the peptide CLN025. The method successfully extracted a small subset of collective variables (RGYR, PC1 and PC2, with weights of 1.0, 3.5 and 4.7) that accurately identified distinct states of the peptide, including the β-pin and collapsed denatured states. This analysis yielded an impressive overall cluster purity of 89% when comparing reduced variable spaces to those constructed from a much larger feature space.

Furthermore, DII was applied to training Behler-Parrinello machine learning

potentials, selecting highly informative subsets of input features from a set of 176 descriptors. The optimized subsets enabled the machine learning potential to maintain nearly the same predictive accuracy while significantly reducing computational costs, achieving a runtime reduction of one-third. This demonstrates DII's effectiveness in improving both the efficiency and accuracy of predictive modeling in complex systems.

While the method can parse any data type, it is most suitable for continuous features. A limitation is given by ground truth metrics with many nominal or binary features, which can lead to a degenerate ground truth rank matrix. Merging the finding described in Chapter 6 with those described in Chapters 4 and 5 remains an open challenge. Overall, the findings underscore DII's potential to help feature selection in practical problems by offering a robust, efficient, and versatile framework. Its accessibility through the python library DADApy enables future explorations in distance-based methods and metric learning.

Key takeaways

The thesis here shows how versatile of a feature selection method the Information Imbalance is: It can be tailored to fit many use cases and provides flexibility in handling diverse data types and structures. Especially the optimizable version marks a leap forward in feature selection, by addressing some open problems in the field. Its implementations in a software package DADApy allow further development by the community.

Contributions

In chapter 4, the clinical analysis was carried out by Dr.med. Emanuela Sozio. In chapter 6, the linear scaling estimator of the DII was developed by Vittorio del Tatto, while the analyses in subsection 6.4.3 were carried out by Felix Wodaczek.

Note

This thesis uses the words "feature" and "variable" interchangeably. The words "target", "output" and "ground truth" space are used synonymously for the target space of the metric, meaning B in $\Delta(A \to B)$.

Chapter 2

Feature selection: The state of the art

2.1 Why select features?

Feature selection is an essential step in many data analysis pipelines. Very often, most of the features defining a data point are redundant, irrelevant, or affected by large noise and have to be discarded or combined. In the related field of *dimensionality reduction*, many powerful methods have been developed to automatically map the data to a low-dimensional representation, without significantly reducing the information content. Prominent examples are principal component analysis [26], autoencoders [27] and kernel-based methods [28, 29]. A critical problem of many dimensionality reduction methods is that the variables obtained are non-interpretable. In autoencoders, the variables at the bottleneck are highly nonlinear functions of the input features. In other approaches, such as Umap [30] and kernel-based methods[28, 29], the variables are not even explicit functions of the features.

However, many use cases do require that the original features are preserved, where the only allowed modification could be neglecting some of these features by a selection procedure or scaling the features relative to each other. The most obvious reason to use feature selection over dimensionality reduction is to preserve interpretability and integrate domain-specific knowledge. In healthcare and finance, the domain experts interpret selected features in order to explain the mechanism of a disease [31, 32, 12], build predictive models [33], or adjust investments strategies [34]. Similarly, in molecular dymanics (MD) simulations, it is beneficial if the selected collective variables (CVs) are interpretable for better mechanistic understanding [20]. In text classification by natural language processing (NLP) feature selection preserves interpretability and improves accuracy [35]. In general, features

can be selected to avoid overfitting and improve predictive performance [36]. In a study on leukemia cancer, for example, it was demonstrated that the disease can be best identified using just 19 out of more than 7000 genes [37]. Finally, in feature spaces where the features are already non-interpretable combinations of some original features of raw data, as in the case of extracted features [38], general dimensionality reduction techniques might add another layer of transformation to the data, which feature selection avoids.

In contrast to general dimensionality reduction, far less methods exist for feature selection.

2.2 Common problems in feature selection

There are uncertainties that are associated to all feature selection applications:

- How do we account for *different units* of measure and / or intrinsic *importance* when selecting feature sets?
- What is the optimal dimension of the reduced feature space, meaning *how many features* do we need to retain a sufficient amount of information?
- How do we quantify this "sufficient amount" of information?
- Which is the optimal combination of relevant, non-redundant features?

The first mentioned difficulty is related with the heterogeneity of the variables: In many cases a data point is defined by features with different nature and units of measures, sometimes referred to as multi-view features [39]. Associating these different features to perform analysis is termed feature fusion [39]. For example, in atomistic simulations one can describe a molecule in water solution providing the value of all the distances between the atoms of the molecule, which are measured in nanometers, together with the number of hydrogen bonds they form with the solvent, which are dimensionless. In a clinical context, the features associated with a patient my include blood exams, gene expression data, and many others [1]. If one wants to mix heterogeneous variables in a low-dimensional description, one should choose a weight factor to match their units of measures. Even if the features have the same unit of measure, some features can carry more information than others and should hence receive a higher weight.

Another difficulty for feature selection, the choice of the number of variables which are actually necessary to describe the system, has a lower bound in the intrinsic dimension [40]. The intrinsic dimension is, informally, the dimension of the manifold which contains the data. However, this number is often scale-dependent [41] and position-dependent [42]. Moreover, if one wants to visualize the data in

one graph, the number of variables is necessarily limited to two or three. This typically implies neglecting part of the information, and poses the problem of choosing which variables should be retained for visualization. The selected and appropriately weighted set of features should contain enough information to effectively address the task at hand. When used with a downstream model, the performance of different models can be compared to quantify the information content of the feature set (see "wrappers" below) [43]. However, this approach leads to a combinatorial explosion when multiple feature sets, each with potentially different relative weights, need to be compared. A more straightforward method for quantifying information content is therefore desirable.

2.3 Types of feature selection algorithms

Feature selection can be characterized according to several criteria, most noteworthy by the presence or absence of a ground truth (supervised and unsupervised), the type of algorithm (filter, wrapper or embedded methods), and considering the accepted data types. An overview is presented in Fig. 2.1.

2.3.1 Supervised and unsupervised feature selection

Feature selection methods can be supervised and unsupervised [44]. In supervised feature selection, labeled data is used to identify features that have the strongest relationship with the target variable [43]. Common methods are *e.g.* mutual information maximization [45], decision trees [46] or recursive feature elimination [47]. In unsupervised feature selection, there is no label or target variable [48]. The selection is based on intrinsic properties of the data, such as variance or redundancy. Examples include Principal Component Analysis (PCA) and clustering-based approaches [48]. There are also semi-supervised feature selection algorithms that combine both [44].

2.3.2 Filters, wrappers and embedded methods

Considering the nature of the algorithm, feature selection methods can be classified into filter, wrapper and embedded methods [43]: Filter methods are independent of downstream task and the features are ranked according to a separate criterion [49]. Wrapper methods, on the other hand, use the downstream task, such as a prediction accuracy, as feature selection criterion, but hence suffer from combinatorial explosion problems because of the need to test all possible feature sets combinatorically or heuristically [43]. If the downstream task is akin to a classification problem, then embedded methods can perform well, because they incorporate the

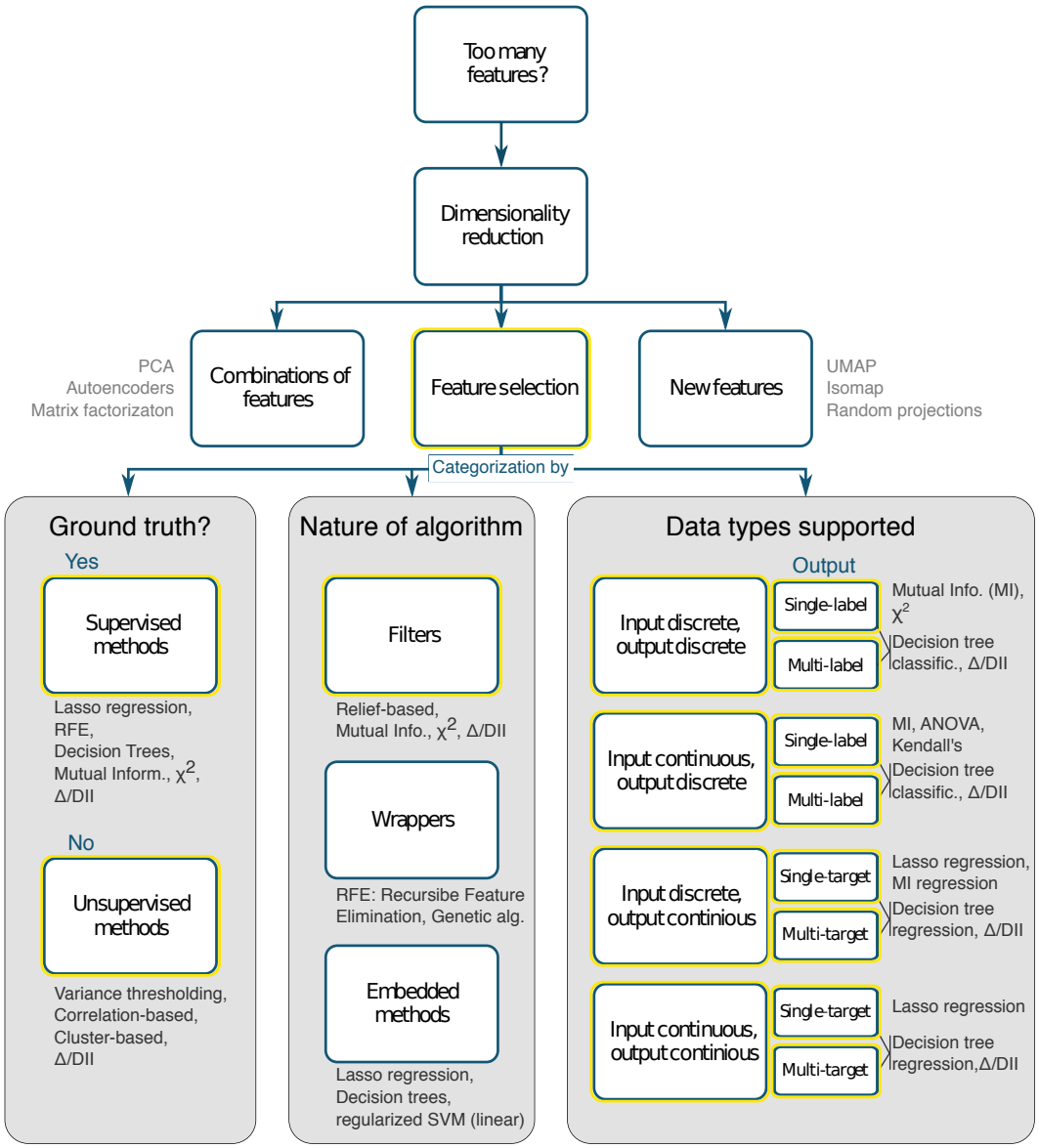


Figure 2.1: The chart shows an overview over the field of dimensionality reduction with a focus on feature selection. Exemplary methods for each category are shown in gray. The yellow highlighted fields denote the categories that are covered by the Information Imbalance Δ or the Differentiable Information Imbalance DII.

feature subset selection into the training [43]. These algorithms are often based on regression, like FSOR [50] and additive models [51], or on support vector machines, like KP-SVM and its variants [52, 53]. Filter methods, on the other hand,

become the logical methods of choice if the downstream tasks are not simple models. While wrapper and embedded methods of feature selection are supervised by definition because they utilize downstream models which building on relationships with the target data, filters include both, supervised and unsupervised methods. Unsupervised filter methods do not utilize target data [54]. They include variance [55] and Laplacian [56] scores and methods which can find feature subsets, preserving clusters of the original data manifold, like multi-cluster feature selection (MCFS) [57] and k-means clustering feature selection [58]. Supervised filters, on the other hand, make use of target data: the "label" or "ground truth". Simple, univariate supervised filters such as correlation coefficient scores, estimated mutual information [59], the chi-square test or ANOVA [60] are efficient, but ignore feature relationships and therefore have problems finding optimal sets [36]. Specific feature subset evaluation filters like FOCUS rely on enumerating all possibly subsets, similar to wrapper methods [61, 62] and have the same combinatorial problems. The relief algorithm and its variants [63, 62] are more efficient because they do not explicitly evaluate the feature subsets. Instead, they utilizes nearest neighbor information to weight features relative to each other (non-myoptic), but feature subsets can still include redundancy [62]. Generally, filter-based feature selection methods have been shown to improve accuracy and precision in many down stream machine learning classficiation algorithms such as SVM and Bayesion Networks [64], while tree-based classifiers tend to work better with more features [64, 65]. A review on feature selection filter methods can be found in [49]. Overall, the field of feature selection is clearly lacking the powerful, automatic tools available to dimensionality reduction, as most of the method we mention have important limitations.

2.3.3 Supported data types

Feature selection algorithms can be categorized based on the types of data they support [66]. Data can be classified as either static or streaming, and the input and output (target) spaces may consist of various continuous or discrete values. While the output space is often one-dimensional (labels), it could also encompass multiple dimensions or even high-dimensional spaces. Additionally, features may be heterogeneous, originating from diverse sources and incorporating different units of measurement [39]. However, very few feature selection algorithms are designed to accommodate multiple data types, as illustrated in Fig. 2.1 ("Data types supported"). On top of this, many methods available in software packages cannot handle missing data.

The primary topic of this thesis is the Information Imbalance method, a filter

approach that can operate in both, supervised and unsupervised manners, and is compatible with most data types, as highlighted in yellow in Fig. 2.1. It is a similarity-based method, seeking to preserve data similarity [66], and also an information-theoretical-based method, maximizing relevance and minimizing redundancy between features, even though it is not directly involving entropy or mutual information [66]. The following chapter introduces this method.

Chapter 3

Information Imbalance: An overview

3.1 Intuition

The Information Imbalance (Δ) is a measure which allows comparing the information content of distances in two feature spaces [13]. Informally, the Information Imbalance quantifies how well pairwise distances in the first space allow predicting pairwise distances in the second space, in terms of a score between 0 (optimal prediction) and 1 (random prediction).

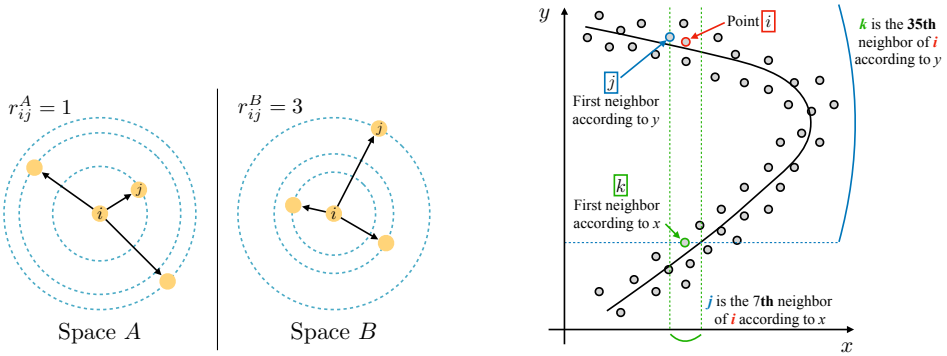


Figure 3.1: **Distance ranks in two feature spaces to measure the relative information contained in these spaces. Reprinted with permission from [13].** a): Illustration of the distance rank of two points in different feature spaces A and B. The rank r_{ij} of point j relative to i is equal to 1 in space A, meaning that j is the first neighbor of i. In space B, j is the third nearest neighbor of point i. b): Illustration of how ranks can be used to verify that space x is less informative than space y. A small distance rank in y automatically implies a small distance rank in x, but not vice versa.

The statistic is based on distance ranks: If data points of the same data set have many features, then we can consider any subset of these features and calculate pairwise distances between the points. For point i, point j, which is the nearest neighbor, receives rank 1, the second nearest neighbor rank 2, *etc.* These ranks will generally be different if we use a different feature set to consider the same points i and j (Fig. 3.1a).

The core idea of the Information Imbalance approach is that distance ranks can be used to identify whether one metric is more informative than the other. In Figure 3.1b, a noisy curved dataset shows that the y-axis is more informative than the x-axis because x can be predicted from y, but not vice versa. This asymmetry is captured in rank differences: point i's nearest neighbor by y-distance is j, but j ranks as the 7th nearest by x-distance. Similarly, i's nearest neighbor by x-distance is k, who ranks 35th by y-distance. In other words, near neighbors in space y are also near neighbors in space x, but near neighbors measured in x might be far in space y. Space y is a good proxy for space x, but space x is not a good proxy for space y

This general property is exploited in the definition of the Information Imbalance which we provide below: Nearest neighbors are better preserved when passing from a more informative to a less informative space than when doing the opposite, as clear from Figure 3.1b.

3.2 Definition of Information Imbalance

Given a data set where each point i can be expressed in terms of two feature vectors, $\mathbf{X}_i^A \in \mathbb{R}^{D_A}$ and $\mathbf{X}_i^B \in \mathbb{R}^{D_B}$ $(i = 1, \ldots, N)$, the Information Imbalance $\Delta(d^A \to d^B)$ provides a measure of the prediction power which a distance built with features A carries about a distance built with features B. The Information Imbalance is defined using copula variables, and estimated as the average distance rank according to d^B, restricted to the nearest neighbors according to d^A [13]:

$$\Delta\left(d^A \to d^B\right) = \Delta(A \to B) \approx \frac{2}{N}\langle r^B | r^A = 1\rangle, \tag{3.1}$$

which is for all practical purposes estimated as:

$$\Delta(A \to B) \approx \frac{2}{N^2} \sum_{i,j:\, r_{ij}^A = 1} r_{ij}^B \tag{3.2}$$

Here, we consider $\Delta\left(d^A \to d^B\right)$ and $\Delta(A \to B)$ (eq. 3.1) as synonymous. $\langle \cdot \rangle$ denotes the expectation value, in this context, the arithmetic mean over the data. N is the number of data points. r_{ij}^A (resp. r_{ij}^B) is the distance rank of

data point j with respect to data point i according to the distance metric d^A (resp. d^B). For example, $r_{ij}^A = 7$ if j is the 7th neighbor of i according to d^A. Information Imbalance hence estimates the conditional rank distribution $p(r^B \mid r^A = 1)$. $\Delta\left(d^A \to d^B\right)$ will be close to 0 if d^A is a good predictor of d^B, since the nearest neighbors according to d^A will be among the nearest neighbors according to d^B. If d^A provides no information about d^B, the ranks r_{ij}^B in Eq. (3.2) will be uniformly distributed between 1 and $N-1$, and $\Delta\left(d^A \to d^B\right)$ will be close to 1.

Several distance metrics could be used to calculate these distance ranks. Since the method is focused on the identification of a feature space which reproduces the nearest neighbors of another feature space, it is not very sensitive to the precise choice of the distance metric. While the distance between two 'far' points will likely be very different if computed, *e.g.*, with the Hamming distance or with the Euclidean metric, the nearest neighbors are more preserved across metrics. In this thesis we will typically use the Euclidean distance, unless otherwise specified.

Unlike the Pearson correlation coefficient, the Information Imbalance is not a symmetric measure and can be calculated from space A to space B and vice versa. Both statistics will result in a number between zero and 1, and together they show the informative relationship between the two spaces.

To illustrate this, consider a dataset from a 3D Gaussian distribution where the z-axis has a much smaller standard deviation than the x and y axes (Fig. 3.2). We can compute distances using all dimensions, $d_{xyz}^2 = (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2$, or just subsets like d_{xy} or d_{yz}.

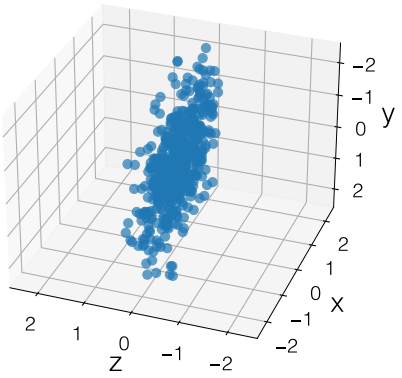
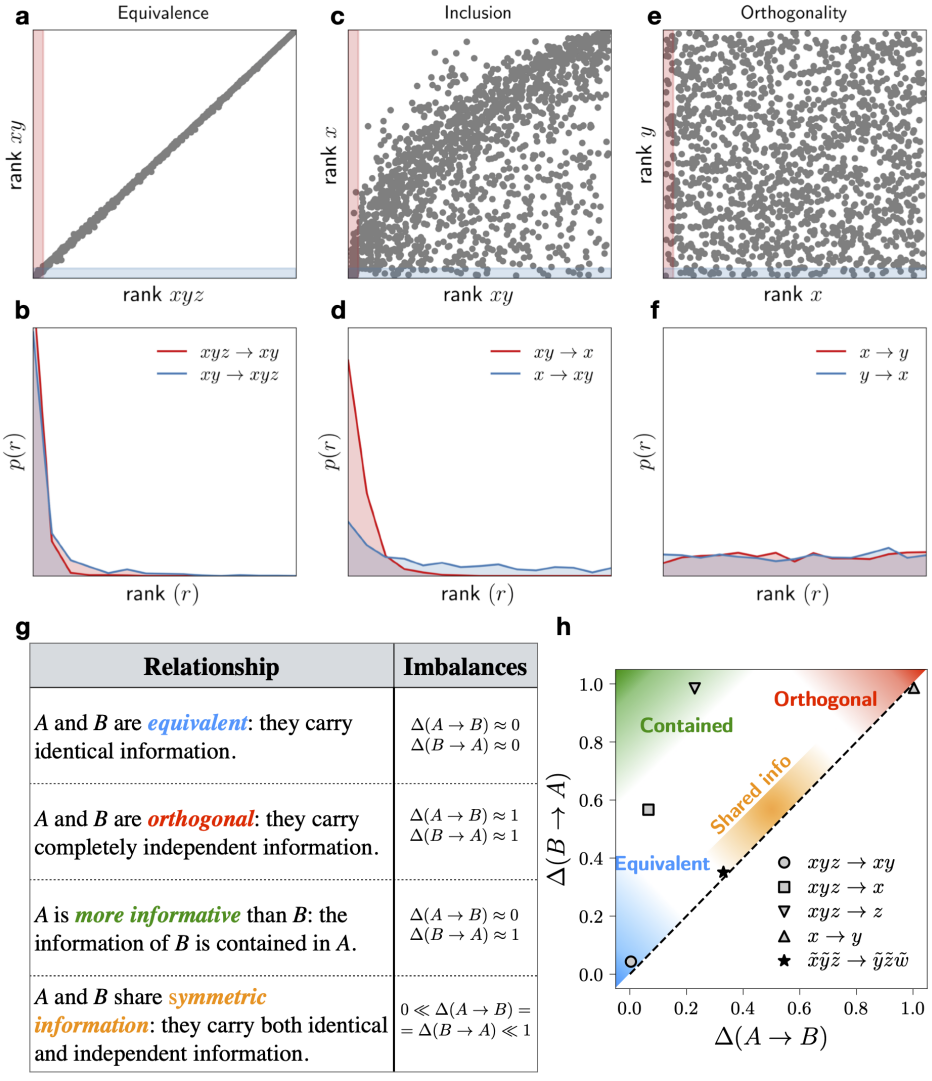


Figure 3.2: 3D Gaussian distribution with small standard deviation along z.

Now one can compare the rank distributions between spaces: In Figure 3.3, the top row compares ranks based on two distances. The second row shows the probability distribution $p(r^A \mid r^B = 1)$, which represents the ranks in space A restricted to nearest neighbors according to distance B and *v.v.*



Relationship	Imbalances
A and B are ***equivalent***: they carry identical information.	$\Delta(A \to B) \approx 0$ $\Delta(B \to A) \approx 0$
A and B are ***orthogonal***: they carry completely independent information.	$\Delta(A \to B) \approx 1$ $\Delta(B \to A) \approx 1$
A is ***more informative*** than B: the information of B is contained in A.	$\Delta(A \to B) \approx 0$ $\Delta(B \to A) \approx 1$
A and B share ***symmetric information***: they carry both identical and independent information.	$0 \ll \Delta(A \to B) =$ $= \Delta(B \to A) \ll 1$

Figure 3.3: **Different relationships between two distance measures using the Information Imbalance. Reprinted with permission from [13].** a), c), e): Scatter plots of the ranks between ordered pairs of points for different feature spaces from a 3D Gaussian dataset with a small variance along z. The highlighted regions indicate the points considered for generating below plots. b), d), f): Probability of ranks in a feature space, given two points are nearest neighbors in the other space. g): The four different types of relationships that can characterize the relative information content of two spaces A and B. h): Information Imbalance plane for the 3D Gaussian dataset discussed. The different colors mark the regions corresponding to the types from panel g.

Panels a and b show that ranks in d_{xyz} and d_{xy} are nearly identical, with distri-

butions sharply peaked around one. This is due to the small variance along z which leads very similar distance ranks in the two spaces. The closer the conditional rank distribution $p(r^B \mid r^A = 1)$ is to being peaked at one, the more information about space B is captured within space A. In panels c and d, comparing d_{xy} to d_x, the more informative d_{xy} leads to ranks clustered around small values but not *v.v.* For independent features (x and y in panels e and f), rank distributions are uniform, leading to an average rank of $\sim \frac{N}{2}$ between pairs of points, and to an Information Imbalance of ~ 1 both ways (uninformative).

The relationship between spaces A and B can now be categorized into four types by comparing the Information Imbalances $\Delta(A \to B)$ and $\Delta(B \to A)$: equivalence, independence, mutual information sharing, or one space fully encompassing the other.

These types, explained in Fig. 3.3g, are visualized by plotting the two Imbalances against each other in *Information Imbalance planes* (like in Fig. 3.3h).

The Imbalance plane for the previously discussed 3D Gaussian dataset in Fig. 3.3h shows that the small variance in the z-axis makes spaces xyz and xy nearly equivalent. It also correctly identifies x as being part of xyz and classifies x and y as orthogonal.

Additionally, a point marked by a black star represents a dataset from a 4D isotropic Gaussian distribution, demonstrating that spaces $\tilde{x}\tilde{y}\tilde{z}$ and $\tilde{y}\tilde{z}\tilde{q}$ share symmetric information.

Since the Information Imbalance relies only on the local neighborhood of each point, it is particularly well-suited for studying nonlinear data manifolds.

The method can be used in a supervised and unsupervised manner, by employing a separate target space or sub-selecting features within one space, respectively. Both have been applied recently in a model of a glass-forming liquid, remarkably with similar selected features [67].

The algorithm for Information Imbalance analyses between feature spaces is publicly available as the `MetricComparisons` class in the Python package DADApy [3] and a comprehensive description can be found in the according documentation [68], which includes a dedicated tutorial.

The following chapter is the first technical chapter of this thesis. A weighted Information Imbalance approach is introduced for feature selection in a medical data set.

Chapter 4

Unbalanced class prediction with categorical variables and missing data in medicine

4.1 Introduction to clinical predictions

In many fields, and in particular in statistical medicine, one attempts to develop a predictor using relatively few data points (the patients), characterized by a number of features which can be large. These features encompass demographics, vital parameters, comorbidities, medications, blood test values, radiological exams, clinical scores and more. Furthermore, they can be of any data type, *e.g.* quantitative (weight, age, blood value levels), binary (presence of diabetes or other comorbidities), nominal (types of ventilation), or ordinal (sequential organ failure assessment (SOFA) score). Many of these features are typically irrelevant or redundant, namely correlated with each other, and a selection of few, relevant features is desirable. For medical professionals, having to consider too many features confuses the clinical work.

Typically, a feature is considered relevant if it correlates with the target, for example if it discriminates between target classes. This simple concept is at the basis of most feature selection algorithms. Briefly recalling the overview provided in chapter 2, feature selection methods can be broadly divided into filter, wrapper and embedded methods. Filters are simple statistics to rank the features independently of the subsequent prediction (classifier agnostic), while wrapper and embedded methods use the predictor as criterion to select feature subsets [43, 69]. Among the classic embedded feature selection methods, lasso-regularized regression [70] provides interpretable feature selection while drawing a linear relationship between input features and target. Sparse additive models (SPAMs) [71] and sparse neural

additive models (SNAMs) [72] extend this to the non-linear case. Variable ranking filter methods tend to be easier and faster to use than other feature selection methods, but they have the drawback of being univariate methods with inability to find the optimal dimension of the feature space, namely the number of variables that are necessary to make a good prediction [43, 36]. Furthermore, many existing feature selection algorithms suffer from the inability to handle missing and noisy data [69].

In this chapter we show that the Information Imbalance [13] can be used as a filter to perform feature selection in a clinical framework. This offers the chance to address two of the challenges mentioned in chapter 2: performing feature selection with missing data, and dealing with mixtures of real, categorical and binary features. The second challenge, as we will see, can be addressed only partially using the Information Imbalance in its standard formulation of ref. [13]. The chapter hereafter, chapter 5, will be dedicated to a more principled manner of addressing it.

We illustrate the procedure on a database of $\sim$ 1300 COVID-19 patients from Udine, including hundreds of features for each patient. These features are extremely heterogeneous, some related with the clinical history, others with the status of the patients at the admission to the hospital, other with the course of the disease, including complications, treatments and clinical outcome. Very importantly, the database is highly incomplete, as is common in clinical databases: the outcome of specific exams (say a TC scan) is typically available only for subsets of the patients, and the clinical history before the admission is often known only partially.

The Information Imbalance approach allows comparing two feature spaces, and deciding if one is more informative than the other. Feature spaces are collections of features that are used to characterize the data. For example, let's say that space A includes age, a specific comorbidity, and the value of a blood test, while space B includes the parameters measured in a TC scan and (also) age. To estimate the Information Imbalance, one finds for each patient their nearest neighbor, which is the other patient that is most similar (closest) according to a distance estimated using the features in space A. In this study we use the Euclidean distance. Say that for patient number 1, the most similar is patient 412, such that patient 412 has distance rank 0 with respect to patient 1 in space A. Next, one computes the Euclidean distance between patient 1 and 412 in space B, *i.e.* using the features of space B, and finds the number of patients which are closer to patient 1 than patient 412. One repeats this test for all the patients and computes the average of this number. The Information Imbalance, denoted in the following $\Delta(A \to B)$, is proportional to this average. If $\Delta(A \to B)$ is small, space A is *predictive* of space B, as patients which are close in A are also close in B, and therefore the average

will be taken over small distance ranks. If instead this number is large, the nearest neighbor patients in A are typically "far" in B, which implies that space A is *not informative* of space B.

In this chapter we show that Information Imbalance can be adapted to work as a filter to perform feature selection in clinical databases. An important advantage over other filter methods is that the approach described in this chapter automatically selects features which are not only relevant, but also uncorrelated. Indeed, the Information Imbalance can directly be computed for distances including arbitrarily many features. This is a practical advantage with respect to other methods which are based on comparison between two variables at a time. It also allows comparing the predictive power of subsets of features of different sizes. This, as we will see, allows finding maximally informative subsets of features along with the optimal dimension.

4.2 Methods

4.2.1 The clincal data set

This chapter includes a retrospective clinical study involving data of 1308 COVID-19 patients from Udine hospital. The data set includes patients admitted to the Infectious Disease ward of the Azienda Sanitaria Universitaria Friuli Centrale Santa Maria della Misericordia of Udine, a 1000-bed tertiary-care teaching hospital identified as a regional referral center for COVID-19 patients, from March 2020 to March 2021. Informed consent was obtained from all participants.

For all patients the following parameters were collected: evaluation of in/exclusion criteria; socio-demographics (age, gender, race, height, weight); date and time of the onset of symptoms and of the admission to the hospital; ward of hospitalization; co-morbidities (dyslipidemia, obesity, diabetes, chronic obstructive pulmonary disease, chronic kidney injures, liver disease, hypertension, solid and hematologic neoplasms, autoimmune diseases, primary or secondary immunosuppression), including Charlson score index; findings from routine physical examination (temperature, heart rate, breathing rate, blood pressure, SPO2, neurological status); routine diagnostics performed (chest X-ray, CT scan, ultrasound, microbiological tests and blood tests performed); date and time of blood sampling initial and final diagnosis; type and focus of infection; date of discharge; date and time of ICU admission and discharge; needs for organ support and/or invasive ventilation; any serious adverse event or complication which occurred during hospitalization; therapies carried out; lab parameters from routine blood testing which were assessed at presentation (within 48 hours of admission); data from blood gas analysis, such as PaO2/FiO2 ratio, alveolar arterial gradient, and lactate.

The features were divided into input (measurable upon hospitalization) and output (severity of outcome of COVID infection) features. 14 output features were decided upon by medical knowledge. The natural hierarchy in the severity of these features led to the creation of a "severity tree" (Fig. 4.1).

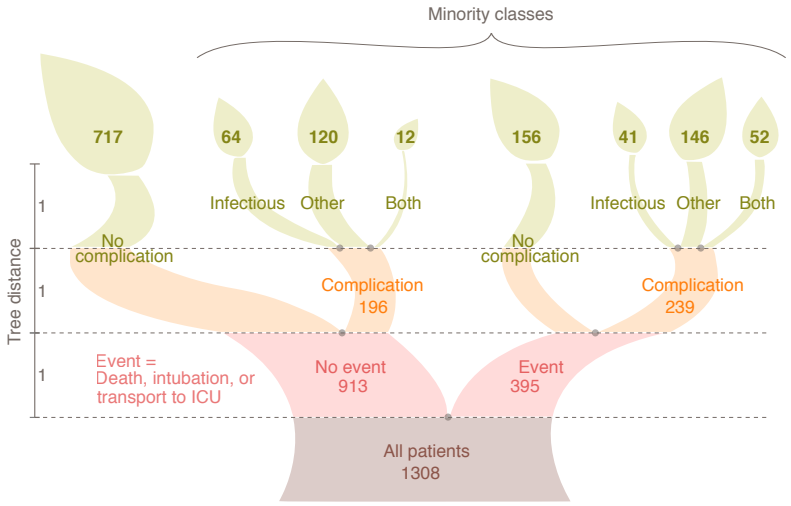


Figure 4.1: The severity tree splits patients into eight severity classes of different class size. These classes are depicted here as the leaves. The gray numbers on the left indicate tree distances between nodes. Distances between patients on the tree are measured one-way, such that the distance between patient A in the first big leaf (no event, no complication) to patient B in the third class (no event, other complication, 120 patients) is 2. This number is found by starting from patient A and reaching patient B: We start in leaf 1, take two steps in tree distance down to the level between orange and red, and then take two steps up to patient B's class. Counting this trajectory only one directional, the distance between the patients is 2.

In the severity tree, death, intubation and transfer to ICU were used to split patients into an "event" group, where at least one of these three events occurred, and a "no event" group. The second split is given by the presence or absence of complications. The difference between infectious and non-infectious complications makes up the third split of the tree, as seen in Fig. 4.1, leading to a classification of patients into eight severity classes.

These output features were available for all patients, yielding a distance - and an identical distance rank - between 0 and 3 for each pair of patients on the tree. The distance between two patients is then estimated by counting the number of links separating their leaves in the severity tree, divided by two (one directional). Since there are only eight classes in the severity tree, the distances between all

patients in this output space are degenerate, meaning that many patients have the same distances from each other. Furthermore, there is a high class imbalance, *i.e.* the biggest class contains more than half of the patients.

138 input variables were selected, including age, gender, physical exams (*e.g.* blood pressure, temperature), blood test (*e.g.* diagnostic antibodies and interleukins), hemogas (*e.g.* partial pressures of oxygen and CO_2, pH of the blood) values and chronic comorbidities (*e.g.* diabetes) and medications (*e.g.* diuretics, steroids). Potential input spaces are tuples of these input variables. As many real world data, the input data is characterized by missing values, which leads to reduced patient numbers for certain combinations of input features. In some possible input spaces also distances between data points are degenerate, due to categorical, binary and repeated values.

4.2.2 Class-corrected Weighted Information Imbalance

As described in the dedicated section section 3.2 in eq. 3.2, the Information Imbalance Δ is defined as follows [13]:

$$\Delta(A \to B) = \frac{2}{N} \frac{\sum_{i=1}^{N} r_i^{B'}}{N}, \tag{4.1}$$

where $r_i^{B'} = r_i^B$ given $r_i^A = 1$. N is the number of all data points, in this case all patients. Here, a data point is a patient. The Information Imbalance between feature space A and feature space B, $\Delta(A \to B)$, is proportional to the average of the neighbor ranks in space B, conditioned to nearest neighbors in space A, and normalized such that if A predicts space B perfectly, $\Delta(A \to B) \approx 0$, and if A has no information on B, $\Delta(A \to B) \approx 1$. Note that due to the tree structure in the COVID-19 output space in this paper we assign the nearest neighbor rank 0, which leads to very small numeric differences in the case of few data points. This chapter uses the Euclidean distance as distance metric in the input feature spaces.

For the high dimensional classification of COVID-19 severity, Information Imbalance is used in the feature selection step. The task is to find input variable spaces in which the nearest neighbors optimally describe the distribution of the output classes. The severity tree output space is degenerate with high class imbalance, where leaf 1 has more than half of the total patients. Using naive classifiers in class imbalanced data sets biases class prediction heavily towards the majority class, especially when feature selection is performed. Subsequently, the class predictive accuracy is low in the minority classes [73]. The original implementation of Information Imbalance was developed for continuous in- and output spaces and has the described shortcomings when applied to few, imbalanced output classes. However, in sick patient prediction there is a high cost associated to false negatives.

To find feature spaces which also predict the neighborhood of the patients in small classes, we introduce class weights $w_{i,l} = \frac{1}{leafsize_l}$, for each patient i in class l. Furthermore, the distances in the severity tree are highly degenerate. The structure of our tree leads to a total of four distinct distances and identical four distance ranks, where rank 0 means that two patients are in the same class and rank 3 means the two patients are on opposite sides of the tree (event *vs.* no event). Since the classes are imbalanced, the average probability to find a certain rank neighbor from the different classes is not uniform. Therefore, the normalization a is built to reflect this and bring the average value of Δ to a value of 1, when the nearest neighbor ranks are distributed randomly. The adjusted "Weighted Information Imbalance Δ_w" becomes:

$$\boxed{\Delta_w(A \to B) \approx a \frac{\sum_{i=1}^{N} r_i^{B'} w_i}{\sum_{i=1}^{N} w_i}} \tag{4.2}$$

If more than one nearest neighbor exists at the same distance in the output space B, the nearest neighbor rank is the mean over these M nearest neighbors of patient i: $r_i^{B'} = \frac{\sum_{j=1}^{M} r_{i,j}^{B'}}{M}$.

For this implementation the input space needs to provide clear nearest neighbor assignments, in order to find the according ranks in the output tree. To resolve the degeneracy in the input space, small random numbers are added to duplicated input values. Since this makes the estimated Imbalance a stochastic variable, we repeated the optimization ten times with different random seeds, verifying that the results are robust. For the ten implementations, the Weighted Information Imbalances of optimal tuples with the same tuple sizes are mostly identical, up to the second digit (equal to figure 4.3a) with standard deviations on the order of 10^{-4}. Also the chosen tuples for each size are largely congruent, with the best single variable always being brain natriuretic peptide (BNP) and the best 13-plets of the ten implementations being identical to the one present in this chapter in subsection 4.3.2, except in one case, where eosinophils (FLEOS) have been selected instead of lymphocytes (FLLINF).

Variables were normalized by dividing them by their standard deviation in order to move them into a comparable value range.

The problem of missing data was treated in a constrained, data-set-reductive manner, by which only feature tuples were considered which were present in at least 100 patients. Then Δ_w was calculated for the feature tuple in question using only these ≥ 100 data points with the additional constraint that the base-2 Jensen-Shannon divergence of the tuple class distribution towards the full class distribution (of the 1308 patients) was ≤ 0.06 , in order to ensure proportionate stratified sampling, *i.e.* such that the share of the classes in the sub-sample is proportionate to the full sample. This subset of ≥ 100 patients has no missing

values and can hence be passed on to the downstream classifiers, kNN and SVC, without problems.

4.2.3 Beam search

Input spaces with more than 1 variable were selected by applying Δ_w to a pool of candidate feature tuples selected by beam search with beam width 55. Beam search is a heuristic algorithm [74] which is employed because the full exhaustive search of all possible variable k-tuples (out of D features) would lead to combinatorial explosion with complexity $\mathcal{O}(D^k)$. Like in the vanilla greedy approach, the pool of candidate feature tuples is sequentially extended by adding new features to the best scoring previous tuples. However, in beam search, not only the one best result is chosen and variables added to it, but the n best results, where n denotes the beam width. In this case, the 55 best results were iteratively extended. The computational complexity of beam search is $\mathcal{O}(D \cdot n \cdot k)$ [75] for beam width n.

4.2.4 Prior-corrected k-NN prediction of severity

After finding Δ-optimized sets of features, we use an adjusted k-nearest neighbor (k-NN) prediction in a leave-one-out (LOO) approach. Comparing the predicted severity class to their actual class, we evaluate the performance of the method with cumulative distribution functions (CDFs) of the distances d, *i.e.* consider the fractions of cases in which the class was predicted correctly ($d = 0$), or a neighboring class was predicted ($d = 1$), or the same side of the severity tree was predicted ($d = 2$). The empirical probability distribution of classes for each patient calculated from their nearest neighbors' classes cannot be taken at face value due to the class imbalance. Effectively, the class imbalance makes it much more likely to find a majority class NN than to a minority class NN. The average global density of minority class points is smaller. For this reason we divide the empirical probabilities by the prior probabilities P^0 of the classes in the samples, to create a metric which measures how much bigger or smaller the local density of the various classes is around the patient, in comparison to the average global density. If this value is greater than 1 for a class, this class is more likely than average to be the class of the patient. Since several classes can have values greater than one, the prediction is based on the class with the maximum value.

$$P^0_{i,l} = \frac{leafsize_{i,l}}{\sum_{k=1}^{Nl} leafsize_{i,k}} \tag{4.3}$$

$$\rho_{i,l} = \frac{P^{emp}_{i,l}}{P^0_{i,l}} \tag{4.4}$$

4.2.5 Identifying important but rarely available features by "usage when available"

Due to the missing values in the input data set, the globally selected best variables are a function of their intrinsic goodness of prediction as well as their availability, especially together with other orthogonal features. To decouple these effects to find intrinsically valuable features, the optimal tuple for each patient is found using Δ_w in a leave-one-out (LOO) procedure. The "usage when available" statistic U_f for each feature f is simply defined as the ratio of the number of times a feature is used in all patient-specific Δ_w-optimized tuples n_{f,Δ_w}, over the count of the availability of that feature across all patients a_f:

$$U_f = \frac{n_{f,\Delta_w}}{a_f} \tag{4.5}$$

4.2.6 Mutual information and sequential feature selection

Two other standard feature selection methods are compared to Information Imbalance. A frequently used filter is the estimated mutual information (MI) between each feature and the output classification [59], and a straight forward wrapper method is sequential feature selection (SFS). For both the implementation in scikit-learn [76] is employed and for both methods the data set has to be complete. Thus missing values were filled in by imputation (scikit-learn KNNImputer with 10 NN, uniform weights and Euclidean distance). The MI between each feature and the output classes were calculated with the scikit-learn class mutual_info_classif (3-NN). Forward SFS was calculated using the wrapper SequentialFeatureSelector (5-fold, 10-NN) and SVC (balanced class weights) respectively). The prior-corrected k-NN predictor was used for classification in a Leave-One-Out (LOO) cross validation approach, and compared with a support vector classification (SVC) LOO prediction as implemented in scikit-learn (sklearn.svm.SVC), using default settings and balanced class weights.

4.3 Results

This study is based on a dataset of 1308 COVID-19 patients with $\sim$ 150 features, and 33% of missing values. Some of these features are *input* features, measurable upon admission in the hospital. These include age, gender, physical exams (*e.g.* blood pressure, temperature), blood tests (*e.g.* biomarkers like interleukins), arterial blood gas (*e.g.* partial pressures of oxygen and CO_2, pH of the blood), chronic comorbidities (*e.g.* diabetes) and chronic medications (*e.g.* diuretics, steroids). 36% of all input features were missing, with only 28 features being complete, and

25 features available in a quarter of the patients or less. The output features were 14, all binary. These features could only be measured later on during the COVID-19 infection, and are therefore the variables that a clinician might be willing to predict. These are available for all patients and include death, intubation, transfer to ICU, and 11 complications (heart attack, pulmonary embolism, arrythmia, atrial fibrillation, stroke, thrombosis, pneumothorax, pneumomediastinum, hemorrhage, delirium, and secondary infections during hospitalization).

4.3.1 Correlation and Information Imbalance between numerical patient features

We first use the classic Information Imbalance to investigate the relationships between the input features. We consider the 90 numerical input features for which it is possible to estimate the standard Information Imbalance introduced in ref. [13]. We computed the Information Imbalance Δ between each pair of features using the implementation in the Python package DADApy [3]. $\Delta(A \to B)$ is close to zero if feature A predicts feature B well. It is close to one if feature A does not provide information on feature B. For each pair of features we also computed the standard Pearson and Spearman correlation coefficients r and ρ, which are ± 1 in the case of a perfect positive or negative correlation, and 0 if there is no correlation. If two features correlate strongly, $\Delta(A \to B)$ and $\Delta(B \to A)$ should both be small and similar numbers, if both predict each other to an equal amount. However, if one feature predicts the other, but not vice versa, there exists an asymmetric correlation, and this is reflected in an asymmetric Information Imbalance. This phenomenon, as we will see, is not captured by Pearson and Spearman correlations.

In the table in Fig. 4.2a, we report the Information Imbalance and the correlation coefficients between the 20 pairs of features with the lowest $\Delta(A \to B)$. To highlight some possible relationships, we plot some of these feature as a function of each other in the bottom panels (Fig. 4.2b).

The Information Imbalance faithfully captures features which have strong correlations with each other. The top-eight positive correlation couples (all $r > 0.8$) are contained in the top-20 Information Imbalances. A clear sanity check is displayed in the first rows: the two different laboratory methods for the glomerular filtration rate (GFR and GFR.1), which hold the same values, display perfect correlations and extremely low Information Imbalances. This is also true for the prothrombin time (and international normalized) ratios (PT/INR and PTR), where one is just a normalized version of the other. Correlation and Information Imbalance pick up on the linear relationship between hematocrit (EHCT) and hemoglobin (EHB). EHCT is the percentage of volume occupied by red blood cells relative to whole blood, and therefore is often related to hemoglobin (EHB). Also,

a

No.	Feature A	Feature B	Pearson r(A,B)	Spearman ρ(A,B)	Δ(A→B)	Δ(B→A)
1	GFR.1	GFR	1.000	1.000	0.037	0.037
2	PT/INR	PTR	1.000	0.998	0.049	0.050
3	EHCT	EHB	0.981	0.976	0.236	0.261
4	EWBC	FLNEU	0.874	0.959	0.276	0.283
5	PaO2/FiO2	PaO2	0.177	0.279	0.351	0.805
6	FiO2	PaO2/FiO2	-0.906	-0.770	0.425	0.441
7	EMCH	EMCV	0.947	0.916	0.448	0.450
8	EHCT	ERBC	0.861	0.853	0.527	0.541
9	CRE	GFR	-0.533	-0.842	0.560	0.592
10	CRE	GFR.1	-0.533	-0.842	0.560	0.579
11	EHB	ERBC	0.820	0.819	0.561	0.592
12	Oxygen saturation	PaO2	0.599	0.807	0.571	0.593
13	BILD	BILT	0.962	0.874	0.573	0.598
14	TROP	FIBCL	-0.496	-0.507	0.574	0.713
15	IGM	HDL	-0.003	0.109	0.612	0.810
16	IGG	HDL	-0.295	-0.055	0.628	0.876
17	A-a gradient	PaCO2	-0.455	-0.695	0.651	0.691
18	BNP	Birth	-0.504	-0.708	0.662	0.792
19	TROP	Anion gap	0.063	-0.382	0.699	0.858
20	HBA1CM	IGA	0.100	0.042	0.711	0.804
...	...	...	...	...	...	...
21	BMI	GFR	-0.077	-0.044	1.029	1.040

b

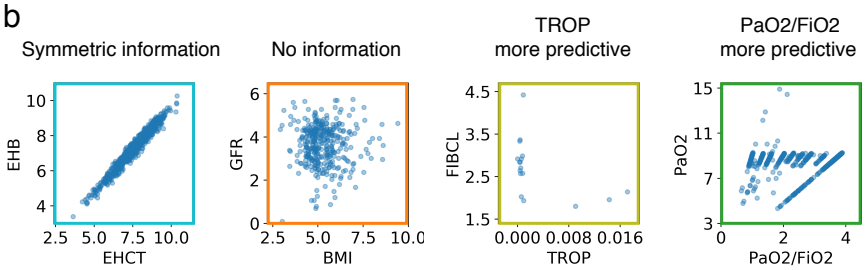


Figure 4.2: **a**: Features ordered according to the lowest Information Imbalances towards another feature, and their Pearson and Spearman correlation coefficients. Yellow colored rows have notably asymmetric Information Imbalances, where $|\Delta(A \to B) - \Delta(B \to A)| > 0.1$. **b**: Scatter plots of several of the feature *vs.* each other from A. The values are the normalized features.

the strongest negative correlation pairing is in the top-20 imbalance table (row 6). Thus the strongest correlations account for nine rows in the top-20 Information Imbalances.

The other eleven rows are made up by pairings which have less strong correlations, but six of them have high asymmetries in their Information Imbalances towards each other (yellow rows in 4.2a), describing a relationship where one variable is more informative about the other than *vice versa*. These six pairings have predominantly very low correlations, showcasing that correlation fails to identify these asymmetric relationships. The effect is especially pronounced in row five,

where PaO2/FiO2 (oxygen partial pressure over fractional inspired oxygen) has a low Information Imbalance towards PaO2 (oxygen partial pressure) and such explains this feature space well, while the same is not true *v.v.* This can be used as a proof of concept because indeed PaO2/FiO2 is the value of PaO2 divided by FiO2 (fractional inspired oxygen) - a simple relationship via one confounding variable which is not detected by correlation (r=0.177). It should be noted that, from a clinical point of view, measuring the PaO2/FiO2 ratio can become very challenging: if patients are not on invasive mechanical ventilation, it is almost impossible to know the exact FiO2, because the devices deliver a variable inspired oxygen concentration. Information Imbalance also detected similar cases where the exact relationship is not known: Here we report asymmetric relationships between troponin (TROP), a well-known marker of cardiac injury, and tissue damage marker fibrinogen (FIBCL), as well as between the immunoglobulins IGM / IGG and the high-density lipoprotein (HDL). TROP values are more predictive of FIBCL values than the other way around. Fibrinogen is a plasma acute-phase reactant protein produced by the liver and is a major coagulation factor. Its concentration increases with inflammation, and it is traditionally considered a risk factor for cardiovascular disease [77, 78], which might explain the connection to troponin. Troponin, on the other hand, is a very specific marker: recent studies showed that troponin dosage should be considered as a prognostic indicator in all patients with moderate/severe COVID-19 at hospital admission and in the case of clinical deterioration [79]. Retrospective data have placed a strong emphasis on the possibility that acute myocardial injury represents a critical component in the development of serious complications in patients hospitalized with COVID-19 [80, 81, 82]. To the best of our knowledge, there is no literature concerning the exact relationships between IGM / IGG and HDL.

We point out here that inter-feature asymmetric relationships could be important. They are not captured by standard correlation analyses, and they could lead to redundancy effects when tuples of features are used for predictive purposes.

4.3.2 Feature selection by optimization of the Information Imbalance

We now consider a classical problem in feature selection, finding a small subset of the 138 input features which is maximally predictive with respect to the output features. In the case of the database analyzed in this chapter, the output features are 14. These features are all binary ("yes" or "no") and are quite heterogeneous in nature. In accordance with the medical insight of the clinicians co-authoring this study, we organized the output features in a "severity tree" (Fig. 4.1 and subsection 4.2.1).

In short, death, intubation and transfer to ICU were used to split patients into two classes, one for which at least one of such events has occurred (the patients whose course has been more severe), the other in which no event has occurred. The other output features are associated to infectious and non-infectious complications, which are important to decide a clinical strategy. This leads to a classification of patients into eight severity classes. The distance between two patients is then estimated by counting the number of links separating their leaves in the severity tree, divided by two (see Figure 4.1).

This tree distance is the target of the feature selection, however the nominal value of the distances is unimportant and only their relative order matters, since the Information Imbalance method uses distance ranks, *i.e.* the closest neighbor in distance is assigned rank 0, the second closest rank 1, *etc.* The feature selection algorithm presented in this chapter works as follows. We try to identify a distance A, built as the Euclidean distance using a combination of several input features, whose distance ranks are maximally informative with respect to distance ranks B measured on the severity tree. Degeneracies in input features were treated by addition of small random numbers (see subsection 4.2.2). A more rigorous procedure aimed at dealing with degeneracies will be introduced in chapter 5. The Information Imbalance between A and B is used as a feature selection filter to discriminate between different choices of A (namely of input features) and select the best one. The classes, "leaves" of the severity tree, are not populated uniformly: class 1 has 717 patients, and the smallest class has only 12 patients. Therefore, the Information Imbalance has been modified by introducing class weights, aimed at compensating the occurrences of the different severity classes in the data set (see subsection 4.2.2). We denote this modified, Weighted Information Imbalance by Δ_w.

To identify the best combination of input features, we find the combination of n input variables minimizing Δ_w , which are present in at least 100 patients (see subsection 4.2.2). For small n, the search can be performed exhaustively by testing all the possible combination of variables. For large n, the number of possible combinations grows factorially. We use the deterministic beam search algorithm (see subsection 4.2.3) which allows finding the best combination of variables for arbitrary n with great confidence. In Fig. 4.3a, we plot the optimal value of Δ_w as a function of n. This value decreases up to $n \simeq 13$, then starts growing slowly, indicating that adding more variables *reduces* the information. This can happen when the new variables only add noise, and no independent information. This analysis indicates that the most informative combination of features includes 13 variables (blue dot in Fig. 4.3a). The globally best n-plets of features as a function of n (until n=13), corresponding to Fig. 4.3 are:

1. BNP

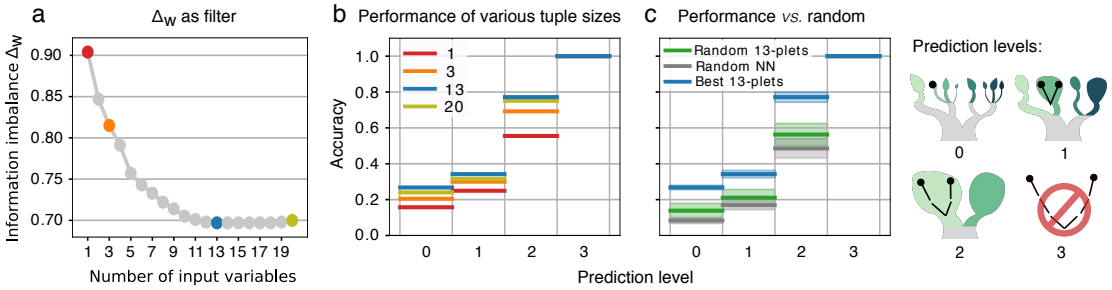


Figure 4.3: **a)** The optimal (lowest) Information Imbalances Δ_w as a function of the number of input features. **b)** Accuracies of 10-NN predictions at given prediction levels for tuple sizes as marked in **a** (the line shows the average over the 10 best Δ_w results for each tuple size). The accuracy corresponds to the fraction of patients predicted correctly at a given prediction level. The prediction level corresponds to the maximum tolerated distance of the true *vs.* the predicted class on the severity tree, as depicted on the right. Together, the lines of one color can be considered the CDF of the fraction of patients predicted correctly. **c)** Accuracies of the top 13-plets, benchmarked *vs.* randomly drawn 13-plets (green), and random nearest neighbors assignments (grey). The average over the top-10 Δ_w results is shown as bold line and the standard deviation as shade. The predictions were generated by LOO. The graphic on the right explains prediction levels: 0 denotes the fraction of patients for whom the correct class was predicted; 1 indicates the fraction for whom the occurrence of event and complication (yes or no) was predicted correctly; 2 means the patients predicted on the correct tree side (event *vs.* no event). The remaining fraction of patients was predicted wrongly (distance = 3).

2. AT3, IP10
3. TC, A-a gradient, IP10
4. FLNEU%, AT3, IP10, ASMA
5. FLNEU%, AT3, IP10, ACARG, ASMA
6. FLLINF%, AT3, IL-10, TROP, ANCA1, ASMA
7. GOT, FLNEU%, AT3, IL-10, TROP, ANCA1, ASMA
8. Steroid therapy, FLNEU%, GPT, AT3, IL-10, TROP, ANCA1, ASMA
9. Hepatopathy, FLLINF%, GPT, AT3, IL-10, TROP, ANA1, ANCA1, ASMA
10. Hepatopathy, FLLINF%, GPT, AT3, IL-10, ACARG, TROP, ANA1, ANCA1, ASMA

11. Hepatopathy, steroid therapy, potassium sparing diuretics, FLLINF%, GOT, AT3, IL-10, TROP, ANA1, ANCA1, ASMA

12. Hepatopathy, steroid therapy, potassium sparing diuretics, FLEOS%, FLNEU%, GPT, AT3, IL-10, TROP, ANA1, ANCA1, ASMA

13. ***Pathologies*: Hepatopathy (liver disease); *Chronic therapies*: Steroid therapy, potassium sparing diuretics; *Blood exams*: Alanin aminotransferase (GPT), antithrombin III (AT3), interleukin 10 (IL-10), troponin (TROP), antinuclear antibodies (ANA), antineutrophil cytoplasmatic antibodies (ANCA), anti smooth muscle antibodies (ASMA), Lymphocytes (FLLINF), percentage of eosinophils (FLEOS%), percentage of neutrophils (FLNEU%).**

In the following, we consider the optimal 13-plet as the feature space optimizing Δ_w. This combination of 13 features suggests on the one hand a systemic inflammation and autoimmunity, signaled by neutrophils and autoantibodies, and on the other hand an immune paralysis and anti-inflammatory effort (*i.e.* steroid therapy, IL-10). Furthermore, it has already been suggested that the up-regulation of inflammatory markers can lead to the progression of the disease to the severe form and eventually cause liver damage in these patients [83], which suggests why hepathopathy might be an important feature.

The information provided by these input features on the severity of the course of the disease is assessed by using these variables to predict the class of each patient. We first use a prior-corrected k-NN classifier, in which the class of a patient is assumed to be the same of their 10 nearest neighbors according to the input features (see subsection 4.2.4 for details). This predictor has no variational parameters, and lacks therefore tunability, but allows assessing directly the consistency between the neighborhood of the patients induced by our optimization procedure. Later in this study, we also use a support vector classifier (SVC) to compare the results to the k-NN classification. In Fig. 4.3b, we plot the accuracies of the prediction at different prediction levels, and for a different number of features ranging from 1 to 20. The accuracy corresponds to the fraction of patients predicted correctly at a given prediction level, defined by the maximum tolerated distance of the true *vs.* the predicted class on the severity tree. Level 0 is the fraction of patients predicted completely correctly, level 1 is the correct prediction of having an event and complication, and level 2 is the fraction for whom only the event was predicted correctly (correct tree side). The remaining percentage belongs to patients misclassified according to having an event or not (level 3). The accuracies are estimated by a leave-one-out (LOO) validation procedure. The performance increases with the number of features, as seen in the height of the CDF curves. This effect staggers with bigger tuple sizes and levels out.

Our approach automatically also leads to a selection of features which are practically uncorrelated. This is demonstrated by calculation of the Pearson correlation coefficient and the pairwise classic Information Imbalance for all the numerical features contained in the Δ_w-optimized nplets (Fig. 4.4). The mean of the pairwise

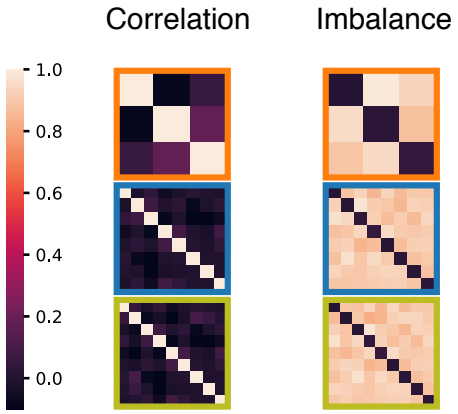


Figure 4.4: The pairwise Pearson correlation heat mat of the numerical features of the Δ_w-optimized n-plets(3, 13, and 20) from Fig. 4.3a, and the pairwise, classical Information Imbalances of the same n-plets.

correlations of numerical features in the best 13-plet is $\overline{r} = 0.02$, and the pairwise Information Imbalances mean is $\overline{\Delta} = 0.96$. Both numbers mean the features are practically uncorrelated with each other and do not hold information about each other. This happens "for free" since adding a feature which can be predicted by other already selected features does not significantly improve Δ_w.

We note that the Weighted Information Imbalance of the top-ranking Δ_w-optimized tuples of the same size is very similar. *E.g.* the best 13-plet has $\Delta_w = 0.69$, while the tenth best 13-plet has $\Delta_w = 0.70$ and differs by only 3 features from the best. Therefore, in Fig. 4.3b and c the plotted lines are averages over the top ten results.

To put these results in context, we need to compare them to a baseline. We first performed a comparison with the predictive performance of randomly selected tuples. Secondly, we performed a comparison with a prediction performed by assigning each patient a "nearest neighbor" at random. To be comparable to our Δ_w-optimized tuples, both comparisons use averages over the ten best performing random 13-plets (Fig. 4.3c with shaded standard deviation). The prediction accuracy is much higher in the Δ_w-optimized 13-plets than when using random variables. With the Δ_w-optimized 13-plets the exact target class predictions are about 27% of cases ($distance = 0$), while predicting the correct side of the severity tree (event *vs.* no event, $distance = 2$) has a quota of around 77%. In comparison, the completely random result distinguishes event *vs.* no event with under 50%

accuracy and the random 13-plets from our data set predict the same with an accuracy of under 60%.

4.3.3 Accuracy of prediction compared to other methods

Accuracy of prediction compared to mutual information and sequential feature selection

We then compared our results to standard feature selection methods, namely to selection by a mutual information (MI) score and to forward sequential feature selection (SFS). MI is a filter method which ranks the individual features against the output classes, while SFS uses the predictor method (here: k-NN and SVC) in a greedy approach to search the best n-plets, and is hence a wrapper method (see subsection 4.2.6). Using the top ranking 13 features selected by MI, the 13-plet selected by SFS (different for k-NN and SVC), the 13-plet selected by our approach, and all features, we perform a prediction with these four sets of features using two different approaches, the k-NN predictor with k=10 and a SVC predictor with balanced class weights.

Unlike Information Imbalance, these two feature selection methods do not provide a way to select the optimal tuple size at the feature selection stage. Therefore, the predictive performances of optimal tuples of several sizes selected with MI and SFS was compared. Using 13-plets of features remained the standard, since in all three feature selection models this size was optimal or nearly optimal. The results are presented in Fig. 4.5a.

Using the k-NN predictor the accuracy of the prediction is significantly higher if one uses our approach. Using SVC, SFS performs better than the other feature selection methods, especially at a prediction level 0 (Fig. 4.5a SVC). However, doing the prediction only for the seven minority classes, *i.e.* excluding the no-event-no-complication class (Fig. 4.5b), the feature tuple obtained with our approach consistently outperforms all other tuples in both, k-NN and SVC predictions. The SFS tuple has especially low accuracy for minority classes in the SVC prediction, even though balancing class weights were employed in the tuple generation and prediction. We recall that in imbalanced multiclass prediction the problem is twofold: prediction in imbalanced datasets tends to favor the majority class, and on top of this, prediction is more complex than in binary prediction, because there can be several majority and minority classes with various relationships towards each other [84, 85]. Furthermore, the error introduced in the imputation could effect the minority classes more, as previously elaborated for standard imputation methods [86].

Their reliance on imputation is an Achilles' heel of both, NMI and SFS, because most standard implementations, such as scikit-learn [76], need data sets without

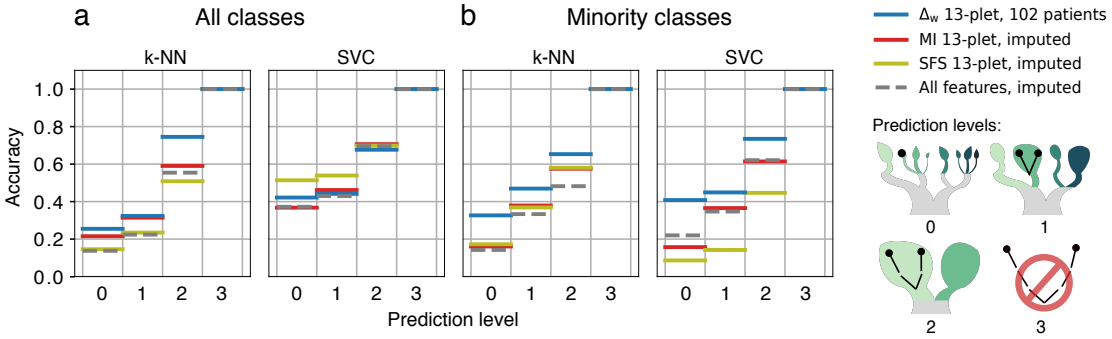


Figure 4.5: Accuracy of prediction using the top Δ_w-optimized 13-plet, *vs.* using the 13 features with the highest mutual information (MI, filter) score on the imputed data set (red) and the 13-plet selected by forward sequential feature selection (SFS, wrapper) on the imputed data set (olive). The dashed grey line shows the prediction using the complete, imputed dataframe without prior feature selection. LOO was used for generating the predictions. Accuracy corresponds to the fraction of patients predicted correctly at a given prediction level. The prediction level corresponds to the maximum tolerated distance of the true *vs.* the predicted class on the severity tree, as depicted on the right. **a)** Accuracies, using k-NN and SVC predictors, respectively. **b)** The same as **a**, but considering the predictions of the seven minority classes only (excluding no-event-no-complication). The SFS 13-plets for k-NN and SVC prediction are two distinct ones, using k-NN and SVC predictors accordingly in the SFS-construction of the 13-plets.

missing values. In our dataset, slightly more minority class data had to be imputed than majority class data (38% *vs.* 35%) and the different feature selection methods selected tuples with 49% (NMI), 19% (SFS with 10-NN) and 32% (SFS with SVC) imputed data. Weighted Information Imbalance, as introduced in this chapter, uses a data-set-reductive approach (see subsection 4.2.2) and hence does not impute features. It is a non-parametric algorithm employing class-balancing weights which can intrinsically handle multiclass ground truths.

Accuracy of prediction compared to regularized classifiers

Information Imbalance is not a classification method. However, in the application described in this chapter the ground truth metric is defined on a severity tree. Its leaves can be considered as categories (classes) which can be used as a target for a classification method. Therefore, a comparison with regularized classifiers, which do not employ explicit prior feature selection, is presented: The prediction accuracies are compared with two regularized classifiers, the sklearn [76] implementations of

1. L_1 (lasso) regularized logistic regression classification
   sklearn.linear_model.LogisticRegression with
   penalty='l1', C=1 or 0.04, max_iter=200, class_weight='balanced',
   solver='liblinear', tol=0.01

2. Regularized sparse SVC
   sklearn.svm.SVC with C=10 or 1, kernel='rbf', gamma='auto',
   class_weight='balanced'.

The results presented in Fig. 4.6 show that, in terms of general prediction accuracy, these regularized classification methods perform at least as good, sometimes better, as the feature-selection-plus-classification models. Sparse SVC even reaches over 50% at the class level. This effect, however, does not translate to the minority classes, where this same classifier only reaches 19% correct predictions, while Information Imbalance selection and subsequent SVC reach 41%. For all prediction levels of minority classes, modestly regularized ($C = 1$) sparse SVC yields the second best results.

All in all, given the importance of minority classes in the clinical setting, we conclude that the Information Imbalance filter method finds a superior feature subspace for severity prediction in the present COVID-19 database than the two other feature selection methods considered here, and also then state-of-the art regularized classifiers, considering the performance in minority classes.

4.3.4 Predictive power for patients without the optimal input tuples

The 13 features which have been identified as optimal as described above are simultaneously available for only 102 patients due to missing values in the data set. Our method, however, can find for each patient their patient-specific best n-plet.

The patient-specific optimal n-plets (Fig. 4.7) were found in a leave-one-out (LOO) approach by considering all features that were present in the respective patient, then beam-searching over these starting from the 1-plets. For each of these feature tuples the Weighted Information Imbalance is calculated using all the patients who have full information in these features, and the search is stopped when the Information Imbalance flattens or starts increasing. In this way, the patient-specific optimal tuple is found, and along with it the optimal dimensionality.

Then we performed a 10-NN prediction of severity for each patient, using their optimal n-plet of features in a LOO cross validation, where we use all other patients who share the same features as training set. As default, the algorithm only considers possible feature tuples which are available in at least 100 patients and

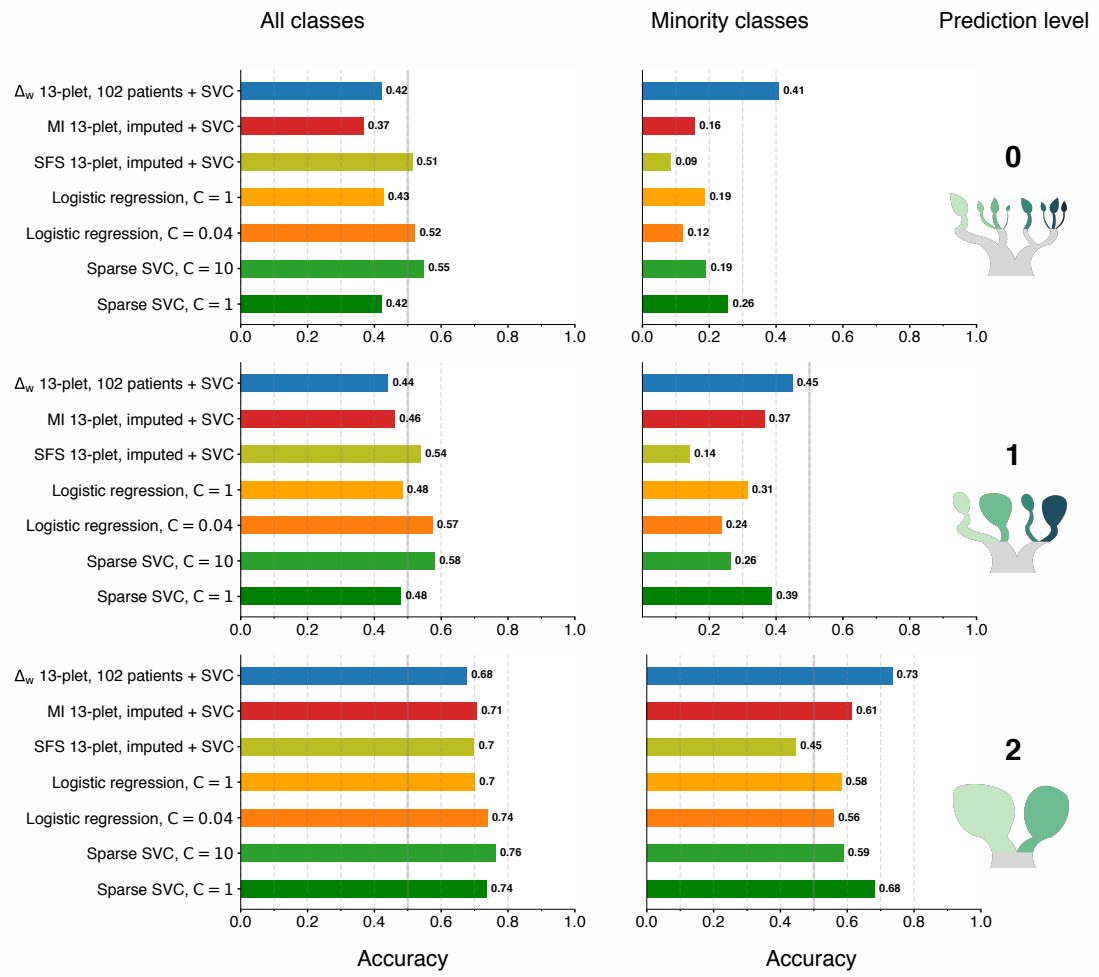


Figure 4.6: Accuraciess of prediction of several models, corresponding to Fig. 4.5 (SVC only), and compared with two classifiers, logistic lasso regression and sparse SVC. The accuracy corresponds to the fraction of patients predicted correctly at a given prediction level. The prediction level corresponds to the maximum tolerated distance of the true *vs.* the predicted class on the severity tree, as depicted on the right. 0: exact class predicted; 1: event and complication predicted; 2: event predicted.

have a base-2 Jensen-Shannon divergence of ≤ 0.06, in order to be representative of all classes in the full set.

The average predictive performance of the features selected in this manner is still very significant but is reduced as compared to what observed for the optimal 13-plets (Fig. 4.7). The prediction of the correct side of the tree (event *vs.* no event) is reduced by about 7% to roughly 70% and the correct class prediction

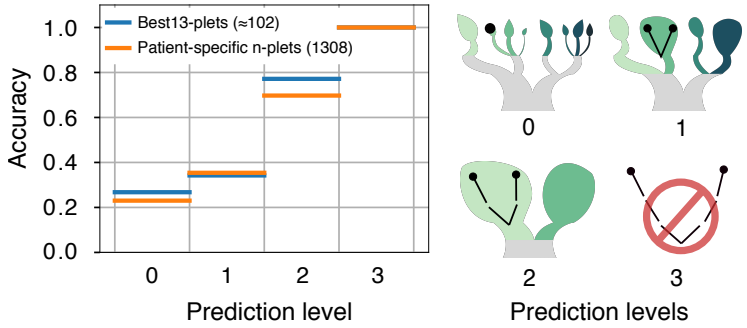


Figure 4.7: Accuracies of 10-NN predictions at given prediction levels by the optimal 13-plets (averaged over the ten best), which are available in roughly 102 patients, *vs.* accuracies using for each patient in the database (1308 patients) their optimal input feature tuple. The accuracy corresponds to the fraction of patients predicted correctly at a given prediction level. The prediction level corresponds to the maximum tolerated distance of the true *vs.* the predicted class on the severity tree, as depicted on the right. Together, the lines of one color can be considered the CDF of the fraction of patients predicted correctly.

drops from 27% to about 23%. This result is not surprising: The variables which turn out to be most informative happen to be simultaneously available only for a relatively small fraction of 102 patients, while the rest the patients do not have complete data for these features. Hence, their respective Δ_w-optimized n-tuple has higher (worse) Information Imbalance than the optimal 13-plet, which influences the prediction accuracy.

4.3.5 Identifying important but rarely available features

Approximately one third of the data are missing in the data set. For this reason, the optimal 13 features as described above are simultaneously available for only 102 patients. Moreover, missing values are not evenly distributed among the features: some "cheap" exams are performed routinely for all the patients, others are performed only for a small fraction of the patients. As a consequence, optimal features might not be available for a generic patient.

This result pushed us to develop a quality measure for the input features which takes into account the fact that Δ_w-optimized n-plets contain features that are "good" in two ways: firstly, they are *intrinsically important*, and secondly, *available together* in the same patients. For this quality measure we use all resulting patient-specific optimal feature tuples from the previous subsection 4.3.4 (Fig. 4.7).

We estimate, for each feature f, the number of patients NP_f for which f is included in the most informative Δ_w-optimized tuple (which is patient-specific).

The usage U_f of feature f is then estimated by the ratio between NP_f and the number of patients for which f is available. If U_f is close to one, then the feature has been chosen for each patient where it was available, and has a high intrinsic importance for the Information Imbalance towards the severity tree target classes.

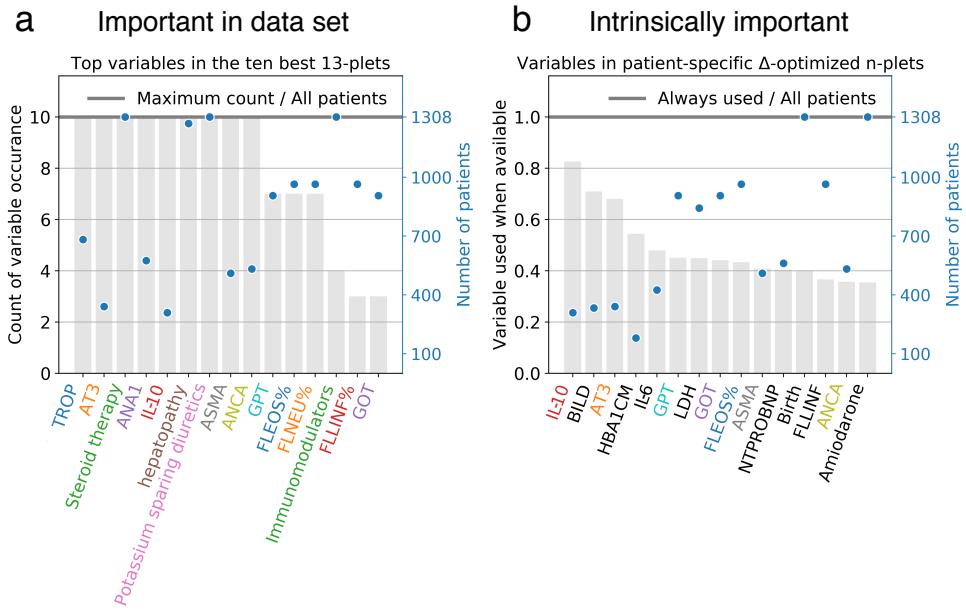


Figure 4.8: **a)** Statistics of the features present in the top ten 13-plets. The grey bar indicates the fraction of 13-plets in which the variable is present. The blue dots indicate in how many patients the feature is available (scale defined on the right y-axis). **b)** Intrinsically important features estimated by the "usage when available" statistic U_f. The gray bars indicates the value of U_f, which is large when a variable is always used when present. The blue dots are the same as in panel A.

Fig. 4.8a shows the features which are used in the 10 most predictive 13-plets (predictive performance of these in Fig. 4.3c) which, we recall, are present at the same time in $\sim$ 100 patients. 9 of the 13 features are used in all the ten best models. The blue dots indicate the fraction of patients for which the variable is available. For example, information about steroid therapy, which is used in all the 10 best models, is available for all the patients. AT3, also used in all the best models, is available for approximately 350 patients.

Fig. 4.8b on the other hand shows the value of the "usage when available " statistic U_f. Some of those variables are present in both sets of Fig. 4.8, namely the cytokine IL-10 (interleukin 10), the anticoagulant protein AT3 (antithrombin III), the autoantibodies ANCA and ASMA (antineutrophil cytoplasmic antibodies and

anti-smooth muscle antibody), the liver enzymes GPT and GOT (alanin aminotransferase and aspartate aminotransferase), and the percentage of eosinophils, FLEOS%.

The U_f statistic finds also intrinsically good predictors which are underrepresented due to missing values, and as such do not appear in the most used features of the Δ_w-optimized tuples. For the COVID-19 severity prediction some of these are direct bilirubin (BILD), the diabetes indicator glycated hemoglobin (HBA1CM), interleukin 6 (IL-6), and the enzyme lactate dehydrogenase (LDH), which indeed have a high value of U_f (panel b) but do not enter in the best model (panel a). IL-6 is a pro-inflammatory cytokine and has previously been linked with COVID-19 severity [87]. Abnormal bilirubin levels indicate sepsis, and the severity of patients could be linked to the fact that they have developed sepsis, and therefore a condition of a dysregulated systemic response. Glycated hemoglobin is linked to a condition of decompensated diabetes, which predisposes to infections. Diabetes is known to predispose severe COVID-19 infections [88].

4.4 Discussion

In this chapter we illustrate a first attempt to use the Information Imbalance [13] to perform feature selection. We considered a clinical database, which introduces two critical difficulties: missing data and the coexistence of variables of totally different nature (binary, categorical and real) in the dataset. While we believe that for the first problem (missing data) the approach described in this chapter is robust, and can be used in other contexts, the difficulties met in mixing variables of different nature in the same distance prompted us to further research, which will be described especially in chapter 6.

Technical discussion

Focusing the discussion on the results presented in this chapter, we considered a database of 1300 COVID-19 patients from Udine hospital with $\sim$ 150 features for each patient and one third of missing data. In order to deal with unbalanced classes in a clinical setting we had to include weights in the Information Imbalance definition. We find that the optimal feature tuples selected by our approach perform better in the k-NN prediction of COVID severity classes than two other standard feature selection methods, mutual information (a filter) and sequential feature selection (a wrapper), implemented in a standard statistical analysis package [76] (see Fig. 4.5). Regulated classifiers which do not include a prior feature selection step can reach slightly higher accuracies for the prediction of all classes (Fig. 4.6). This effect, however, does not translate to minority classes, which in our database

correspond to patients developing very severe symptoms. Information Imbalance feature selection followed by SVC classification outperforms all other methods in identifying those patients.

The classification task, as applied in our study, serves to validate the utility of the Δ_w selected features in distinguishing between severity classes of COVID-19, rather than introducing a complete novel classification model. A future direction of research could extend our algorithm into a framework including techniques for quantifying of the uncertainty of prediction, but is beyond the scope of the current chapter and would overshadow the core contribution of the presented feature selection method.

As common in most real world clinical data, also in this case the accuracy does not allow to perform exact prediction of the patient fate into their severity class (prediction level 0 in Figs. 4.3 and 4.5). Exact predictions, as tested in a leave-one-out-cross-validation approach, happen in 30% - 40% of cases, depending on which predictor is used and which classes is considered. However, in over 70% of the cases we were able to predict if the patient will suffer a serious event (death, transport to ICU or intubation, prediction level 2) or not. In this sense, the hierarchical tree structure of the output space is of advantage, because even if the exact prediction is not possible, a warning along a less stratified prediction level of the output tree is nevertheless possible. The feature tuples derived from our approach keep the high accuracy described above also for minority class patients, as opposed to tuples generated with other feature selection techniques.

One important advantage of the approach presented here is that it works without imputation[89], namely it does not require the preprocessing step of assigning missing values of the input features. While MI and SFS in their standard implementations need complete data sets and such require imputation, our approach finds patient-specific optimal input feature tuples in the original, incomplete data.

A limitation of the method is presented by the nature of the ground truth space. Since Information Imbalance finds input feature spaces which reproduce the neighborhood relationships observed in a ground truth space, it works best if the ground truth is either continuous, or at least has classes for which one can meaningfully identify a distance similar to our severity tree. The method, as formulated in this chapter, is less suitable for learning a ground truth distance which can take only a few values, and would not be appropriate for binary classification tasks. Furthermore, continuous input features are more likely to be chosen in small tuples, because they can carry more information than categorical variables. The optimal 13-plet of this study includes three binary "yes/no" variables (hepatopathy, steroid therapy, potassium sparing diuretics), proving that there are several informative binary features in this COVID-database that hold information complementary to the chosen numerical features.

The threshold of $\geq$ 100 patients without missing data and Jensen-Shannon divergence of $\leq$ 0.06 for Information Imbalance feature selection were practical choices, which can influence the specific results of the optimal tuple. While these choices have proven to balance performance and representativeness effectively, different informative feature tuples could be selected for different thresholds. The optimal 13-plet should be considered one out of several possible informative feature combinations in this data set.

Clinical discussion

The optimal variables of our data set include cytokines (such as IL-10), autoantibodies (ANA, ASMA, ANCA), and therapies that reduce the immune response (steroid therapy and immunomodulators as chronic home therapy). These findings are of medical interest because the pathogenic mechanisms that drive COVID-19 clinical deterioration can likely be contributed to systemic inflammation, disordered coagulation (AT3), and immune dysfunction. The cytokine storm that characterizes the unfavorable outcome of patients comprises classical markers of systemic inflammation such as IL-6, which is now largely disposable as a single diagnostic test also at urgent request in the majority of hospitals around the world. IL-6 increases during COVID-19 illness decline as patients recover, correlating with the severity of the disease course [87]. When IL-6 levels are already very high, the focus can be shifted to the degree of immunoparalysis and anti-inflammatory effort (IL-10) [87].

COVID-19 is known to alter the coagulation state and, in severe cases, lead to hypercoagulation, which is causally involved in negative patient outcomes. AT3 levels decrease in inflammatory conditions and AT seems to possess anti-viral properties [90]. The role of transaminases (GPT and GOT), and history of liver disease are also interesting. The hepatic consequences of an SARS-CoV-2 infection are recognized as an important component of COVID-19 and this aspect is most clinically relevant in patients with pre-existing cirrhosis [91]. There are several other potential contributors to abnormal liver biochemistries in COVID-19, including ischaemic hepatitis, hepatic congestion related to cardiomyopathy, and transaminase release due to the breakdown of skeletal and cardiac muscle [92].

This chapter also tested the features for their intrinsic importance decoupled from their availability in the data set by employing a simple usage statistic, which we called U_f. This analysis can be used to provide recommendations to clinicians for future data collections, because it identifies potentially important features for COVID severity prediction, which are, however, not abundant enough in our data set. For example, we find that conjugated (direct) bilirubin (BILD), is available for less than 30% of the patients in the current data set, yet it is selected very often, when present, for the patient-optimal prediction. A similar scenario is found

for glycated hemoglobin (HBA1CM) and interleukin 6 (IL-6). The collection of these features should be emphasized in future data collection efforts.

The use of clinical features and diagnostic features as biomarkers is of great clinical interest, in order to facilitate improved triaging and earlier therapeutic decisions. The model presented here could help the clinicians to focus on the variables of greatest interest in order to target the allocation of resources and escalation of care.

The analysis of patient severity in this chapter included many categorical variables, especially in the output space. We solved the issue of degenerate values and distances in these spaces by (a) developing a specific severity tree distance as target for the output space, and by (b) adding small random numbers to the degenerate values in the input space. In the following chapter 5 we extend the Information Imbalance framework to include categorical values and allow quantification of information between categorical and continuous spaces. As highly relevant use case we consider biodiversity data from the Amazon Rainforest.

Chapter 5

Treating categorical variables: Biodiversity data in ecology

Many datasets include categorical, non-ordinal variables, which may convey valuable information about other features but cannot be handled by methods that assume the variable's value carries intrinsic meaning. In the Information Imbalance approach, distances between data points are computed using feature values. However, commonly used distance measures depend on the magnitude of feature values, while categories are only labels, whose value has no specific meaning.

A relevant example of non-ordinal variables can be found in location-based features in ecological datasets, which will be analyzed in this chapter. In ecological studies, particularly those involve large geographic regions, the full population cannot be classified directly, so the area is sampled through smaller plots distributed across the region [93]. These plots are characterized by various features, including administrative units like countries and regions. Such metadata are categorical, typically non-ordinal and represented by a word string: The feature "region" could include 50 different geographical regions, and to use this feature in Information Imbalance, it needs to be encoded in numbers. Yet the nominal value of the numbers should not be considered an order, since their sequence is arbitrary. Despite this, categorical features can still provide valuable information about target features.

To account for this, this chapter introduces two new forms of Information Imbalance that can extract insights from categorical data. $\Delta_{con2cat}$ and $\Delta_{cat2con}$ are designed to capture the information content of a continuous feature about a categorical feature and *v.v.* Both only use the distance information from the continuous space, while considering instead the classes present in the categorical feature. We use these measures to analyze an Amazon Rainforest dataset.

5.1 Information Imbalance between categorical and continuous features

The dataset which will be analyzed in this chapter includes **categorical data without order**, namely not ordinal data. In these use cases, if a point j is not in the same class as point i, it is irrelevant in which other class it is, and it is always considered as "other". If a classifiers predict it in another class, the prediction is wrong, whatever the other class is. The specific challenge that we tried to address is quantifying the Information Imbalance between those variables and other variables which are instead real numbers, which can be sorted and ordered.

5.1.1 Predicting categorical features with continuous features

If a continuous groundtruth space A is informative about a categorical (binary, discrete) input space B, then the nearest neighbors according to the continuous A should be in the same "bin" in the categorical space B. Since all the points in one bin will have the same distance from any given point, these distances should therefore be assigned the same rank. Denoting by α_i the class of data point i we define

$$r_{ij}^B = \begin{cases} 0, & \text{if } \alpha_i = \alpha_j \\ N, & \text{otherwise} \end{cases}$$

With this definition of the ranks in space B we estimate the Information Imbalance as

$$\boxed{\Delta_{con2cat}\left(A \to B\right) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{N - N_{\alpha_i}} \sum_{j:r_{ij}^A=1}^{N} r_{ij}^B}, \tag{5.1}$$

where $N_{\alpha_i} = \sum_j \delta_{\alpha_i,\alpha_j}$ is the number of data points belonging to the same class of point i.

This definition is justified by considering the normalization factor of the Information Imbalance, which is $\frac{1}{\mathbb{E}_i(r_{ij})}$. $\mathbb{E}_i(r_{ij})$ is the expected average rank of any random point to point i. For the classic Information Imbalance $\mathbb{E}_i(r_{ij}) = \frac{N}{2}$ for every point. In the case of categorical variables the normalization can be derived taking into account that point i in class α_i (with $N_{\alpha(i)}$ points) has a class probability of $p_{\alpha(i)} = \frac{N_{\alpha(i)}}{N}$. The expected distance rank of a generic other point to i is $\mathbb{E}_i(r_{ij}) = p_{\alpha(i)}\, 0 + (1 - p_{\alpha(i)})\, N$. Hence, the normalization factor is

$$\frac{1}{\mathbb{E}_i(r_{ij})} = \frac{1}{p_{\alpha(i)}\, 0 + (1 - p_{\alpha(i)})\, N} = \frac{1}{N - N_{\alpha(i)}}. \tag{5.2}$$

Importantly, $\Delta_{con2cat}(A \to B)$ exhibits the same statistical behavior than the classic Information Imbalance $\Delta(A \to B)$, as described below in Limiting cases.

Limiting cases

If the continuous space A has perfect information about B, meaning that each point i's nearest neighbor in A, is in the same class as i according to B, all the ranks will be 0 and the result is $\Delta_{con2cat} = 0$.

In the opposite extreme case, in which each nearest neighbour from A is far in B (meaning in another class), each assigned rank is N, the sum of the ranks is N^2, and $\Delta_{con2cat} = 2\frac{N^2}{N^2} = 2$. This case corresponds to a situation in which the nearest neighbor according to distance A is systematically the furthest neighbour according to distance B and is the same values as for the classic Information Imbalance.

In the case of two equal sized classes with $N_{\alpha(i)}$ being half of N, the presented statistic happens to reduces to the classic Information Imbalance, because in both cases the expectation value of a random rank is $\frac{N}{2}$. This is coincidental and does not hold for any other class distribution.

Simple example

An example is shown in Fig. 5.1. The categorical (binary) variable on the x-axis is distributed in several different ways over a continuous variable (y-axis).

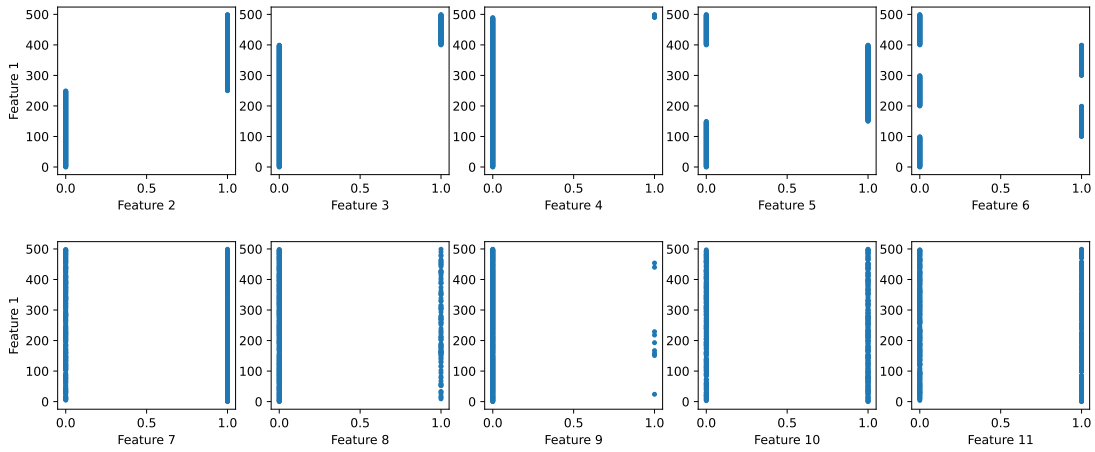


Figure 5.1: The x-axis shows a binary feature with different class distributions. The x-axis holds a continuous feature with full information about the discrete feature (top row) or no relationship with the binary feature (bottom row).

Fig. 5.2 shows the results of the calculations using the classic Δ and the version "continuous predicts categorical" $\Delta_{con2cat}$.

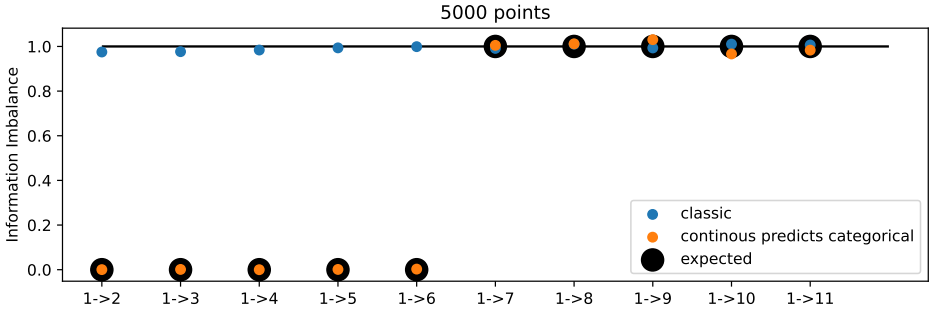


Figure 5.2: The classic (blue) and the continuous-to-categorical (orange) Information Imbalances for the example in Fig. 5.1 for 5000 points. Expected values of the Information Imbalance are marked with big black dots.

When categorical spaces 2 to 6 are predicted by the continuous feature, the classic Information Imbalance (blue) completely fails to capture the information content and outputs values of approximately 1, *i.e.* misidentifies the relationship as uninformative. $\Delta_{con2cat}$ correctly asses the full informativeness of feature 1 predicting these categorical features. Categorical features 7 to 11 are randomly distributed across the continuous feature. The lack of information is correctly identified by both frameworks.

Limitations

The presented statistic, $\Delta_{con2cat}$, is derived univariately, *i.e.* by considering only single features for building the distances between points. In the future one could extend the algorithm to multivariate distances. This could be done, *e.g.* by considering the degeneracy of multivariate distances between point i and all other points. Furthermore, the definition 5.1) is designed to estimate the Information Imbalances for "few" ($< \sqrt{N}$) classes. As the number of classes increases, the information that a continuous variable holds about a categorical one, as measured by $\Delta_{con2cat}$, decreases. In the appendix (Comparison of Classic Information Imbalance and Categorical Information Imbalance) we examine this phenomenon and consider the case of categorical but ordinal variables within the presented framework.

5.1.2 Predicting continuous features with categorical features

For the opposite case of predicting continuous features with categorical features, we follow this logic: A categorical ground truth space A is informative about a continuous input space B if all points within one class of space A are as close as possible in space B. If class α_i from the categorical space has, say, 20 points including point i, then in the most informative case, the other 19 points are distance ranks $1-19$ from point i in continuous space B. Based on this consideration, we define the Information Imbalance from categorical to continuous variables as

$$\Delta_{cat2con}\left(A \to B\right) := \frac{1}{N}\frac{1}{N_{class}}\sum_{k=1}^{N_{class}}\left(\frac{2\sum_{i=1,i\in\alpha}^{N_\alpha}\sum_{j=1,j\in\alpha}^{N_\alpha} r_{ij}^{B}}{N_\alpha^2}\right), \tag{5.3}$$

where N_α is the number of points in a class α, and N_{class} is the total number of classes. The underlying logic of eq. 5.3 is explained in more detail in the appendix in 'Logic of the Information Imbalance from categorical to continuous features'.

Simple example

In the toy example in Fig. 5.3 an increasing number of classes (x-axis, $[1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 16, 25, 50, 100, 250]$ classes) is distributed over a continuous variable (y-axis). Consider features on the x-axis predicting feature on the y-axis.

We note that this example includes best case scenarios, where varying numbers of discrete classes are distributed in a way to hold optimal information about the continuous feature. *E.g.* the x-feature in the second plot in the top row, which has two distinct values (classes), holds a certain amount of information about the continuous feature: When its value is 0, then the continuous feature takes on small values between $0-249$, while when its value is 1, then the continuous feature is larger, $250-500$. The information held by the categorical feature is certainly not perfect, because within the given continuous ranges we do not know exact nearest neighbor relationships.

If eq. 5.3 is applied to the the use case in Fig. 5.3 (500 data points), we find the expected behavior. If all points are in a single class, there is no information, and $\Delta_{cat2con} \approx 1$ (Fig. 5.4). When more and more classes are added, and the categorical space approaches the distribution of a continuous space, then $\Delta_{cat2con}$ approaches zero.

Fig. 5.4 shows that when the data is distributed over more and more classes, the classic Information Imbalance becomes reliable, while with few classes (< 10) the results are very noisy.

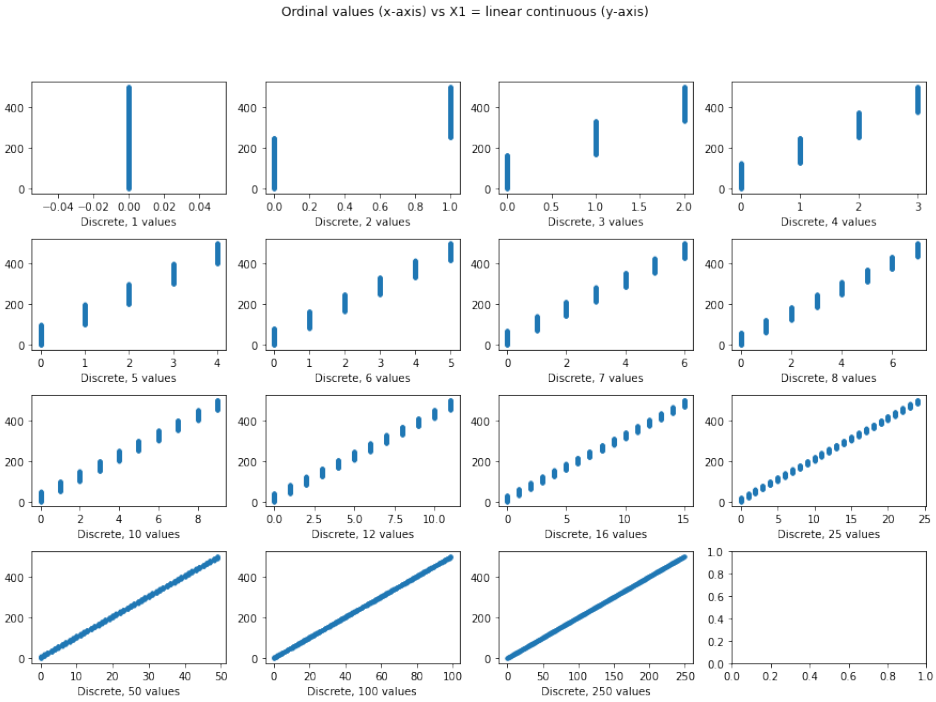


Figure 5.3: The x-axis shows a discrete feature with increasing number of classes. The y-axis represents a continuous feature.

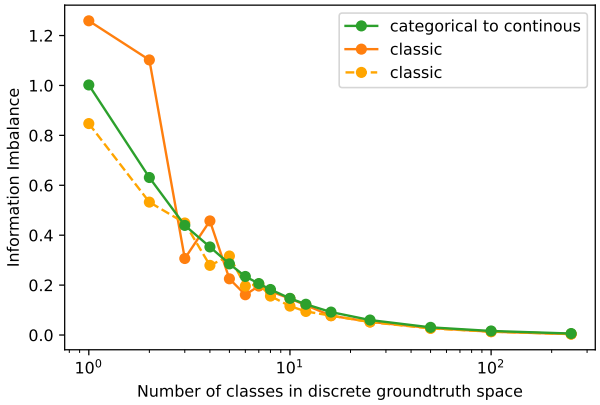


Figure 5.4: Information Imbalances for categorical features with various numbers of classes within 500 data points predicting a continuous feature. The categorical features are maximally informative about the continuous feature (see Fig. 5.3). $\Delta_{cat2con}$ in green. Orange and yellow show results using the classic implementation of Δ with two different parameter settings.

5.2 An application to the analysis of predictivity in ecology

5.2.1 The Amazon Rainforest

Figure 5.5: Example aereal view of the Amazon rainforest [94].

The lowland Amazonia rainforests of the Amazon River basin and the Guiana Shield span an area of nearly 6 million square kilometers. These forests are home to an estimated 390 billion trees (diameters at breast height, *i.e.* at 1.30 meters of 10 cm or greater) [95]. It is a cradle for life and biodiversity, and home to an estimated 15.000 tree species alone [95]. In comparison, in Europe we have around 500 tree species in total [96]. However, a cumulative total of 17% of the Amazon Rainforest was deforested by 2019, with the majority for agricultural use, and an additional 17% classified as degraded due to logging, fires, and other human activities [97]. An example can be seen in image 5.6.

Furthermore, in a business as usual scenario estimates that by 2050, about 40% percentage of the original Amazon Rainforest will deforested [24].

This year in August and September, the Amazon forest saw the worst wild fires in the last two decades, with smoke covering large parts of South America [99] (see

Figure 5.6: Deforestation of secondary growth forest around Km 114 of the BR-163 highway. Tapajós National Forest to the left. [98].

image 5.7), with health warnings all over the continent, even in distant cities like Buenos Aires [100].

The stresses of deforestation, droughts and wildfires, together with warming temperatures and climate change, pose a great risk to the Amazon: The Amazon rainforest system may soon cross a tipping point, where the stable interactions that dominated the forest's interaction with environmental conditions (the climate) can be replaced by other feedback systems. When this happens to the Amazon rainforest, large scale forest collapse and widespread, self-reinforcing savannization are the result [101]. This new system will be dominated by a positive feedback loop, exacerbated and exacerbating climate change. Any process intensifying the stress on the climate system or the Amazon forest system is hence expected to further deteriorate both [102].

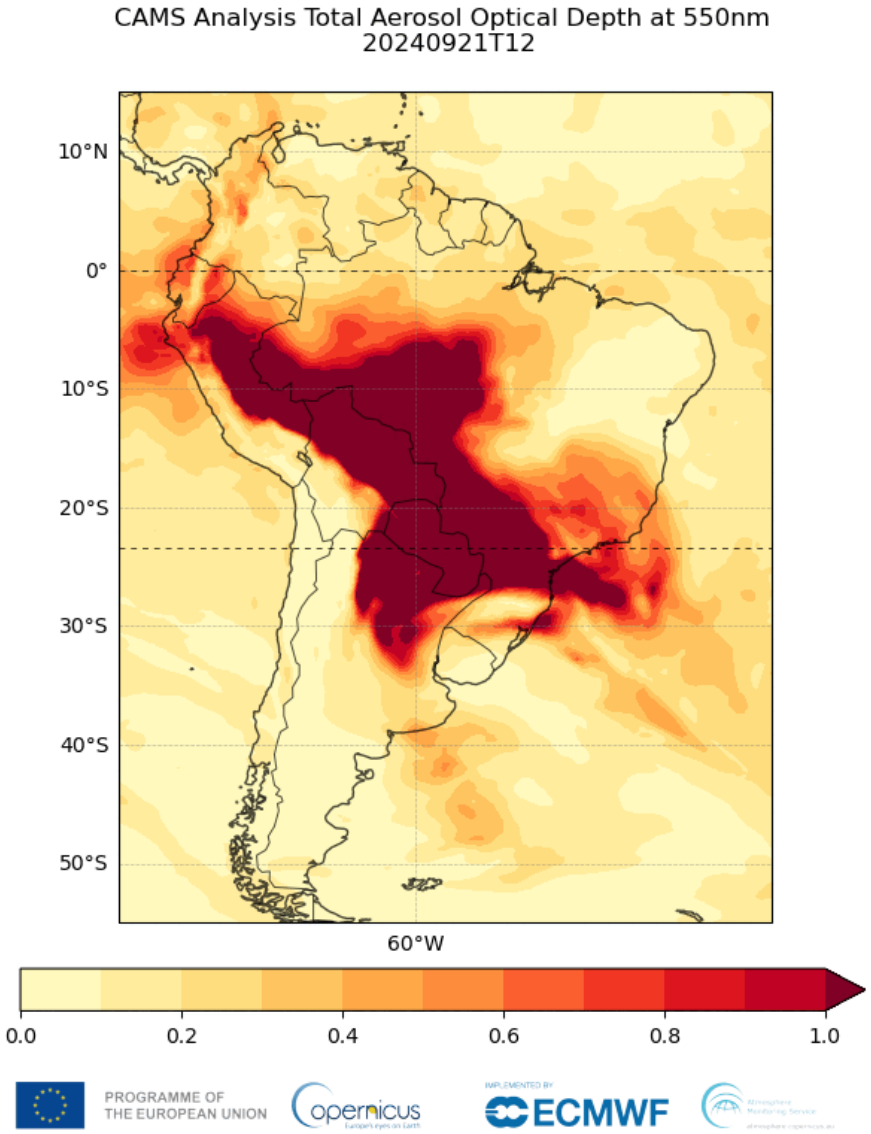


Figure 5.7: Total aerosol optical depth (AOD) analysis at 550 nm indicating smoke transport over South America on September 21st, 2024, by the Copernicus Atmosphere Monitoring Service (CAMS) [99].

5.2.2 Biodiversity and related estimators

Biodiversity is measured in many ways, most frequently via its correlating proxy, species diversity. Species diversity is a measure of the variety and abundance of different species within a community, ecosystem, or geographic area, which combines two main components [103]:

1. Species richness R: The total number of different species present in an area. It simply counts how many species exist without considering their abundances.

2. Species evenness E: The relative abundance of each species in an area. Evenness measures how evenly individuals are distributed among the species. A community where all species have similar abundances has high evenness, while a community dominated by one or a few species has low evenness.

Species richness R can be considered a richness density, since $R = \frac{S}{area}$, where S is the number of species in the area sample. Evenness (E) is calculated from diversity and richness, not directly measured. It is often treated as a secondary calculation to understand how much of the diversity is due to evenness rather than just the number of species. Species diversity D, species richness R, and evenness E, then have the following relationship [103]:

$$D = R \cdot E \tag{5.4}$$

There are many estimators of species diversity and evenness. The software Bioverse features over 200 diversity indexes [103]. Among the most used proportional abundance indexes which describe species diversity, are

- Fisher's alpha [104, 105], $\alpha = S/ln(1 + \frac{N}{\alpha})$ (solved iteratively; emphasizes number of species),
- the Shannon-Wiener Index [106], $H' = -\sum(p_i ln(p_i))$ (an entropy measure; emphasizes rare species) and
- the Inverse Simpson's Index [107], $D = \frac{1}{\sum p_i^2}$ (emphasizes common species).

Most diversity indices aim to be independent of the area sampled, but in practice they can still be influenced by the area indirectly. When the sampled area is increased one tends to find more species [105] and their value can change [103]. Nevertheless, they are often used to estimate of species richness in large areas, following three steps [105, 25]:

1. First, Fisher's alpha is calculated with the sample number of stems N_s and sample number of species S_s,
2. then the number of trees is extrapolated to the full area, and finally
3. the overall species richness is estimated as $S = \alpha\, ln(1 + \frac{N}{\alpha})$.

This estimation assumes a log series behavior [25, 105] of the species abundance distribution ("SAD"; histogram of abundance data, *i.e.*individual counts, for each species in the sample). It has been criticized because in reality SADs often do not follow log series [25].

In real applications, assessing species diversity, especially at biogeographic extent (large scales) is difficult [103]. Two common types of estimations are:

(1) Extrapolation of species richness relationships, SRRs (only presence/absence data at each location for each species are required) using a 'species accumulation curve' [93]. It shows the number of species discovered *vs.* an effort of collecting them, *e.g.* the number of individuals sampled or a proxy such as area or biomass sampled, or hours of collection. The true species richness R_{max}, can be estimated by considering the asymptote of the curve and might be corrected in various ways, such as with maximum likelihood optimization [93].

(2) Estimation by species abundance distribution, SAD (abundance data for each species at each location required) [103]. These methods estimate the true species richness R_{max} by leveraging the number of rare species as an indication for the number of unobserved species [103]. Typical examples include the non-parametric Chao1 [108] and Chao2 [109] estimators, first- and second-order Jackknive and bootstrapping [93]. However, non-parametric estimators are ineffective when the total area surveyed is small (e.g. less than 2% as often the case in nature reserves), and many rare species are not observed.

Among the many active research questions in biodiversity research, two important ones are:

(1) On large scales (biogeographic scales), sampling units are often just species lists from various sources. Research is needed on how to combine data from different sources optimally and to design better sampling schemes. Should one sample a greater number of smaller sampling units or less and larger units [103]? When and how to include historical digitized data [110]? Also the integration of citizen science [111] with other data could be evaluated.

(2) For conservation we need to know how changes in community composition affect ecosystem function. In order to establish the most effective protected areas, one needs to know the current diversity, but, importantly, also be able to maintain diversity while species are lost. A metric is needed to indicate whether rare species, which are more likely to go extinct, uphold the most diversity. That metric should encompass abundance, phylogenetic information and evenness [103]

In the following sections we will introduce an ecological dataset of the Amazon Rainforest including various measures of species richness, and several categorical variables, which, importantly do not encode an order. These are the variables that have to do with the geographic regions of data collection, including binary variables.

5.2.3 Methods

The dataset

We applied the aforementioned framework of categorical Information Imbalance, along with the classic Information Imbalance, to examine the pairwise information content among 27 features from a 2023 study on Amazon rainforest biodiversity [105]. The features are described in detail in table 5.1. After excluding 233 entries with missing values for the feature "bases_sum," the dataset comprised 1,819 points. Following [1], we added small random numbers to duplicated values in features classified as "continuous" (as indicated in the "Interpreted type" column) in order to create non-degenerate distances between points when using these features univariately for the distance rank calculation.

Information Imbalances between continuous and categorical variables in the Amazon biodiversity data set

All subsequent results are derived from univariate Information Imbalance analyses, unless stated otherwise: We examined the information content of individual features in relation to other single features. For all continuous features, we applied the classic Information Imbalance method from DADApy [112] `MetricComparisons`, using the function `return_inf_imb_matrix_of_coords()`. To determine the Information Imbalance from continuous features to categorical ones, we utilized the function `imb_cont2cat`, as specified as $\Delta_{con2cat}$ in eq. 5.1. Conversely, we used the function `imb_cat2cont`, represented as $\Delta_{cat2con}$ in eq. 5.3. The code for these functions can be found in the appendix in Code for Categorical Information Imbalance.

Network graph visualizations

Network graph visualizations were created with Cytoscape [113] (version 3.10.2) after exporting (1 - pairwise Information Imbalances) as a square matrix, *i.e.* the pairwise interaction matrix. The diagonal was filled with zeros. The plugin app aMatReader was used to import the matrix files as an adjacency matrix. The visualization is a degree sorted circular layout.

Table 5.1: The 27 features from the Amazon biodiversity study [105] considered, extracted from the supplementary material files "PlotData.csv" and "PlotsAbiotic.csv". 233 points with missing values in the feature "bases_sum" were deleted, leading to a data set of 1819 points. The first column is their name as used in this thesis, the 5^{th} column the name according to the aforementioned .csv files. "Nr. classes" describes how many unique values the feature has, and "Interpreted type" refers to how the feature was treated according to this thesis.

Feature	Category	Nr. classes	Interpreted type	Name in [105] SM	Explanation
Country	location	9	categorical	Country	Country in which plot is located
Subdivision	location	57	continuous	Subdivision	Subdivision of country
Region	location	6	categorical	region	Region in which plot is located
Longitude	location	1556	continuous	longitude	Longitude of the plot
Latitude	location	1645	continuous	latitude	Latitude of the plot
Pebas region	location	2	categorical	Pebas	Pebas region or not
Forest	abiotic	7	categorical	forest	Forest-soil type combination
Flooded	abiotic	2	categorical	Flooded	Floodplain forests (Várzea and Igapó), permanently inundated terrain, or waterlogged swamps
NutrientPoor	abiotic	2	categorical	Podzol	Very nutrient-poor white sand podzols (terrains)
Terrafirm	abiotic	2	categorical	/	Terrafirm terrain; not flooded and not nutrient poor
year	metadata	52	continuous	year_est	Year of establishment of the plot
ColD	metadata	432	continuous	ColD	Collection density/intensity
PlotSize	measure of area	60	continuous	PlotSize	Size (ha) of the plot
N_plot	tree count	655	continuous	N	Number of trees per plot; counted directly
D_ha	tree count	586	continuous	D	Number of trees per hectare (tree density)
S_plot	diversity index	271	continuous	S	Number of species per plot (tree species richness per plot); counted directly
S_ha	diversity index	277	continuous	S.ha	Number of species per hectare (tree species richness per hectare); spatial prediction
S_500	diversity index	247	continuous	S.500	Number of species per 500 trees (tree species richness per 500 stems); spatial prediction
fa_plot	diversity index	1798	continuous	fa.plots	Fisher's alpha diversity index per plot; calculated iteratively from S_plot and N_plot
bases_sum	abiotic	1552	continuous	SB	Log(sum of bases); soil parameter encoding soil fertility
pH	abiotic	1734	continuous	pH	Acidic = low pH; soil parameter encoding soil fertility
AnnualRain	climatic	773	continuous	AnnualRain	Annual Rainfall (mm)
CWD	climatic	645	continuous	CWD	Cumulative Water Deficit (mm); measure of drought stress
MCWD	climatic	910	continuous	MCWD	Maximum Cumulative Water Deficit (mm); maximum drought stress over a year
AnnualT	climatic	774	continuous	AnnualT	Annual average temperature (°C)
TSeas	climatic	774	continuous	TSeas	Temperature seasonality (standard deviation of monthly temperature)
AnSeas	climatic	774	continuous	AnSeas	Annual seasonality in rainfall; a measure of the seasonal variation in rainfall

5.3 Results

5.3.1 Information Imbalance between pairs of variables

Firstly, we computed the Information Imbalance between individual features. $\Delta_{con2cat}$ and $\Delta_{cat2con}$ were used where applicable. The values are plotted all together in Fig. 5.8a, which shows the value of $\Delta(A \to B)$ versus $\Delta(B \to A)$ for each pair of variables A and B. The pairwise Information Imbalances between individual features exhibit a range of behaviors.

The relationships range from symmetric informative relationships, to symmetric but less informative relationships, and to asymmetric relationships.

Fig. 5.8b illustrates this concept: $\Delta(\text{Country} \to \text{Subdivision})$ is high (uninformative), while $\Delta(\text{Subdivision} \to \text{Country})$ is low (informative). This suggests that knowing the sub-region within a country provides significant information about the country, but the reverse is not true.

Fig. 5.8c highlights the relationships between features and the diversity index 'species richness in 500 trees' (S_500). The species richness itself is not particularly informative about other features (except for related diversity indices), whereas several other features, particularly climatic variables, are moderately informative about species richness ($\Delta \geq 0.49$). For instance, annual average temperature provides some information about species richness. This can be interpreted as follows: while species richness does not predict the climate, certain climatic conditions can, to some extent, predict species richness. Additionally, collecting intensity (ColD) is informative about species richness, a point that will be discussed in the following subsection.

Interestingly, feature pairs with asymmetric Information Imbalance relationships tend to not exhibit correlations. Both, collecting intensity and annual average temperature have very low Pearson correlation coefficients with species richness (Figs. 5.8d and e), yet, according to Information Imbalance, they are moderately informative. However, this informativeness is unidirectional.

5.3.2 Information Imbalance network graphs

Network graphs can be employed to visualize pairwise informative relationships by using bold, dark arrows to represent informative connections. Each pair of features is depicted with two arrows, corresponding to the directions $(A \to B)$ and $(B \to A)$. For these network graph visualizations, it is desirable to assign higher values to more informative relationships so that the thickness of the arrows reflects the degree of information content, where thicker arrows correspond to smaller Information Imbalances. Fig. 5.9 illustrates the network for the present use case of 27 ecological variables, with two different cut-off thresholds.

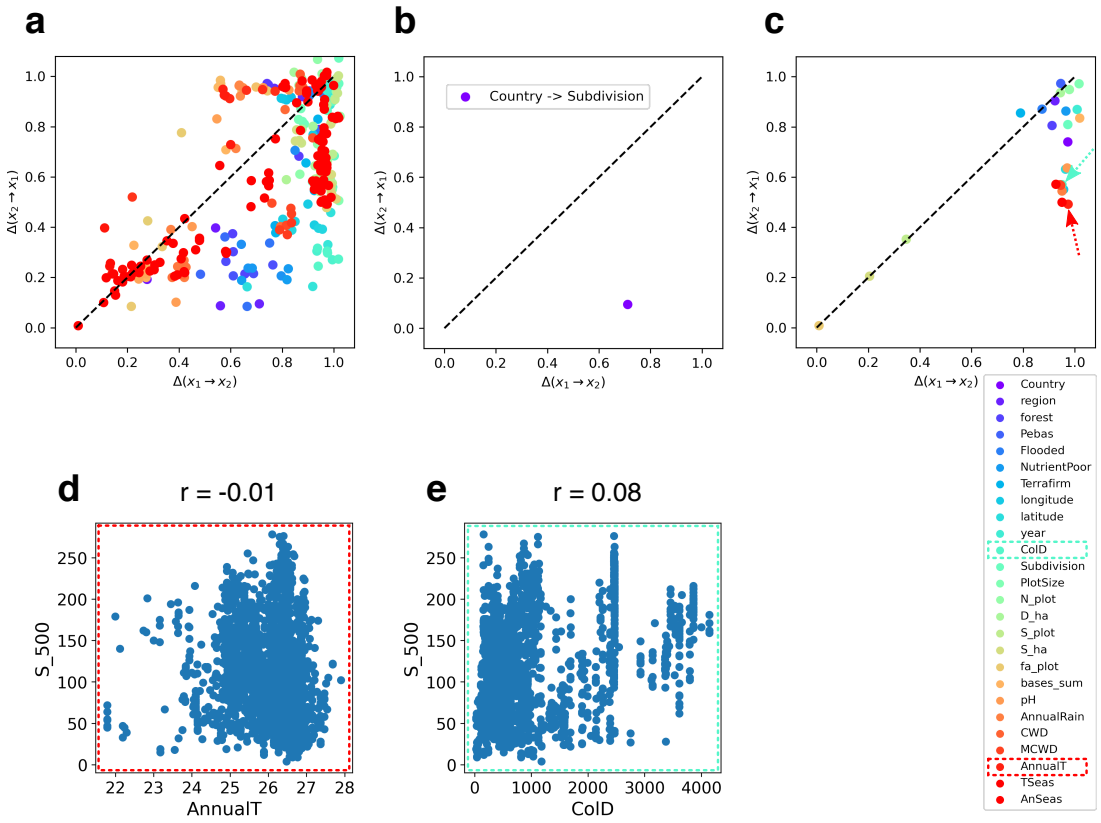


Figure 5.8: **a, b, c)** The pairwise Information Imbalance $\Delta(A \to B)$ and $\Delta(B \to A)$ plotted into Information Imbalance planes. **a)** All pairwise imbalances between the 27 features. There are many symmetric relationships with varying information content, and asymmetric relationships. **b)** Proof of concept: $\Delta(Country \to Subdivision)$ is high (uninformative), while $\Delta(Subdivision \to Country)$ is low (informative). **c)** Information Imbalances from and to the species richness in 500 trees, $\Delta(S\_500 \to Feature)$ on x-axis and $\Delta(Feature \to S\_500)$ on y-axis. **d and e)** Scatter plots of the species richness in 500 trees (S_500) *vs.* the annual average temperature (AnnualT) and collection intensity (ColD), respectively, with the Pearson correlation coefficient stated above the plots.

In Fig. 5.9a, we observe that the four diversity indices—S_plot, S_ha, fa_plot, and S_500 (highlighted in the pink circle)—do not exhibit highly informative relationships with other features, particularly in the outgoing direction. Species richness does not appear to provide substantial information about abiotic, climatic, or location-based variables. As previously noted in Fig. 5.8, certain features, particularly climatic ones, hold moderate amounts of information about species richness, as indicated by the yellow incoming arrows. In contrast, stronger relationships are

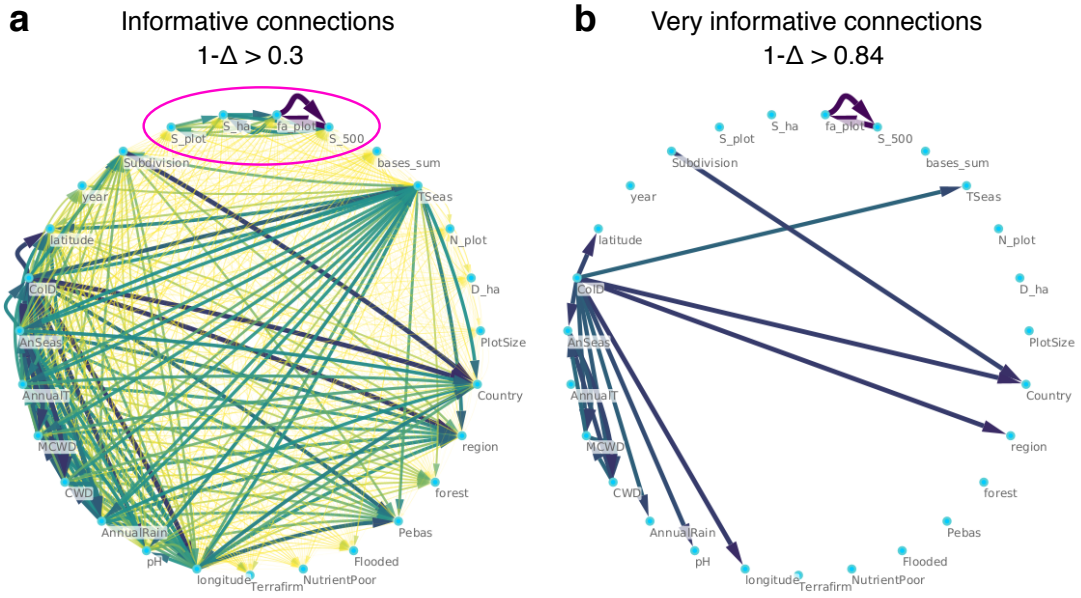


Figure 5.9: Network graphs can be used to visualize pairwise informative relationships, by featuring bold, dark arrows for an informative relationship (thicker arrow for smaller Information Imbalance). **a)** Relationships between features with Information Imbalances smaller 0.7. **b)** Only very informative relationships are displayed, with $\Delta \leq 0.16$.

observed between other features.

To focus on the strongest relationships, Fig. 5.9b visualizes only very informative connections, specifically where $\Delta < 0.16$. There is nearly perfect information between two of the diversity indices: $\Delta(fa\_plot \rightarrow S\_500)$ and $\Delta(S\_500 \rightarrow fa\_plot)$, both carry an Information Imbalance ≈ 0. This is not surprising, as the species richness in 500 trees (S_500) is a monotonic function of Fisher's alpha diversity index, $S\_500 = \alpha\, ln(1 + \frac{500}{\alpha})$.

A second noteworthy observation is that collecting intensity (ColD) is highly informative about several climatic variables, soil pH, and certain location-based variables. At first glance, this may seem counterintuitive: shouldn't location and climate be more predictive of where data collection is most intense? The answer is that while location and weather are indeed predictive to some degree, they are less informative than the reverse relationship. In simple terms, not every region with favorable environmental conditions has high data collection intensity, but when collection intensity is high at a particular plot, it is likely that favorable climatic and geographic conditions are present. Although weaker, collecting intensity also holds some information about species richness, indicating that the sampling ef-

fort has not yet reached its asymptote. In other words, continued sampling still provides additional information about species richness. This relationship is expected to diminish as sampling coverage and intensity increase—a particularly challenging task in the Amazon. The fact that the relationship is already not very strong ($\Delta(\text{ColD} \to S\_500) = 0.57$) highlights the remarkable effort of ecologists in characterizing the tree species richness of the world's largest rainforest.

5.4 Discussion

The aim of many studies, including ours, is to explain species richness, or other biodiversity proxis, using features that are easier to measure.

The two versions of Information Imbalance dealing with categorical (non-ordinal) values, $\Delta_{con2cat}$ and $\Delta_{cat2con}$, were applied to analyze categorical features, including features of geographic region and binary ("yes"/"no") features. They correctly identified the asymmetric information between "Country" and "Subdivision" of the country: The knowledge of a subdivision automatically implies the knowledge of a country, but not vice versa.

The Information Imbalance analysis provides insights into the relationships between features in the Amazon biodiversity dataset. Notably, species richness in 500 trees (S_500) shows asymmetric relationships with climatic and location-based variables, where climatic features such as annual average temperature hold some predictive value for species richness, but not vice versa. This indicates that while the richness of species does not provide information about environmental conditions, certain environmental factors can moderately predict biodiversity levels. Also the authors of the database found the climatic variables, especially cumulative water deficit, moderately predictive of species richness by linear regression [105]. However, with the most informative relationship being $\Delta(AnnualT \to S\_500) \approx 0.49$, we can conclude that no single feature in this data set holds enough information to predict species richness. This also shows the need to expand the Categorical Information Imbalance framework as presented here to multi-feature application.

Interestingly, these asymmetric relationships are invisible to correlation. The Pearson correlation coefficient between the annual average temperature and the species richness in 500 trees is $r = -0.01$, while from our distance-based perspective, the annual average temperature does hold a moderate amount of information about the species richness.

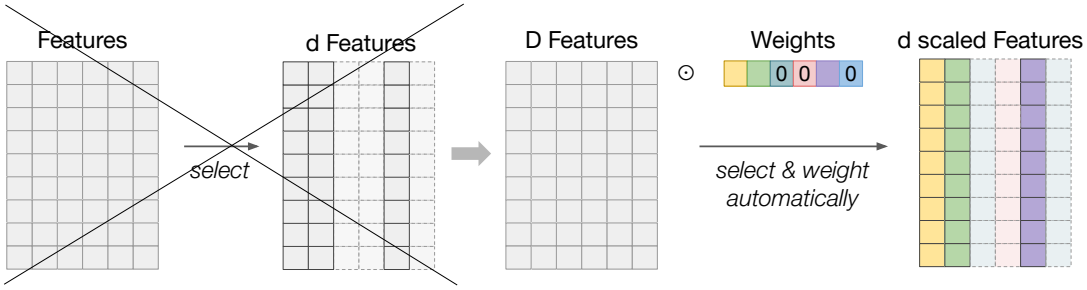
We find symmetric, nearly perfect information between two of the diversity indices: Fisher's alpha per plot, fa_plot, and the species richness in 500 trees, S_500. This is due to the fact that S_500 is a monotonic function of Fisher's alpha. Hence, from an information-theoretic perspective, the use of one index or

the other is essentially equivalent, and ecologists may choose whichever is more practical or intuitive for their purposes. It has been noted [105] that S_500 is easier to understand.

Also collecting intensity (ColD) holds some predictive power for species richness (but not vice versa), suggesting that more intense data collections can still contribute valuable information about biodiversity, even though the value is moderate. A similar notion was mentioned in the study [105]. This result highlights the remarkable achievement of ecologists who have been sampling the Amazon until now, as well as the ongoing need for more sampling in many plots to fully capture the richness of Amazonian biodiversity. This emphasizes the importance of understanding how sampling strategies may influence biodiversity data and the ongoing effort to achieve representative sampling across the vast and ecologically complex Amazon region.

Chapter 6

An optimizable Information Imbalance for high dimensionsional data



In the last two chapters we saw methods to deal with categorical and missing data in Information Imbalance analysis. However, important inefficiencies remain:

- Enumerating all possible feature sets leads to a combinatorial explosion.
- The correct alignment for different units of measure and importance.

This chapter introduces an attempt to treat these problems in a new manner, which does not require performing a combinatorial search. We introduce a variant of the Information Imbalance which can be optimized by gradient descent. This allows performing feature selection and determining feature weights in a unified framework. In particular, the value of the new statistic can be used to determine the optimal dimension of a reduced feature set. The method's capabilities are demonstrated on real world applications: Selecting collective variables (CVs)

from a molecular dynamics simulation, and selecting input features for a machine learning potential.

6.1 Introduction to automatic feature selection and weighting

As described in chapter 2, feature selection methods can be broadly divided into wrapper, embedded, and filter methods [43]. Information Imbalance belongs to the filter methods, a group of efficient feature selection methods which are independent of a downstream task and make use of a separate criterion to rank features.

While unsupervised filter techniques exploit the topology of the original data manifold in various ways [54, 55, 56, 57, 58], the supervised ones calculate a statistic in relation to a ground truth.

The classic supervised filters include correlation coefficient scores, mutual information [59], chi-square tests, and ANOVA methods [60], which are efficient but typically consider one feature at a time, resulting in selected subsets with redundant information [36]. More advanced filters can select subsets of features and assign relative weights to the features. The relief algorithm and its variants [63, 62] employ nearest neighbor information to weight features. However, the identified subsets can include redundant features [62].

Many of these methods are limited by the data types that are permissibly for input and output features. Furthermore, the field of feature selection is lacking the numerous powerful and out-of-the-box tools that are available in related fields such as dimensionality reduction. For the several remaining challenges, there is no user-friendly method which solves them all:

The first, shared challenge in most of these feature selection approaches is related to the choice of the number of variables that are actually necessary to describe the system. A lower bound to such a number is provided by the intrinsic dimension [40]. Moreover, if one wants to visualize the data within a single graph, the number of variables is necessarily limited to two or three. This typically implies neglecting part of the information, and poses the problem of choosing which variables should be retained.

A second complication arises when the variables are heterogeneous; in many cases, a data point is defined by features with different nature and units of measures [39]. For example, in atomistic simulations, one can describe a molecule in water solution by providing the value of all the distances between the atoms of the molecule, which are measured in nanometers, together with the number of hydrogen bonds that they form with the solvent, which are dimensionless. In order to mix heterogeneous variables in a low-dimensional description, feature se-

lection algorithms should enable the automatic learning of feature-specific weights to correct for units of measure [39] and information content [114].

A third problem is posed by the combinatorial explosion, which arises from explicit enumeration of all possible feature subsets. This enumeration is accepted for the sake of finding non-redundant subsets of features by so called supervised feature subset evaluation methods [61, 62]. When various relative scalings of the features have to be considered for unit alignment or importance weighting, the enumeration approach described above becomes even more unfeasible.

In this chapter, we propose a feature selection filter algorithm which mitigates many of the aforementioned problems. Our approach aims to find a small subset of features that can best reproduce the neighbors of the data points based on a target feature space that is assumed to be fully informative. The algorithm finds, for each input feature, an optimal *weight* that accounts for different units of measure and different importance of the features. It also provides information on the optimal number of features.

The approach builds on the Information Imbalance (Δ) as introduced in chapter 3, which allows comparing the information content of distances in two feature spaces [13]. In all previous works based on Information Imbalance, the analyses were either univariate (see chapter 5), or the distance space maximizing the prediction quality has been constructed by means of strategies including full combinatorial search of the optimal features [15], greedy search approaches [1] (see chapter 4) and grid search optimization of scaling parameters [115], with drawbacks related to the algorithm efficiency.

Here we make a major step forward by introducing the Differentiable Information Imbalance DII, which allows learning the most predictive feature weights by using gradient-based optimization techniques. The input feature space, as well as the ground truth feature space (targets, labels), can have any number of features. This provides a data analysis framework for feature selection where the optimal features and their weights are identified automatically. Moreover, carrying out the optimization with a sparsity constraint, such as L_1 regularization, allows finding representations of a data set formed by a small set of interpretable features. If the full input feature set is used as ground truth, then the approach can be used as an unsupervised feature selector, whereas it acts in a supervised fashion if a separate ground truth is employed [67]. To our knowledge, there is no other feature selection filter algorithm implemented in any available software package which has above mentioned capabilities. The DII algorithm is publicly available in the Python package DADApy [3] and a comprehensive description can be found in the according documentation [68], which includes a dedicated tutorial.

In the following, we will first show the effectiveness of our method on artificial examples in which the optimal set of features is known. Then we move to a real-

world application and show that our approach allows addressing one of the most important challenges in molecular modeling and solid state physics: Identifying the optimal set of collective variables (CVs) for describing the configuration space of a molecular system. As a second application, we use our method to select a subset of Atom Centered Symmetry Functions (ACSFs), descriptors of atomic environments, as input for a Behler-Parrinello machine learning potential [116], which learns energies and forces in systems of liquid water. In the same application, we show that Smooth Overlap of Atomic Orbitals (SOAP) [117, 118] descriptors can be used as ground truth to choose informative subsets of ACSF descriptors.

6.2 Differentiable Information Imbalance

Given a data set where each point i can be expressed in terms of two feature vectors, $\mathbf{X}_i^A \in \mathbb{R}^{D_A}$ and $\mathbf{X}_i^B \in \mathbb{R}^{D_B}$ $(i = 1, \ldots, N)$, the standard Information Imbalance $\Delta(d^A \to d^B)$ provides a measure of the prediction power which a distance built with features A carries about a distance built with features B. The Information Imbalance is proportional to the average distance rank according to d^B, restricted to the nearest neighbors according to d^A [13]:

$$\Delta\left(d^A \to d^B\right) := \frac{2}{N^2} \sum_{i,j:\, r_{ij}^A = 1} r_{ij}^B. \tag{6.1}$$

Here, r_{ij}^A (resp. r_{ij}^B) is the distance rank of data point j with respect to data point i according to the distance metric d^A (resp. d^B). For example, $r_{ij}^A = 7$ if j is the 7th neighbor of i according to d^A. $\Delta\left(d^A \to d^B\right)$ will be close to 0 if d^A is a good predictor of d^B, since the nearest neighbors according to d^A will be among the nearest neighbors according to d^B. If d^A provides no information about d^B, instead, the ranks r_{ij}^B in Eq. (6.1) will be uniformly distributed between 1 and $N-1$, and $\Delta\left(d^A \to d^B\right)$ will be close to 1. As shown in ref. [115], the estimation of Eq. (6.1) can potentially be improved by considering k neighbors for each point. Considering d^B as the ground truth distance, the goal is identifying the best features in space A to minimize $\Delta(d^A \to d^B)$. If the features in A and the distances d^A are chosen in such a way that they depend on a set of variational parameters $\boldsymbol{w}$, finding the optimal feature space A requires optimizing $\Delta\left(d^A(\boldsymbol{w}) \to d^B\right)$ with respect to $\boldsymbol{w}$. However, Δ is defined as a conditional average of ranks, which cannot be minimized by standard gradient-based techniques.

Here we extend Eq. (6.1) to a differentiable version that we call Differentiable Information Imbalance (DII) in order to automatically learn the optimal distance $d^A(\boldsymbol{w})$. We approximate the non-differentiable, rank-dependent sum in Eq. (6.1) by introducing the softmax coefficients c_{ij}:

$$\boxed{DII\left(d^A(\boldsymbol{w}) \to d^B\right) := \frac{2}{N^2} \sum_{\substack{i,j=1 \\ (j \neq i)}}^{N} c_{ij}(\lambda, d^A(\boldsymbol{w}))\, r_{ij}^B}, \tag{6.2}$$

where

$$c_{ij}(\lambda, d^A(\boldsymbol{w})) := \frac{e^{-d_{ij}^A(\boldsymbol{w})/\lambda}}{\sum_{m(\neq i)} e^{-d_{im}^A(\boldsymbol{w})/\lambda}}. \tag{6.3}$$

The coefficients c_{ij} in Eq. (6.2) approximate the constraint $r_{ij}^A = 1$, such that $c_{ij} \to \delta_{1, r_{ij}^A}$ as $\lambda \to 0$ (δ denotes the Kronecker delta). Therefore, in the limit of

small λ, the DII converges to Δ (see also [68]):

$$\lim_{\lambda \to 0} DII\left(d^A(\boldsymbol{w}) \to d^B\right) = \Delta\left(d^A(\boldsymbol{w}) \to d^B\right) . \tag{6.4}$$

For any positive and small λ, the quantity $DII\left(d^A \to d^B\right)$ can be seen as a continuous version of the Information Imbalance, where the coefficients c_{ij} assign, for each point i, a non-zero and exponentially decaying weight to points j ranked after the nearest neighbor in space d^A. The parameter λ is chosen according to the average and minimum nearest neighbor distances (see subsection 6.2.1).

The DII is differentiable with respect to the parameters $\boldsymbol{w}$ for any distance d^A which is a differentiable function of $\boldsymbol{w}$. In this chapter, we assume that the variational parameters are weights, $\boldsymbol{w} = (w^1, \dots, w^{D_A})$, scaling the features in space A as $\boldsymbol{w} \odot \mathbf{X}_i^A = (w^1 X_i^1, \dots, w^{D_A} X_i^{D_A})$ (the symbol $\odot$ denotes the element-wise product). We construct $d^A(\boldsymbol{w})$ as the Euclidean distance between these scaled data points, $d_{ij}^A(\boldsymbol{w}) = \|\boldsymbol{w} \odot \left(\mathbf{X}_i^A - \mathbf{X}_j^A\right)\|$. In this case, the coefficients c_{ij} can be written as

$$c_{ij} = \frac{e^{-\|\boldsymbol{w} \odot \left(\mathbf{X}_i^A - \mathbf{X}_j^A\right)\|/\lambda}}{\sum_{m(\neq i)} e^{-\|\boldsymbol{w} \odot \left(\mathbf{X}_i^A - \mathbf{X}_m^A\right)\|/\lambda}}, \tag{6.5}$$

and the derivatives of $DII\left(d^A(\boldsymbol{w}) \to d^B\right)$ with respect to the parameters w^α can be computed:

$$\frac{\partial}{\partial w^\alpha} DII\left(d^A(\boldsymbol{w}) \to d^B\right) = \frac{2\, w^\alpha}{\lambda\, N^2} \sum_{\substack{i,j \\ (i \neq j)}} c_{ij}\, r_{ij}^B \left(- \frac{(X_i^\alpha - X_j^\alpha)^2}{\|\boldsymbol{w} \odot \left(\mathbf{X}_i^A - \mathbf{X}_j^A\right)\|} + \sum_{m(\neq i)} c_{im} \frac{(X_i^\alpha - X_m^\alpha)^2}{\|\boldsymbol{w} \odot \left(\mathbf{X}_i^A - \mathbf{X}_m^A\right)\|} \right) . \tag{6.6}$$

These derivatives can be used in gradient-based methods to minimize the DII with respect to the variational weights.

If one aims at a low-dimensional representation of the feature space A, as in the case of feature selection, it is desirable that several of the weights are set to zero. While for up to $D_A \sim 10$ a full combinatorial search of all feature subsets can be carried out, optimizing the DII over each subset, for larger feature spaces a sparsification heuristic becomes necessary. We complement the DII optimization with two approaches for learning sparse features: Greedy backward selection and L_1 (lasso) regularization. Greedy selection removes one feature at a time from the full set, according to the lowest weight. L_1 regularization selects the subset of features that optimizes the DII while simultaneously keeping the L_1 norm of the weights small (see section 6.2.3). While greedy backward selection gives reliable results for up to ≈ 100 features, in larger feature spaces this algorithm becomes computationally demanding, and it is advisable to use L_1 regularization to find sparse solutions.

6.2.1 Adaptive softmax scaling factor λ

Qualitatively, the scaling factor λ in the softmax coefficient $c_{ij}(\lambda, \boldsymbol{w})$ defines the size of the neighborhoods in the input space $d^A(\boldsymbol{w})$ used for the rank estimation. Since λ is the same for every data point, regardless of whether the point is an outlier or within a dense cloud, this factor mainly decides how many neighbors are included in dense regions of the data manifold. Importantly, choosing λ too small makes the optimization less efficient, as in the limit $\lambda \to 0$ the derivative of the DII (see Eq. (6.6)) can be shown to vanish for almost all values of the parameters $\boldsymbol{w}$.

To automatically set λ, we take the average of two distance variables, $\hat{d}^A_{\text{min}}$ and $\hat{d}^A_{\text{avg}}$, which heuristically define the "small distance" scale in space d^A. Both of these numbers are based on $\hat{d}^A_i$, here denoting the difference between 2nd and 1st nearest neighbor distances for each data point i, $\hat{d}^A_i = d^A_{ik} - d^A_{ij}$, where $r^A_{ij} = 1$ and $r^A_{ik} = 2$:

$$\hat{d}^A_{\text{min}} := \min_i \hat{d}^A_i \,, \tag{6.7a}$$

$$\hat{d}^A_{\text{avg}} := \frac{1}{N} \sum_i \hat{d}^A_i \,. \tag{6.7b}$$

Setting λ to the average of $\hat{d}^A_{\text{min}}$ and $\hat{d}^A_{\text{avg}}$ at each step of the DII optimization has proven to enhance both the speed and stability of convergence. Indeed, using differences between nearest neighbor points to determine λ is more robust than using nearest neighbor distances directly, as in high dimensions first-, second- and higher-order neighbor distances tend to be very similar on a relative scale [119, 120].

6.2.2 Invariance property of the DII

In the limit $\lambda \to 0$, the DII defined in Eq. (6.2) is invariant under any global scaling of the distances in space A, $d^A_{ij} \mapsto |c|\, d^A_{ij}$ with $c \in \mathbb{R}$. Similarly, in the small λ regime, $DII(d^A(\boldsymbol{w}) \to d^B)$ is invariant under any uniform scaling of the weight vector, $\boldsymbol{w} \mapsto c\,\boldsymbol{w}$, if $d^A(\boldsymbol{w})$ is built as the usual Euclidean distance in the scaled feature space. This property can be easily verified by observing that the softmax coefficients c_{ij} can be replaced by $\delta_{1,r^A_{ij}}$ when $\lambda \to 0$, and the ranks r^A_{ij} are invariant under a global scaling of the distances d^A_{ij}. The same invariance holds even for $\lambda > 0$ if λ is chosen adaptively (see subsection 6.2.1), as in the adaptive scheme a global scaling of the distances d^A_{ij} implies a scaling of λ by the same factor, which leaves the c_{ij} coefficients untouched.

6.2.3 Optimization of the DII

The optimization of the DII is implemented in `FeatureWeighting.return_weights_optimize_dii` in DADApy by gradient descent utilizing the analytic derivative of the DII. The default value of the initial feature weights is the inverse standard deviation of each feature. Pseudocodes of the DII optimization algorithms are provided in the appendix in 'DII pseudocodes'.

Learning rate decay

We employ two different schemes of learning rate decay, (1) cosine learning rate decay and (2) exponential learning rate decay. When both schemes are used, we select the solution with lower DII among those found with the two schemes. In the first scheme, the learning rate is updated according to $\eta^k = 0.5\eta^0 \cdot (1 + \cos(\frac{\pi k}{n_{\text{epochs}}}))$, where k denotes the training epoch, η^0 the initial learning rate, and n_{epochs} the total number of epochs in the training. The exponential decay follows $\eta^k = \eta^0 \cdot 2^{\frac{-k}{10}}$. This schedule cuts the learning rate by half every 10 epochs. While the cosine decay leads to optimal results in the absence of L_1 regularization, or for weak regularization, the exponentially decaying learning rate is especially suited for high L_1 regularization [121]. In both schemes, "GD clipping" is used, as described hereafter in the section on L_1 regularization.

L_1 regularization

This method is implemented in DADApy in `FeatureWeighting.return_weights_optimize_dii` when a L_1 penalty different from 0 is chosen, and several different L_1 values are screened in `FeatureWeighting.return_lasso_optimization_dii_search`. Optimizing the DII with respect to the feature weights while simultaneously introducing sparsity, *i.e.* limiting the number of features used, can be considered a convex optimization problem of the form:

$$\min_{\boldsymbol{w}\in\mathbb{R}^D} \big(f(\boldsymbol{w}) + p\,\Omega(\boldsymbol{w})\big), \tag{6.8}$$

where $f : \mathbb{R}^D \to \mathbb{R}$ is a differentiable function such as $DII\left(d^A(\boldsymbol{w}) \to d^B\right)$, at least locally convex, and $\Omega : \mathbb{R}^D \to \mathbb{R}$ is a sparsity-inducing, non-smooth, and non-Euclidean norm with penalization strength p [122]. We use the L_1 norm, $\Omega(\boldsymbol{w}) = \sum_{\alpha=1}^{D} |w^\alpha|$ (also called lasso regularization):

$$\min_{\boldsymbol{w}\in\mathbb{R}^D} \left(DII + p\,\Omega(\boldsymbol{w})\right) = \min_{\boldsymbol{w}\in\mathbb{R}^D} \left(\frac{2}{N^2} \sum_{\substack{i,j=1 \\ (j\neq i)}}^{N} c_{ij}(\lambda, d^A(\boldsymbol{w}))\, r_{ij}^B + p \sum_{\alpha=1}^{D} |w^\alpha| \right) \tag{6.9}$$

The L_1 norm has the shortcoming that in $N \ll D$ setting, with very few samples but many dimensions, a maximum of N variables can be selected. The L_1 regularization tends to select just one variable from a group of correlated variables and ignore the others [123], which helps building optimal groups of maximally uncorrelated features, like also the Information Imbalance itself does (see Figure 4.4 in chapter 4).

Naive gradient descent with L_1 regularization usually does not produce sparse solutions, as a weight becomes zero only when it falls directly onto zero during the optimization [121]. This is very unlikely with most learning rate regimes. Instead, we employ the two-step weight updating approach [124], also known as "GD clipping" [121]:

$$\begin{aligned} w^\alpha_{t+\frac{1}{2}} &= w^\alpha_t - \frac{\partial DII\left(d^A(\boldsymbol{w}) \to d^B\right)}{\partial\, w^\alpha} \\ &\textbf{if } w^\alpha_{t+\frac{1}{2}} > 0 \textbf{ then } w^\alpha_{t+1} = \max(0, w^\alpha_{t+\frac{1}{2}} - \eta\, p) \\ &\textbf{if } w^\alpha_{t+\frac{1}{2}} < 0 \textbf{ then } w^\alpha_{t+1} = |\min(0, w^\alpha_{t+\frac{1}{2}} + \eta\, p)| \end{aligned} \tag{6.10}$$

Here, p denotes the L_1 penalty strength, and t is the epoch index. First, the update is performed only with the GD term, which may result in a change of sign for the weight. Subsequently, the L_1 term is applied, shrinking the weight magnitude. If this shrinkage would change the weight's sign, the weight is instead set to zero. Since the DII is sign invariant, all weights are kept positive during the optimization.

Backward greedy optimization

This approach is implemented in DADApy in `FeatureWeighting.return_backward_greedy_dii_elimination`, and the pseudocode in the appendix, algorithm 2. It starts with a standard optimization run using all the D_A features of the input space. From the solution of the first optimization, the feature corresponding to the smallest weight is discarded (set to zero), and a new optimization with $D_A - 1$ features is carried out. This procedure is iterated until the single most informative feature is left. The greedy backward approach is an alternative to the L_1 regularization and is applicable to moderately large data sets with $D_A \lesssim 100$ features and $N \lesssim 500$ data points, since the computational complexity scales linearly with the number of features.

6.2.4 A linear scaling estimator of the DII

The DII scales quadratically with the number of points N, with a computational complexity of $\mathcal{O}(N^2 \cdot D)$, where the main steps of the algorithm (computing the

DII and its gradient, Equation 6.2 and Equation 6.6) both involve a double sum over the rows (index i) and the columns (index j) of the $N \times N$ matrices in the equations. D is the number of features. We notice that the additional sums over index m (denominator of the c_{ij} coefficients, Equation 6.3, and second term in the gradient, Equation 6.6 do not depend on index j and can therefore be precomputed for each index i, avoiding three nested loops.

The computational time can be dramatically decreased by subsampling the rows of the matrices r_{ij}, d_{ij} and c_{ij} appearing in Eq. (6.2), reducing them to a rectangular shape $N_{\text{rows}} \times N$ (with $N_{\text{rows}} < N$). This subsampling is performed only once at the beginning of the training, so that the rectangular shape of such matrices is kept fixed during all the DII optimization. If the DII is written as the average of N conditional ranks,

$$DII\left(d^A(\boldsymbol{w}) \to d^B\right) = \frac{2}{N}\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{\substack{j=1\\(j\neq i)}}^{N} c_{ij}(\lambda, d^A(\boldsymbol{w}))\, r_{ij}^B\right) = \frac{2}{N}\langle r^B | r^A \approx 1\rangle\,, \tag{6.11}$$

the subsampling is equivalent to replacing $1/N\sum_{i=1}^{N}$ with $1/N_{\text{rows}}\sum_{i=1}^{N_{\text{rows}}}$. This means computing the average of N_{rows} conditional ranks instead of N. Different schemes to set N_{rows} result in different scaling laws of the algorithm with respect to N. Setting N_{rows} to a fraction of N (green curve in Fig. 6.3A, $N_{\text{rows}} = N/2$) brings to a quadratic scaling with a smaller prefactor, while sampling a fixed number of points N_{rows} independently of N (red curve, $N_{\text{rows}} = 100$) brings to a linear scaling $\mathcal{O}(N \cdot D)$. In the latter case we observe a striking reduction of the runtime, while the accuracy of the recovered weights is almost perfectly preserved (Fig. 6.3B).

6.3 Methods

We here provide details on the two datasets used in this chapter: a molecular dynamics of a small peptide, and a set of configurations of liquid water used for training a neural network potential.

Extraction of collective variables from the CLN025 MD

All collective variables used in subsection 6.4.2 were extracted from the MD simulation using PLUMED 2 [125]. The ground truth pairwise heavy atom distances were computed using the "DISTANCE" CV on all pairs of non-hydrogen atoms. The radius of gyration was obtained with the "GYRATION" CV and the C_α atoms. The number of hydrophobic contacts were calculated using the "COORDINATION" CV (R_0=0.45) and using the amino acids THR, TRP, and TYR of

CLN025, and sidechain carbons not directly bonded with an electronegative atom. The number of hydrogen bonds was also calculated using the "COORDINATION" CV (R_0=0.25). For backbone H-bonds and sidechain H-bonds only hydrogens and oxygens of the backbone and the sidechain were considered, respectively, while for the sidechain-to-backbone interactions, the cross of these were considered. For the quantification of the alpha-helical content and the anti-parallel beta sheet content, the CVs "ALPHARMSD" and "ANTIBETARMSD" were used with all residues of the peptide. For the principle components PC1, PC2 and the PCA residual, first a pdb file containing the average structure of the trajectory and the two first principle directions was created using the CVs "COLLECT_FRAMES_ATOMS" with all heavy atoms, and "PCA" using the previous output and optimal alignment. Subsequently, each frame of the trajectory was projected onto the two principle components referenced in the pdb file using "PCAVARS".

Block cross validation of CLN025

To account for the equilibration of the system, the first $\sim$ 15 ns of the trajectory were discarded throughout the analysis (1,580 of 41,580 trajectory frames). Block cross validation (Fig. 6.4A) was carried out by splitting the remaining frames into 4 consecutive blocks. The training blocks were built by subsampling each block to every 7th frame to de-correlate, leaving 1428 points per training block. The optimal tuple and weight results from each block were used to calculate the DII in 21 test sets built from the remaining three blocks (repeatedly subsampling each block with stride 7, starting from frames 1 to 7).

ACSF and SOAP descriptors

The systems for creating ACSF and SOAP descriptors are based on 1593 liquid H_2O structures whose forces and energies were found using DFT via the CP2K [126] package with the revPBE0-D3 functional. We use the DScribe Python package [127, 128] to calculate SOAP and ACSF descriptors from the atomic positions. The data points were chosen as follows: The 1593 structures (with 64 H_2O molecules each) yielded 192,000 atomic environments, from which a subset of $\sim$350 was sampled to reduce the computational time of feature selection. The ACSF descriptors were constructed on a grid of hyperparameters (G2: $\eta \in [10^{-3}, 10^{0.5}]$ logspace $n_\eta = 15$, $R_S = 0$, G4: $\eta \in [10^{-3}, 10^{0.5}]$ logspace $n_\eta = 6$, $\zeta \in \{1, 4\}$, $\lambda \in [-1, 1]$ linspace $n_\lambda = 4$, $R_S = 0$), resulting in 176 (+2 cutoff functions) different features for each atomic environment. The 546 SOAP descriptors were selected with $n_{max} = 6$, $l_{max} = 6$ and a cutoff radius of 6Å.

The optimization of ACSF with respect to the ground truth of SOAP is carried out starting from $\gamma_i = 1 \; \forall \, i \in [1, 176]$.

6.4 Applications and Results

6.4.1 Benchmarking the approach: Gaussian random variables and their monomials

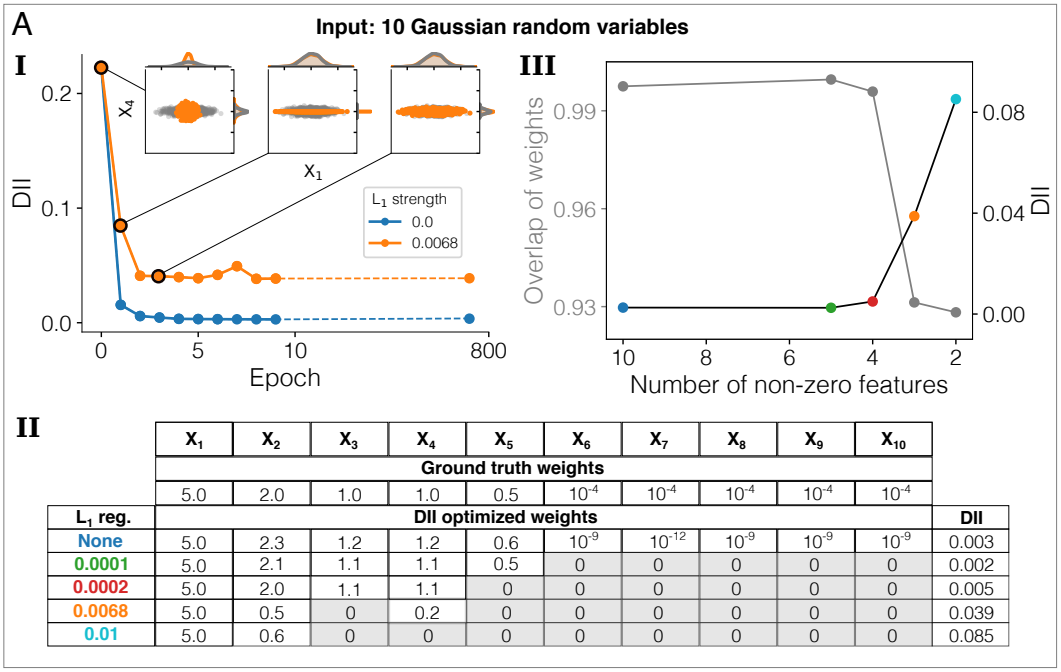
We first test the DII approach using two illustrative examples where the distances $d^A(\boldsymbol{w})$ and d^B are built with the same features, so that the target weights minimizing Eq. (6.2) are known. In particular, we take as ground truth distance d^B the Euclidean distance in the space of the scaled data points $\boldsymbol{w}_{GT} \odot \mathbf{X}_i$, where the weights $\boldsymbol{w}_{GT}$ are fixed and known. We aim at recovering the target weights by scaling the unscaled input features, $\boldsymbol{w} \odot \mathbf{X}_i$, with the proposed DII-minimization.

In each example, we carry out several optimizations, both without regularization and in presence of a L_1 penalty, which induces sparsity in the learned weights. For each optimization, we employ a standard gradient descent algorithm, initializing the parameters $\boldsymbol{w}$ with the inverse of the features' standard deviations (see subsection 6.2.3 for details). In order to judge the quality of the recovered weights in the various settings, we calculate the cosine similarity between the vector of the optimized weights and $\boldsymbol{w}_{GT}$. This evaluation metric, which is bounded between 0 (minimum overlap) and 1 (maximum overlap), only depends on the relative angle between the two vectors, reflecting the fact that the DII recovers the target weights up to a uniform scaling factor (see 'Invariance property of the DII').

10 Gaussian random variables

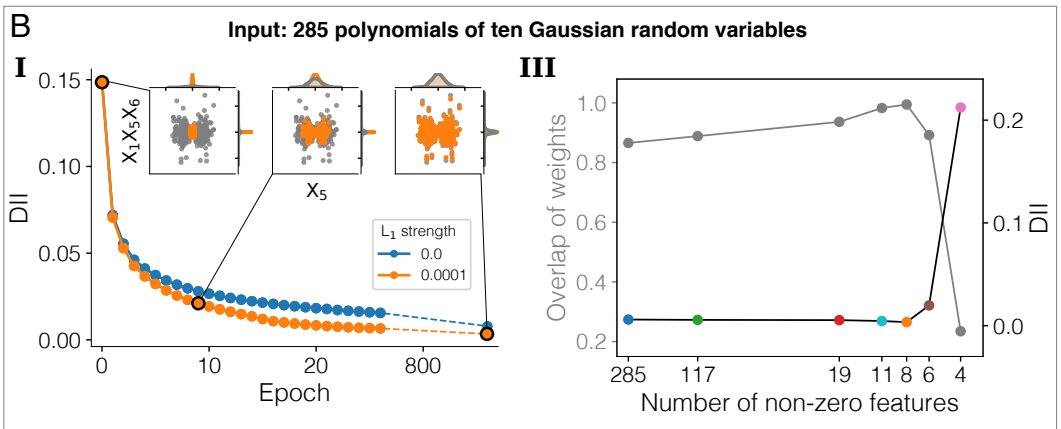
In the first example, we use a data set of 1500 points drawn from a 10-dimensional Gaussian with unit variance in each dimension, and we construct a ground truth distance d^B by assigning non-zero weights w^{α}_{GT} to all its 10 components (Table II in Fig. 6.1A). The target weights w^{6}_{GT} to w^{10}_{GT} are close to zero, such that these features carry almost no information.

The optimization without any L_1 regularization yields a very good result in terms of DII and overlap (blue in Fig. 6.1A I, II, and III). If a soft L_1 regularization strength is employed, the results are qualitatively the same, but the irrelevant features α = 6-10 receive zero weights, leading to an effective feature selection (green in Fig. 6.1A II and III). Table II in Fig. 6.1A shows the learned weights for different strengths of the L_1 penalty, scaled in such a way that the largest weight is identical to the largest component of $\boldsymbol{w}_{GT}$. Since in DII only the relative weights are important, this is permissible and helps illustrate. By increasing the regularization strength, more features are set to zero following the order of their ground truth weights. When features of higher importance, namely with higher ground truth weights, are forced to zero, then the resulting DII increases and the cosine similarity decreases, showcasing the loss of information (Fig. 6.1A III).



II

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	
	Ground truth weights										
	5.0	2.0	1.0	1.0	0.5	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	
L_1 reg.	DII optimized weights										DII
None	5.0	2.3	1.2	1.2	0.6	10^{-9}	10^{-12}	10^{-9}	10^{-9}	10^{-9}	0.003
0.0001	5.0	2.1	1.1	1.1	0.5	0	0	0	0	0	0.002
0.0002	5.0	2.0	1.1	1.1	0	0	0	0	0	0	0.005
0.0068	5.0	0.5	0	0.2	0	0	0	0	0	0	0.039
0.01	5.0	0.6	0	0	0	0	0	0	0	0	0.085



II

	X_5	$X_1X_5X_6$	X_3	X_2^2	X_6	X_{10}	X_1X_2	$X_8X_{10}^2$	X_8	X_5X_8	Sum of other weights	
	Ground truth weights											
	10.0	7.0	6.0	5.0	5.0	4.0	3.0	2.0	1.0	1.0	0	
L_1 reg.	DII optimized weights											DII
None	10.0	6.2	6.9	2.9	6.1	5.1	2.2	2.5	0.8	0.7	72.4	0.006
$4 \cdot 10^{-6}$	10.0	6.2	6.9	2.9	6.1	5.1	2.2	2.4	0.8	0	53.0	0.006
$5 \cdot 10^{-4}$	10.0	4.5	6.0	1.8	5.1	4.0	1.6	1.6	0	0	8.9	0.005
$5 \cdot 10^{-5}$	10.0	6.2	6.5	3.7	5.6	4.5	3.1	2.1	0.7	0	2.6	0.005
0.0001	10.0	6.3	6.3	5.0	5.4	4.3	3.0	2.1	0	0	0	0.003
0.0014	10.0	6.9	6.2	0	5.7	3.1	0	0	0	0	3.8	0.020
0.0038	2.3	1.1	1.4	0	0	0	0	0	0	0	10	0.212
0.0023	10.0	0	0	0	0	0	0	0	0	0	0	0.606

Figure 6.1: **A**: The input features are 10 i.i.d. Gaussian random variables, X_1-X_{10}. The same features are used as ground truth, but scaled. **I**: *DII* as a function of the number of non-zero weights, optimized with (orange) and without (blue) L_1 regularization. The insets show two example features, with ground truth weights (gray) and weights during optimization (orange). **II**: Ground truth and final optimized weights at selected L_1 strengths. **III**: Cosine similarity (overlap) of the ground truth and optimized weights in gray, *DII*s in black with colored markers, for several L_1 strengths and associated numbers of non-zero features. **B**: The feature space consists of the 285 monomials up to order three of the ten Gaussian variables from **A**. As ground truth, ten features were selected at random and scaled, while all the other feature weights are zero. **I**, **II**, **III**: Analogous to **A**.

285 monomials

Secondly, to test the method in a high-dimensional setting, we created a data set with 285 features including all the products up to order three of the 10 Gaussian random variables used in the previous example. Products of Gaussian random variables are distributed according to Meijer G-functions, which may not be Gaussian [129]. The ground truth distance d^B is here built by only selecting ten of these monomials, with various weights (Table II in Fig. 6.1B). All other feature weights in the ground truth can be considered zero.

Since in this case the correct solution is very sparse in the full feature space, an appropriate sparsity-inducing regularization becomes essential to obtain good results. Without any L_1 regularization, all the 285 features receive a non-zero weight. Even if, in this case, the ground truth features are assigned the highest weights, there might not be a clear cut-off in the weight spectrum to distinguish them from the less-informative features.

As shown in Fig. 6.1B, the correct level of regularization can be identified by computing the DII as a function of the non-zero features or regularization strength. Several different L_1 strengths lead to the same number of non-zero features with different features and/or weights. In these cases, the lowest DIIs per numbers of non-zero features should be selected, as in Fig. A.2 (appendix). The intermediate L_1 strength of 0.0001 results in the best performance, as it coincides with the lowest DII and the largest weight overlap (orange in Fig. 6.1B I, II, and III). The eight most relevant ground truth features are correctly identified, with an overlap between the learned and the ground truth weights which is remarkably close to 1.

Furthermore, panel I in Fig. 6.1B shows that weights found with L_1 regularization have a lower DII than the ones without L_1 regularization in the same optimization time, which means that the weights resulting from a certain level of regularization are effectively better than the unregulated ones. As in the previous example, when the regularization is too strong, some of the relevant features are discarded, resulting in a drop in the weight overlap and an increase in the DII.

Comparison with other methods

We then benchmarked the DII method against other feature selection methods. We perform the benchmark on the example with 285 monomials, in which the ground truth is known.

There are very few methods available in software packages which can be applied to the specific task we are considering, which is selecting and scaling features from a high-dimensional input space to be maximally informative about a multi-dimensional continuous ground truth, defining a pairwise distance. Considering

filter methods, we compare *DII* to **relief-based algorithms (RBAs)**, specifically RReliefF and MultiSURF, implemented in scikit-rebate [130], which support a continuous ground truth [62]. The RBAs stand out among filter methods because they can assign feature weights. They output weights between -1 (most irrelevant) and 1 (most relevant) for each feature, but importantly only work with one-dimensional ground truth. This poses a problem for all use cases in this paper because the ground truth is always defined by the multi-dimensional vector of features used to compute the target distance. RBAs extended to the multi-label case [131, 75] but, to our knowledge, are not implemented in software packages. We applied the algorithms vs. each feature of the ground truth separately and (*a*) summed all resulting weights that scale the input features (orange and red in Fig. 6.2) or (*b*) set all resulting weights except the largest to zero and summed these sparse vectors (green and purple in Fig. 6.2). Then we calculated the cosine similarity (overlap) with the 285-dimensional ground truth vector (all weights zero except the ten relevant weights, which are set to their value). The methods detect the most important input feature in most cases, leading to overall cosine similarities ranging from 0.56 to 0.84 for the various settings (Fig. 6.2).

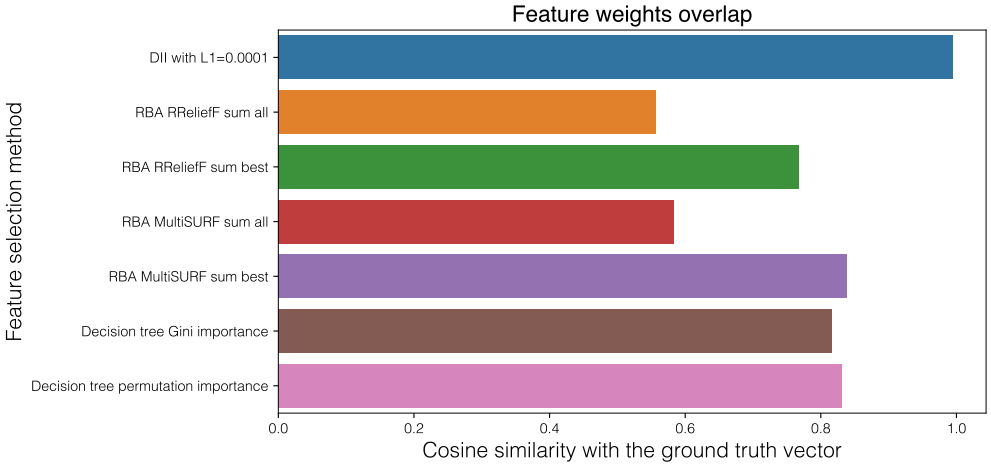


Figure 6.2: The overlap of the weight vector resulting from different feature selection and weighting methods with the ground truth weight vector, calculated as the cosine similarity. For the Relief-based algorithms (RBAs) RReliefF and RBA MultiSURF, "sum all" refers to the sum over the ten individual optimizations (for each ground truth feature as univariate label) of the resulting weight vectors, while "sum best" means the same sum over individual optimizations, but setting all weights to zero except the largest one in each feature vector. The "decision tree" results refer to the Gini and Permutation importance selected feature weights as provided by the decision tree regression aimed at the full ten-dimensional target space.

As a second benchmark we use a method from scikit-learn [76], which can handle the task's requirements: The **decision tree regressor** (`sklearn.tree.DecisionTreeRegressor`). Unlike DII and the relief-based algorithms, this method is not a filter but an embedded method. The feature selection is determined as a side product during the building of a regressor model. There is no filter algorithm implemented in scikit-learn which can solve a problem as posed here. Tuning the algorithm for various error criteria and splitters, the defaults (`criterion='squared_error'`, `splitter='best'`) performed best. The feature importances were derived with two metrics: Gini importance and Permutation importance for for feature evaluation [46]. The latter is expected to be more robust for features with many unique values, like in the test example here. They lead to feature vectors with a cosine similarity of up to 0.83 with respect to the ground truth. In comparison, the DII method with a L_1regularization of 0.0001 (orange in Fig. 6.1B) finds a weight vector with eight non-zero weights and a cosine similarity of 0.99.

Scalability test

We test the scalability of our algorithm with respect to the number of points N used to perform the minimization of the DII on the example of 285 monomials (Fig. 6.1B). We construct the ground-truth distance d_B by multiplying 5 features with non-zero weights. In Fig. 6.3A we show the runtime for a single optimization of the DII as a function of N (ranging from 100 to 10000). As a quality validation measure, we report in Fig. 6.3B the overlap (cosine similarity) between the learned and the ground-truth weights. The tests have been performed using the JAX implementation of the algorithm on a GPU nVidia TU104GL [Quadro RTX 5000].

The standard algorithm ($N_{\text{rows}} = N$, blue line) scales quadratically with the number of points N. The computational time can be dramatically decreased by performing the sum over i only on a fixed subset of points, as explained in subsection 6.2.4. This leads to a striking reduction of the runtime, and in the best case to a linear scaling $\mathcal{O}(N \cdot D)$ in the number of data points. Notably, the accuracy of the recovered weights is almost preserved (Fig. 6.3B).

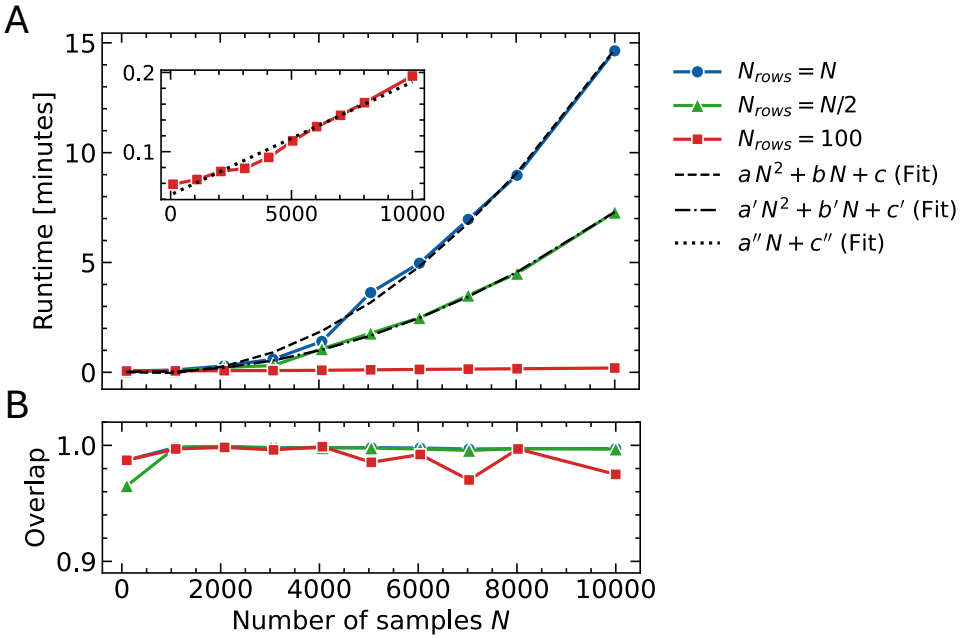


Figure 6.3: **A**: Runtime of the DII optimization as a function of the data set size N, using the 285 monomial features in the input space and 5 non-zero features in the ground-truth space. Each optimization is carried out with 1000 epochs, setting the L_1 penalty to 10^{-3}. Different colors show how the runtime is affected by different row subsampling schemes: No subsampling (blue), linear subsampling in the number of points (green), constant N_{rows} (red). The dashed and dotted black lines show least square fits with the expected scaling laws. **B**: Overlap between learned weights and ground truth weights, computed as the cosine similarity between the two weight vectors.

In conclusion, in all test examples the DII method is able to recover the ground truth weights with good accuracy, and better than the very few other applicable methods, as measured by a larger weight overlap with the ground truth. In the following sections, we apply our feature selection method to cases in which the optimal solution is not known and illustrate how our approach can be used to give an explicit system description by extracting few features from a larger data set.

6.4.2 Identifying the optimal collective variables for describing a free energy landscape of a small peptide

We now illustrate how the DII can be used to identify the most informative collective variables (CVs) to describe the free energy landscape of a biomolecule. As opposed to the previous example, in this test the ground truth variables and the input variables are different sets.

We consider a temperature replica-exchange MD simulation (400 ns, 340 K replica analyzed only, $dt = 2$ fs) [132] of the 10-residue peptide CLN025 [133], which folds into a β-hairpin. The data set is composed of 1429 frames (subsampled from 41,580 trajectory frames) containing all atom coordinates. The ground truth metric d^B is constructed in the feature space of all the 4,278 pairwise distances between the 93 heavy atoms of the peptide, which can be assumed to hold the full conformational information of the system. We consider a feature space A with ten classical CVs that do not depend on knowledge of the folded state of the β-hairpin peptide: Radius of gyration (RGYR), anti-β-sheet content, α-helical content, number of hydrophobic contacts, principal component 1 (PC1), principal component 2 (PC2), principal component residuals, the number of hydrogen bonds in the backbone, in the side chains, and between the backbone and side chains (see section 6.3 for details).

Since the CV feature space is only 10-dimensional, it is possible to look for the optimal distance d^A by an exhaustive search of all possible 1023 subsets containing one to ten CVs, without using the L_1 regularization to produce sparse solutions. For each subset of CVs, the DII is used as a loss to automatically optimize the scaling weights, which are initialized to the inverse standard deviations of the corresponding variables. Even when all feature subsets can be constructed, gradient descent optimization of the DII is useful, as the most naive choices of the scaling weights - setting them to the inverse standard deviations of the variables, or all equal to 1 - likely define suboptimal distances, since the CVs have different units of measure and importance. The optimization of the feature weights for all 1023 subsets takes about 4.5 h on a CentOS Linux 7 with 24 CPUs Intel Xeon E5-2690 (2.60GHz) with 15 GB RAM using the function "return_weights_optimize_dii" with 80 epochs (Fig. 6.4A, green curve).

Fig. 6.4A shows the results of the subset optimizations by computing the DII with block cross-validation (see section 6.3). The training and validation DIIs averaged over all cross-validation splits show a high degree of consistency, verifying the transferability of the DII results between non-overlapping pieces of the trajectory. As shown in the inset graph in Fig. 6.4A, the DII result improves during the gradient descent optimization. The best single CV is anti-β-sheet content [134], while the best triplet contains RGYR, PC1 and PC2 with weights of 1.0, 3.5 and 4.7. Remarkably, the weight of PC2 is higher than the weight of PC1,

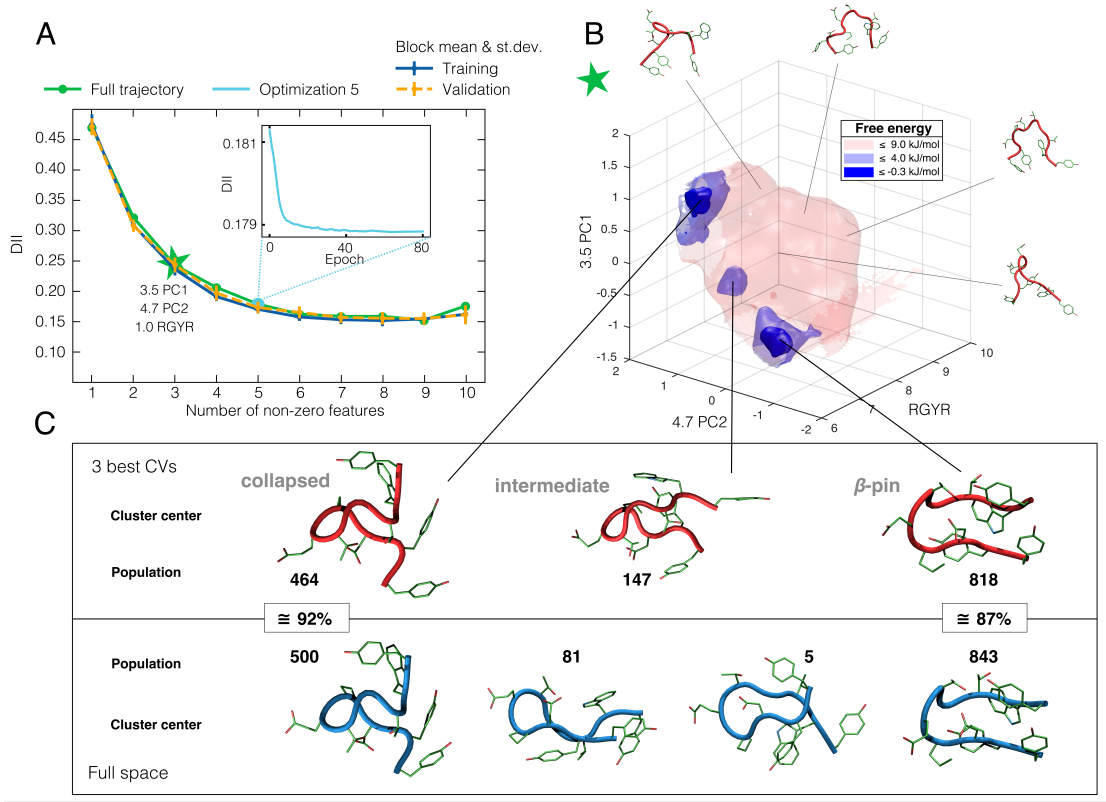


Figure 6.4: **A**: Green: Optimal DII results for CV subsets of different sizes, using their gradient descent optimized weights. Inset: DII gradient descent optimization for the optimal 5-plet. Blue and orange: Average and standard deviations of the DII calculated from all the training-validation splits. **B**: Free energy isosurfaces in the space of the optimal 3-plet of CVs (RGYR, PC1 and PC2, with weights of 1.0, 3.5 and 4.7), corresponding to three different values of the free energy. The renderings around the free energy surfaces are sampled at different values of the CVs and free energy. **C**: Red and blue renderings are cluster centers obtained from the optimal 3-plet space and from the full space of all pairwise heavy atom distances, respectively. The two main cluster centers of both belong to the dominant peptide conformations: The β-pin and the collapsed denatured state. The collapsed and β-pin clusters identified in the optimal 3-plet space share 92% and 87% of the frames with the corresponding full space clusters.

confirming that the gradient optimization of the DII provides non-trivial results. We estimated the density in the space of the best three scaled variables (Fig. 6.4B) using point-adaptive k-NN (PAk) [135], implemented in the DADApy package [3]. The free energy derived from this density clearly shows two favorable main states,

which are the folded β-hairpin state and a denatured collapsed state [136] with negative values of the free energy in Fig. 6.4B.

The cluster centers found by Density Peak Clustering in its unsupervised extension [137] are depicted by the renderings denoted "collapsed", "intermediate", and "β-pin" in Fig. 6.4C, while additional example structures from less favorable free energy regions are shown around Fig. 6.4B. The clustering was also performed in the full space of all 4,278 heavy atom distances, which holds the full information of the system.

The populations of both, β-pin and collapsed clusters show a remarkable overlap between the clustering structures obtained in the optimal 3-plet case and from the full feature space of 4,278 heavy atom distances. Taking the cluster populations from the full space as ground truth classes, such overlap can be simply measured as the fraction of points (trajectory frames) that belong to the same cluster in both representations, also referred to as cluster purity [138]: The β-hairpin cluster from the 3-plet space has 87% purity, and the collapsed state cluster has 92% purity, considering the full space as reference. Taken together, all clusters have a 89% overall cluster purity towards the full space clusters. This consistency also emerges by visually comparing the red and blue renderings of the two dominant cluster centers (left and right structures in Fig. 6.4C). As a comparison, running the clustering algorithm using the single best CV, the anti-β-sheet content, brings to an overall cluster purity of 45%, *i.e.*, the trajectory frames clustered into the pin, collapsed, or other clusters using the single best variable, capture 45% of the same frames of the according clusters using the full space for clustering. Hence, no single one-dimensional CV is informative enough to describe CLN025 well, but a combination of only three scaled CVs carries enough information to achieve an accurate description of this system.

Benchmarking the results against decision tree regression

Because of the good performance of decision tree regression on the previous example and its ability to handle multi-target (even high-dimensional), continuous ground truth data, we apply this feature selection algorithm also to this use case. The results are illustrated in Fig. 6.5. The best three variables using the Gini importance weights are: 0.29 anti-β-sheet content, 0.25 PC1, 0.1 PC2; using the permutation importance they are: 1.27 PC1, 1.04 anti-β-sheet content, 0.97 PC2.

Clustering in these reduced spaces leads to maximum cluster purities compared to the full space clusters of 55% for Gini importance and 63% for the permutation importance and several additional inconsistencies when compared to the full space clustering: The collapsed loop cluster is bigger than the native β-pin cluster, which contradicts the results obtained by full space clustering, where the β-pin cluster is the largest (with most frames) with the lowest free energy. Visual comparison

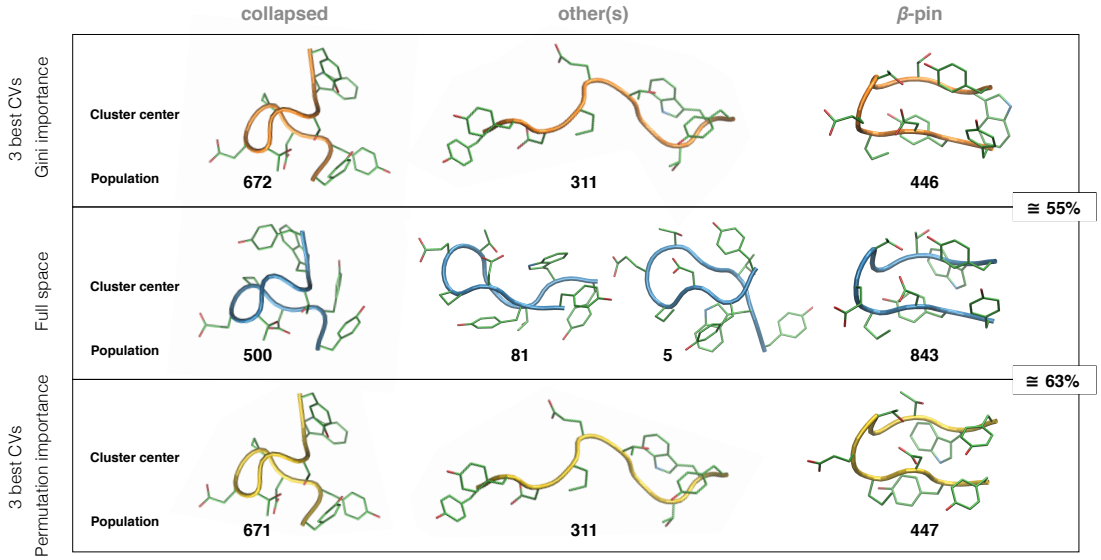


Figure 6.5: The Density Peak Clustering cluster centers and populations of CLN025 are shown derived from three different sets of collective variables: blue is the clustering done in the full space of 4,378 variables, orange in the space of the three best variables as selected and weighted by the Gini importance of the Decision Tree Regression model, and yellow as selected and weighted by the Permutation importance. The numbers on the right are the cluster purities with respect to the full space, measured as the fraction of points (trajectory frames) that belong to the same cluster in both representations.

confirms that the cluster centers of the two metastable states, the β-hairpin and the collapsed loop, derived from the decision tree regression feature selection, are less similar to the full space cluster centers than the DII-derived CV space in Fig. 6.4. In particular, even if the backbone is in a similar conformation, the side chains are arranged in a different manner.

Robustness test

We also test the robustness of the method using four uncorrelated trajectory blocks and performing the DII-optimization in each of these blocks. This was done by cutting the MD trajectory of the peptide CLN025 into 4 blocks of 10000 frames each and subsampling every 7th frame, to create 4 uncorrelated data sets of 1429 points each - the same size of data set as in the main analyses (subsection 6.4.2). For each of these sets, we found the optimal number of non-zero features and optimal relative weights of these features with DII-optimization, using the full space consisting of 4,378 heavy atom distances as target. Fig. 6.6 shows that the

DII decreases steadily until approximately 5 to 6 features, when the information content reaches its maximum. More than seven or eight features increase the DII, since noise seems to be added but no more independent information.

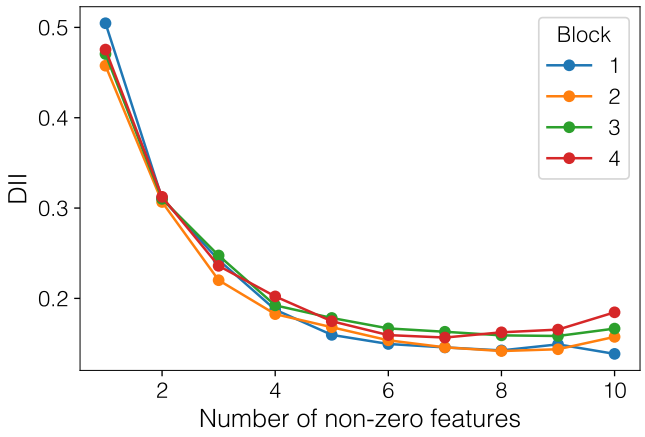


Figure 6.6: The DII for each of four uncorrelated trajectory blocks of CLN025 with 1429 frames each, *vs.* the number of selected (non-zero) features of a total of ten input features.

The input variables on which we performed the feature selection were the same collective variables as previously ((subsection 6.4.2)). Fig. A.3 in the appendix shows the normalized (to the unit vector) relative weights for the selected number of non-zero features and each block, corresponding to Fig. 6.6.

The resulting DIIs, as well as selected features and their weights, show excellent consistency across the blocks. (Fig. 6.6 and appendix: Robustness test of DII selection features from peptide CLN025 MD trajectory), meaning that they hold similar information for the same number of non-zero features.

6.4.3 Feature selection for Machine Learning Potentials

In another use case of the DII approach, we demonstrate its capabilities for selecting features for training Behler-Parrinello machine learning potentials (MLPs) [116]. MLPs can learn energy and forces of atomic configurations derived from quantum mechanical calculations. The Behler-Parrinello MLP uses Atom Centered Symmetry Functions (ACSFs) as inputs for the predictions [139]. The ACSFs are a large set of radial and angular distribution functions, which describe the environment around an atom, and are permutationally, rotationally, and translationally invariant.

The data set used here consists of $N \sim 350$ atomic environments of liquid water molecules, derived from a larger data set that has previously been used to fit a MLP, which can accurately predict various physical properties of water [140]. The input features in this example are 176 ACSF descriptors (see section 6.3). The ACSF descriptor dimensions combinatorially grow with the number of atom types, which makes them computationally costly and makes feature selection attractive [141]. Since the ACSF space is too large for full combinatoric feature selection, we search for sparse solutions using both, L_1 regularized DII and greedy backward selection ("L_1 reg." and "greedy" in Fig. 6.7, see section 6.2.3). We aim to select informative ACSFs before the training to reduce the number of input features and thus reduce the training and prediction time.

Here we can use another descriptor as ground truth: In the case of atomic environments, one of the most complete, accurate, and robust descriptions is given by the Smooth Overlap of Atomic Orbitals (SOAP) descriptors [117, 118], based on the expansion of the local density in spherical harmonics. 546 SOAP features ($n_{max} = 6$, $l_{max} = 6$) are defined as the ground truth for feature selection. In this manner, we can put SOAP and ACSF, two comprehensive representations of atomic environments, into relation [144] and show that SOAP is a suitable ground truth to select informative ACSFs as inputs for a MLP. The SOAP space captures the full spacial arrangement of atoms by encoding the local atomic densities and accounting for symmetries [145]. Both SOAP and ACSF descriptor spaces, as well as further local atomic density descriptors, such as the atomic cluster expansion (ACE) representation, have been shown to be compressible without significant loss of information, improving computational efficiency [112, 146].

The resulting DII for various numbers of ACSFs can be seen in Fig. 6.7A. With both greedy and L_1 regularized selection, we find that the optimized DII asymptotically approaches an optimal value with growing number of non-zero features. However, even relatively small feature spaces with $\sim$ 10-30 non-zero features have low DII values, making effective feature selection possible. We validate the selected features and their weights on validation sets of atomic environments of equal size as the training set. The resulting DIIs are slightly higher but mostly

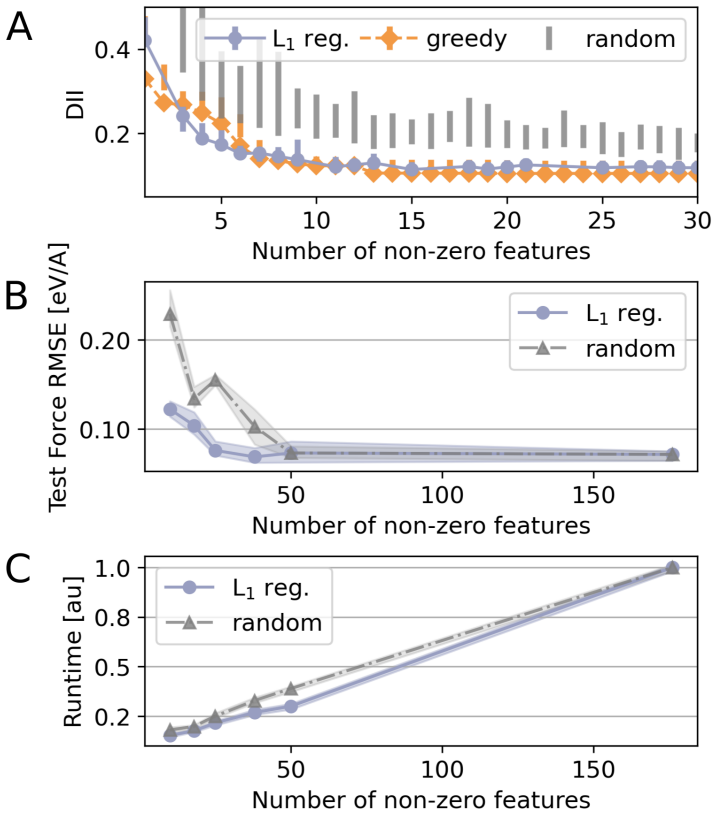


Figure 6.7: Selection of the optimal feature subsets from $D_A = 176$ ACSF descriptors, against a ground truth of $D_B = 546$ SOAP descriptors, using a data set of $N \sim 350$ atomic environments. **A**: The optimized DII per number of non-zero features is shown by blue circles and orange diamonds, using L_1 regularized search and greedy backward selection, respectively. The bars represent validation results, using as data points atomic environments other than the ∼350 environments used for DII feature selection. The gray bars depict the range between the lowest and highest DII results for 10 random selections of the specified number of non-zero features. **B**: Test root-mean-square error (RMSE) with features chosen via L_1 regularized DII (blue circles) and at random (gray triangles) by Behler-Parrinello-type MLPs [116] as implemented in n2p2 [142, 143]. Markers represent the average RMSE of six MLPs with different train-test splits per number of non-zero features, the filled area shows the range from worst to best performer. **C**: Run-time of force and energy prediction on a single structure performed by the same MLPs as in **B**. The filled area shows the range from worst to best performer, despite being barely visible due to similar run-times across the six MLPs.

comparable to the training DIIs, showcasing the robustness and transferability of the results. As a sanity check for our selection, we also show that randomly selected feature sets have a significantly higher DII than optimized sets, meaning they are less informative about the ground truth space (Fig. 6.7A gray).

To show that the features selected by DII are indeed physically relevant, we report in Fig. 6.7B the root-mean-square error (RMSE) of atomic forces for Behler-Parrinello MLPs using ACSF subsets of different sizes ($n_{ACSF} \in \{10, 18, 25, 38, 50, 176\}$). We find that MLPs with features selected by L_1 regularized DII optimization outperform random input features for all tested numbers of input features n_{ACSF}. The difference in prediction accuracy is most pronounced at small n_{ACSF}, where it is least likely that random selection chooses meaningful features. After $n_{ACSF} \approx 20$ input features, the optimized subsets reach an accuracy of < 100 meV, which is on par with the original MLP trained on these data [140]. Compressions of local atomic density representations for machine learning potentials have also previously been shown to require a minimum set size of 10-20 PCA features, since further compression fails to faithfully preserve the geometric relationships between data points and leads to increased prediction errors [147]. With $n_{ACSF} = 50$ input features, the MLP performs roughly equally well to using the full data set, while having less than half the run-time (Fig.6.7C). This shows that DII can be used to select features for downstream tasks such as energy and force fitting in MLPs, by optimizing for a complex ground truth and finding a space with fewer but optimally weighted features that contain the same information.

6.5 Discussion

This chapter presents the Differentiable Information Imbalance, DII, designed to automatically learn the optimal distance metric d^A over a set of input features. The metric reproduces the neighborhoods of the data points as faithfully as possible according to a ground truth distance d^B. Here d^A is defined as the Euclidean distance, and the optimization parameters are weights that scale individual features, such that the presented DII is an automatic and universal feature selection and weighting algorithm.

While many other methods are restricted to single variable outputs as "labels" or "targets", DII can handle any dimensionality of input and output. Continuous and discrete data is supported and the method can be used in a supervised and unsupervised manner. The weights are optimized automatically, and by using the values of the DII as a quality measure one can compare the information content of several feature sets, and select the sets corresponding to the lowest DII for each number of features (see appendix: Choosing tuples by DII). It is one of very few filter methods that account for feature dependencies but do not rely on explicit feature subset evaluation [62].

In illustrative examples where the optimal feature weights are known, we showed that the DII can reliably find the correctly weighted ground truth features out of high-dimensional input spaces. The behavior of the DII as a function of the subset size appears to be anti-correlated with the cosine similarity between ground truth and optimized weights. This implies that the DII value can be used for assessing the quality of the selected feature subsets when the actual ground truth weights are unknown. The weighted feature sets as provided by DII optimization have a higher cosine similarity to the ground truth than sets derived from two other feature selection classes, relief-based algorithms (RBAs) [130] and decision tree regressions.

We further applied the method to analyze a molecular dynamics (MD) simulation of a biomolecular system. Extracting a small subset of informative collective variables (CVs) from a pool of many candidate CVs from a MD trajectory is a general problem with both practical and conceptual benefits, including using such CVs in enhanced sampling techniques and obtaining an interpretable description of the free energy landscape. For the peptide CLN025, the selected CVs are the first two principle components (3.5 PC1, 4.7 PC2) and the radius of gyration (1.0 RGYR). Applying clustering in the space of these three scaled CVs leads to the correct identification of the β-pin state and collapsed denatured state of CLN025, in accordance with the clusters built from a much larger feature space, which includes all heavy atom distances. The reduced space clusters are highly meaningful with a 89% overall cluster purity towards the extended space clusters, while reduced variable spaces built from clustering results of the decision tree regression

lead to lower cluster purities. Tests of uncorrelated parts of the MD trajectory show great consistency of the results, highlighting the robustness of the method.

In a second application, our method successfully selects highly informative subsets of input features for training a Behler-Parrinello machine learning potential that achieves optimal performance in terms of the mean absolute error of force and energy. We find that using just 50 informative ACSF descriptors selected by our approach, instead of 176, significantly reduces the MLP's computational cost, cutting the runtime by one third while maintaining nearly the same accuracy.

The DII is not necessarily a simple monotonic function of the number of non-zero features post-optimization (cardinality). In some cases, the selection of additional features can introduce noise or redundancies that can negatively impact the description of the ground-truth space. Furthermore, if the optimal non-zero features for a two-dimensional description are, say, X3 and X61, the optimal features for a three dimensional description could be completely different, say X5, X9 and X44. The DII is hence also not necessarily a submodular function of the number of features.

To extract small subsets of features from high dimensional input data, we implemented two different sparsity inducing heuristics: L_1 regularization and greedy backward optimization. Greedy algorithms have previously been shown to be a fast and effective alternative to convex L_1 regularization in sparse coding [148], and work even if the problem is only approximately submodular [149]. When a feature space is very large, greedy backward optimization will lead to long calculations and L_1 regularization becomes more suitable. Both heuristics are able to find relevant results in the examples presented here.

Like RBAs [62], also DII has a computational complexity of $\mathcal{O}(N^2 \cdot D)$, where N is the number of points and D is the number of features. However, by applying a simple subsampling trick (see subsection 6.2.4), the computational complexity reduces up to $\mathcal{O}(N \cdot D)$ with a degradation of the accuracy which is barely detectable.

The requirement of a ground truth reference space could pose a difficulty to some applications. In MD simulations, all heavy atom distances are a good, translationally invariant alternative to the set of all atomic positions, if one wants to completely encode the conformation of a molecule. In other cases, if no independent ground truth is known or a lower-dimensional subspace is desired, the full space could be used as ground truth. This approach could be employed, for example, for large gene sequencing data with thousands of features and just hundreds of data points. In this fashion, the method acts as an unsupervised feature selection filter. An open question in this case is the relative weighting of the ground truth features.

Furthermore, even though the method can be applied to any data set, it is most

suitable for continuous features. A limitation is given by ground truth metrics with many nominal or binary features, which can lead to a degenerate ground truth rank matrix, making the optimization more difficult.

The Differentiable Information Imbalance introduced in this thesis could have relevant implications in a wide range of distance-based methods, such as k-NN classification, clustering, and information retrieval. The approach could also be used to identify how much information original features carry compared to otherwise not-interpretable transformations such as UMAP [30] or highly non-linear neural network representations, by optimizing the original features towards such representations.

The Differentiable Information Imbalance has been implemented in the Python library DADApy [3] and is well-documented [68], including a tutorial for ease of use. This accessibility allows for a wide audience to explore further use cases and limitations effectively.

This chapter has shown a powerful development of the Information Imbalance: Its evolution into an optimizable statistic. This allowed us to remedy not only the challenges that remained from its other applications, but also to solve several open question in feature selection in general.

The next chapter will discuss the results of this thesis and outline the key takeaways, as well as open questions and possible future directions.

Chapter 7

Conclusion

This thesis has explored the idea of using the Information Imbalance and its variants *to perform feature selection*, providing a comprehensive study of its applicability across diverse fields such as medicine, ecology, and molecular dynamics. Central to this research is the idea that effective feature selection and weighting are crucial for interpretability and enhancing model performance, especially in scenarios where data is complex and high-dimensional. The foundation of Information Imbalance revolves around the disparity of information contained within different feature spaces towards each other. Having the possibility of estimating efficiently the information content of a description allows understanding how different variables contribute to model outcomes.

The exponential growth of data, transitioning from petabytes in the 1990s to the current zettabytes, has triggered the development of sophisticated approaches to manage, interpret, and derive knowledge and value from vast datasets. As outlined in the introduction, the dual nature of data's impact—enabling advancements while also presenting ecological challenges—sets the stage for all kinds of innovative solutions, among them the Information Imbalance.

The research was guided by three core technical questions:

1. How can Information Imbalance be adapted to handle class imbalances and missing data in medical datasets while ensuring robust feature selection?

2. In what ways can Information Imbalance be enhanced to optimize feature selection in datasets with continuous and categorical features?

3. Could the introduction of feature weights be a way to circumvent combinatorial problems and unit alignment for selection of features in high-dimensional spaces?

Several extensions of the Information Imbalance method have been shown in chapter 4, chapter 5 and chapter 6:

Initially we derive a weighted Information Imbalance approach which is able to deal with class imbalanced medical data and can handle missing data.

Then versions of Information Imbalance are introduced which can work with binary and discrete data, as we exemplified on biodiversity data from the Amazon Rainforest, where variables such as "region" are categorical, yet other variables, such as diversity indexes, are continuous variables.

Finally we described a variant of the Information Imbalance which is differentiable and optimizable, and which is also implemented in an easy-to-use python package, DADApy. This approach allows choosing feature subsets automatically along with the optimal size of subset. Moreover, it allows assigning meaningful relative weights to the chosen features.

Key Findings

The findings of this research demonstrate significant progress in addressing the challenges of feature selection:

In the first application of a severity prediction for a medical use case, Information Imbalance was compared as a feature selector to other methods, and the features selected by it were used to perform a subsequent kNN and support vector classification. The adaptation of the statistic to account for class imbalance in clinical datasets proved to be effective. The construction of a custom distance for the output space, the severity tree, was especially well suited to quantify patient fate. The study on COVID-19 severity prediction showcased the successful identification of a 13-feature subset from a pool of approximately 150 features. This method not only outperformed traditional techniques but also handled the complexities of missing data, emphasizing the need for improved data collection strategies for critical medical features.

The application of Information Imbalance in ecological studies, specifically assessing biodiversity in the Amazon Rainforest, highlighted the asymmetrical information between environmental conditions and species richness. By leveraging both, continuous and categorical features with dedicated statistics, we identified important predictors, underscoring the necessity for enhanced sampling strategies to better capture the diversity of ecosystems.

The introduction of the Differentiable Information Imbalance (DII) as an optimizable feature selection method marks an advancement in the field of feature selection. Utilizing gradient descent, DII effectively addressed the combinatorial challenges associated with high-dimensional data. The feature weights, which are the targets of the optimization, correct for different units of measure and relative importance, and allows inducing feature selection through sparsity. The method's application in molecular dynamics simulations demonstrated its robustness, achieving high descriptive power with a significantly reduced feature set of

only three collective variables.

Takeaway

The overarching takeaway from this research is the versatility and broad applicability of the Information Imbalance framework for feature selection. This framework offers flexibility in handling a variety of data types and structures: its variants can be adapted across fields to enhance predictive analyses in areas like healthcare, ecology, and molecular dynamics. By effectively managing continuous, categorical, and binary features across both high and low dimensions, the framework enables the integration of diverse data types. Additionally, the Differentiable Information Imbalance variant introduces an approach to optimally align features and determine the most suitable reduced feature set by optimizing a "loss function".

Limitations

A challenge that remains open in this thesis is the possibility to integrate the various Information Imbalance formulations designed to address different use cases. *E.g.* the functional form of $\Delta_{cat2con}$, the variant that captures information from a categorical space in relation to a continuous space, differs from the original Information Imbalance. Looking ahead, it would be desirable to develop a single formulation capable of handling all data types, or to create an integrated framework that automatically determines the appropriate variant. Ideally, this integrated version would also be incorporated into the differentiable framework, thereby enhancing the Differentiable Information Imbalance.

A second limitation is posed by ground truth feature spaces with many different features. While the Differentiable Information Imbalance can weight input features relative to each other, the challenge of constructing an optimal ground truth by weighting multiple target features relative to each other remains. What should one do when we have several features that would make good targets for our feature selection, but they have different units and intrinsic importance? Indeed, the integration of several reasonable target variables into a model is an open field of research [150]. For Information Imbalance, such features cannot simply be bundled into one Euclidean distance space without alignment. Future research could borrow from fields like multi-criteria decision-making (MCDM) in operations research, where the several ground truth features are called 'criteria' [151, 152], or latent variable modeling, where the given ground truth features could be interpreted as 'observed variables' [153].

Outlook

This thesis has demonstrated the broad applicability of the Information Imbalance method, which offers flexibility across diverse data types and dimensions. This adaptability arises from its conceptual foundation, relying on distance ranks between data points, which can accommodate any number of dimensions and can be tailored to system-specific needs, as shown with the COVID severity tree distances. Its applications extend beyond feature selection and weighting, with promising uses in fields like dimensionality reduction [154, 155] and metric learning [156], through the construction of a distance space $d^A(\boldsymbol{w})$ with a more expressive functional form.

Information Imbalance reproduces neighborhood relationships using a small number of variables, which may or may not include the target space variables, and therefore adds fluidity between supervised and unsupervised methods: When using a distinct ground truth, it functions as a supervised feature selector, while it works as an unsupervised method when sub-selecting features within the full feature space. An intriguing hybrid approach could involve using space B as target, and reusing all features of space B, additionally to features of space A, as input data space from which features are selected, enabling feature selection that bridges supervised and unsupervised methods.

Additionally, the Differentiable Information Imbalance optimizes feature weights to identify a reduced feature space. Creating new features by combining input features could lead to even more compressed and informative representations, though potentially at the cost of interpretability. This could involve extending the weight vector to a matrix that is optimized [157, 122], providing a more flexible and possibly lower-dimensional solution.

Beyond these possibilities of advancing the method, there are many more applications of the here presented Information Imbalance framework. Potential areas include analyzing genetic sequencing data [22], high-dimensional imaging in neuroscience [158] and medicine [159], and large datasets in astronomy [160] and environmental monitoring [161], where advanced imaging and model outputs provide ample high-dimensional data.

In summary, this thesis significantly broadens the scope of Information Imbalance applications, offering a solid foundation for future developments in feature selection and dimensionality reduction. Information Imbalance based methods present advantages over existing alternatives in automated feature selection and flexible analysis of high-dimensional data.

Acknowledgements

First and foremost I want to thank my supervisor, Alessandro Laio, for being the best supervisor imaginable and generally a person who I will always cherish and look up to. Not only did your scientific intuitive genius inspire me, but also your human kindness. I wish all of academia was like you. Next, I want to thank my amazing and fun collaborators in the two projects, Vittorio del Tatto, Felix Wodaczek and Emanuela Sozio, who is now pursuing her own inspiring journey.

I would like to extend my thanks to the people (at SISSA) who have served as role models and whose opinion I deeply appreciate. Besides Alessandro, these are Nour el Kazwini, Sara Folchini, Olivier Languin-Cattoën, Tullio Bigiarini and Simona Cerrato. Nour and Sara are some of the most intelligent and kindest people I know, not only helping me with physics and life, but also working for others constantly, for example by volunteering on their Saturdays in Sant'Egidio. If a work place would be made up only of Oliviers, it would be healthy, inclusive, productive and fun - your advice is the best. Tullio and Simona have a rare sense of what is important in life and strive to live according to it, which I hope to do as well. My thanks also go to the SISSA Mensa staff—Patrizia Fontanot, Debora Dicandia, and the entire team—for providing healthy and delicious meals, accompanied by a friendly smile, despite their demanding work. I'm sorry we students failed in improving your contracts.

During my time in SISSA I had to face challenges related with my social-environmental activities and I had the pleasure to be called trouble maker in a few occasions. This made me realize in the hard manner that what is considered appropriate for university and society can often be imposed by people who are higher up in the hierarchy. In my childhood, I envisioned community spaces being created through collaboration and discussion. Instead, at times I faced obstacles from individuals who actively hindered initiatives I and others were passionate about.

I am also not grateful to the toxic, capitalistic traits of academia. Social and environmental values are, in my view, critical aspects of any (scientific) endeavor, yet these are mostly overlooked in favor of a "more is better" and "this does not concern us" mindset. It would be wonderful to see a greater emphasis on envi-

ronmentally conscious research practices, including optimizing and testing code to reduce unnecessary resource use and selecting meaningful projects and data to work with.

On this notion, I want to extend my heartfelt gratitude to the people from the sustainability group SEA@SISSA, for caring and putting their free time into what we know matters most: Raising awareness for the ecological and climate crises and making SISSA a more sustainable place. Especially Dan Aguero Cerna, who leads the group with passion and ingenious ideas, and Anna Nikishova for founding it with me and organizing many meaningful events. I also want to thank Alex Zhang, Edward Donkor, Nina Javerzat, Regis Turuban, Younes Benyahia and Sasha Kenjeeva for being great friends with whom you can talk about anything, even and especially politics and philosophy. You always make me feel myself and have such honest positive vibes. To my mates from student council I want to say a huge thank you for working with me on important topics for our community and having some good laughs while doing the heavy lifting.

I would like to thank my parents, Heidi Regnet-Wild and Rainer Wild, for giving me all the necessary tools in life, figuratively by providing love and support, and literally, by helping me renovate and providing all tools from saws to drills. I love you. I want to thank Björn Michelsen for being a major positive influence on my thinking and direction. I also thank my best friend, Eugenia Plischuk, for being the kindest and most constructive person I know, and my partner in crime on many adventures. I thank my friend and colleague Marija Sorokina for becoming a data scientist with me during laughs and tears, and my friend and former supervisor Nishtha Gulati, for being the first person to show me what a scientist actually is. Finally, I want to thank my great love, moj maček, Matevž Kladnik, for always supporting me, cheering for me, practicing with me, and inspiring me. You are the best and smartest, and your humor in surprising moments makes life much brighter.

Even though this is just a tiny contribution to a specialized field within machine learning, for me this thesis meant being able to build a much better understanding of quantitative methods. I hope to put this to good use.

Bibliography

[1] Wild, Romina and Sozio, Emanuela and Margiotta, Riccardo G. and Dellai, Fabiana and Acquasanta, Angela and Del Ben, Fabio and Tascini, Carlo and Curcio, Francesco and Laio, Alessandro. "Maximally informative feature selection using Information Imbalance: Application to COVID-19 severity prediction". In: *Scientific Reports* 14.1 (May 2024), p. 10744. ISSN: 2045-2322. DOI: 10.1038/s41598-024-61334-6. URL: https://doi.org/10.1038/s41598-024-61334-6.

[2] Romina Wild et al. *Automatic feature selection and weighting using Differentiable Information Imbalance*. 2024. arXiv: 2411.00851 [cs.LG]. URL: https://arxiv.org/abs/2411.00851.

[3] Aldo Glielmo et al. "DADApy: Distance-based analysis of data-manifolds in Python". In: *Patterns* 3.10 (2022), p. 100589. ISSN: 2666-3899. DOI: https://doi.org/10.1016/j.patter.2022.100589. URL: https://www.sciencedirect.com/science/article/pii/S2666389922002070.

[4] Anders S. G. Andrae and Tomas Edler. "On Global Electricity Usage of Communication Technology: Trends to 2030". In: *Challenges* 6.1 (2015), pp. 117–157. ISSN: 2078-1547. DOI: 10.3390/challe6010117. URL: https://www.mdpi.com/2078-1547/6/1/117.

[5] Nicola Jones. "How to stop data centres from gobbling up the world's electricity". In: *Nature* 561 (7722 Sept. 2018), pp. 163–166. ISSN: 14764687. DOI: 10.1038/D41586-018-06610-Y.

[6] Carole-Jean Wu et al. "Sustainable AI: Environmental Implications, Challenges and Opportunities". In: *Proceedings of Machine Learning and Systems*. Ed. by D. Marculescu, Y. Chi, and C. Wu. Vol. 4. 2022, pp. 795–813. URL: https://proceedings.mlsys.org/paper_files/paper/2022/file/462211f67c7d858f663355eff93b745e-Paper.pdf.

[7] Udit Gupta et al. "Chasing Carbon: The Elusive Environmental Footprint of Computing". In: *IEEE Micro* 42 (4 2022), pp. 37–47. ISSN: 19374143. DOI: 10.1109/MM.2022.3163226. URL: https://dl.acm.org/doi/10.1109/MM.2022.3163226.

[8] EnerData. *European Union Energy Information | Enerdata*. Accessed on Sep 21, 2024. URL: https://www.enerdata.net/estore/energy-market/european-union/.

[9] U.S. Energy Information Administration. *Use of electricity - U.S. Energy Information Administration (EIA)*. Accessed on Sep 21, 2024. URL: https://www.eia.gov/energyexplained/electricity/use-of-electricity.php.

[10] EnerData. *India Energy Information | Enerdata*. Accessed on Sep 21, 2024. URL: https://www.enerdata.net/estore/energy-market/india/.

[11] H. Lee Core Writing Team and J. Romero (eds.) *IPCC, 2023: Climate Change 2023: Synthesis Report. Contribution of Working Groups I, II and III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Ed. by Paola Arias et al. July 2023. DOI: 10.59327/IPCC/AR6-9789291691647. URL: https://www.ipcc.ch/report/ar6/syr/.

[12] Emanuela Sozio et al. "The role of asymmetric dimethylarginine (ADMA) in COVID-19: association with respiratory failure and predictive role for outcome". In: *Scientific Reports* 13 (June 2023). DOI: 10.1038/s41598-023-36954-z.

[13] Aldo Glielmo et al. "Ranking the information content of distance measures". In: *PNAS Nexus* 1.2 (Apr. 2022), pgac039. DOI: 10.1093/pnasnexus/pgac039.

[14] Jens K. Roehrich Wendy Phillips and Dharm Kapletia. "Responding to information asymmetry in crisis situations: innovation in the time of the COVID-19 pandemic". In: *Public Management Review* 25.1 (2023), pp. 175–198. DOI: 10.1080/14719037.2021.1960737. URL: https://doi.org/10.1080/14719037.2021.1960737.

[15] Edward Danquah Donkor, Alessandro Laio, and Ali Hassanali. "Do Machine-Learning Atomic Descriptors and Order Parameters Tell the Same Story? The Case of Liquid Water". In: *Journal of Chemical Theory and Computation* 19.14 (2023). https://arxiv.org/pdf/2211.16196, pp. 4596–4605. DOI: 10.1021/acs.jctc.2c01205. URL: https://app.dimensions.ai/details/publication/pub.1156217616.

[16] C. Fuchs et al. *Internet and Surveillance: The Challenges of Web 2.0 and Social Media*. Routledge Studies in Science, Technology and Society. Taylor & Francis, 2013. ISBN: 9781136655265. URL: https://books.google.it/books?id=K3xzhq5UulEC.

[17] I. Chaston. *Internet Marketing and Big Data Exploitation*. Online Marketing and Big Data Exploration. Palgrave Macmillan UK, 2015. ISBN: 9781137488961. URL: https://books.google.it/books?id=hUsTBwAAQBAJ.

[18] Thomas Christiano. "Algorithms, Manipulation, and Democracy". In: *Canadian Journal of Philosophy* 52.1 (2022), pp. 109–124. DOI: 10.1017/can.2021.29.

[19] Massimo Ragnedda. "New Digital Inequalities. Algorithms Divide". In: *Enhancing Digital Equity: Connecting the Digital Underclass*. Cham: Springer International Publishing, 2020, pp. 61–83. ISBN: 978-3-030-49079-9. DOI: 10.1007/978-3-030-49079-9_4. URL: https://doi.org/10.1007/978-3-030-49079-9_4.

[20] Ferry Hooft, Alberto Pérez de Alba Ortíz, and Bernd Ensing. "Discovering Collective Variables of Molecular Transitions via Genetic Algorithms and Neural Networks". In: *Journal of Chemical Theory and Computation* 17.4 (2021), pp. 2294–2306. ISSN: 1549-9618. DOI: 10.1021/acs.jctc.0c00981. URL: https://doi.org/10.1021/acs.jctc.0c00981.

[21] Nour El Kazwini and Guido Sanguinetti. "SHARE-Topic: Bayesian interpretable modeling of single-cell multi-omic data". In: *Genome Biology* 25.1 (Feb. 2024), p. 55. ISSN: 1474-760X. DOI: 10.1186/s13059-024-03180-3. URL: https://doi.org/10.1186/s13059-024-03180-3.

[22] Yan Wu and Kun Zhang. "Tools for the analysis of high-dimensional single-cell RNA sequencing data". In: *Nature Reviews Nephrology* 16.7 (July 2020), pp. 408–421. ISSN: 1759-507X. DOI: 10.1038/s41581-020-0262-0. URL: https://doi.org/10.1038/s41581-020-0262-0.

[23] Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. "Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Ed. by Chengqing Zong et al. Online: Association for Computational Linguistics, Aug. 2021, pp. 7319–7328. DOI: 10.18653/v1/2021.acl-long.568. URL: https://aclanthology.org/2021.acl-long.568.

[24] Hans Ter Steege et al. "Estimating the global conservation status of more than 15,000 Amazonian tree species". In: *Science Advances* 1 (10 Nov. 2015). ISSN: 23752548. DOI: 10.1126/SCIADV.1500936. URL: https://www.science.org.

[25] Anna Tovo et al. "Upscaling species richness and abundances in tropical forests". In: *Science Advances* 3 (10 Oct. 2017). ISSN: 23752548. DOI: 10.1126/SCIADV.1701438/SUPPL_FILE/1701438_SM.PDF. URL: https://www.science.org/doi/10.1126/sciadv.1701438.

[26] I. T. Jolliffe. "Principal Component Analysis and Factor Analysis". In: *Principal Component Analysis*. New York, NY: Springer New York, 1986, pp. 115–128. ISBN: 978-1-4757-1904-8. DOI: 10.1007/978-1-4757-1904-8_7.

[27] Yasi Wang, Hongxun Yao, and Sicheng Zhao. "Auto-encoder based dimensionality reduction". In: *Neurocomputing* 184 (2016). RoLoD: Robust Local Descriptors for Computer Vision 2014, pp. 232–242. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.08.104. URL: https://www.sciencedirect.com/science/article/pii/S0925231215017671.

[28] Stéphane S. Lafon. "Diffusion Maps and Geometric Harmonics". PhD thesis. Yale, 2004.

[29] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. "Nonlinear Component Analysis as a Kernel Eigenvalue Problem". In: *Neural Computation* 10.5 (1998), pp. 1299–1319. DOI: 10.1162/089976698300017467.

[30] Leland McInnes et al. "UMAP: Uniform Manifold Approximation and Projection". In: *Journal of Open Source Software* 3.29 (2018), p. 861. DOI: 10.21105/joss.00861. URL: https://doi.org/10.21105/joss.00861.

[31] YuWen Chen, Ju Zhang, and XiaoLin Qin. "Interpretable instance disease prediction based on causal feature selection and effect analysis". In: *BMC Medical Informatics and Decision Making* 22.1 (Feb. 2022), p. 51. ISSN: 1472-6947. DOI: 10.1186/s12911-022-01788-8.

[32] Beatriz Remeseiro and Veronica Bolon-Canedo. "A review of feature selection methods in medical applications". In: *Computers in Biology and Medicine* 112 (2019), p. 103375. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2019.103375. URL: https://www.sciencedirect.com/science/article/pii/S0010482519302525.

[33] Muhammad Salman Pathan et al. "Analyzing the impact of feature selection on the accuracy of heart disease prediction". In: *Healthcare Analytics* 2 (2022), p. 100060. ISSN: 2772-4425. DOI: https://doi.org/10.1016/j.health.2022.100060. URL: https://www.sciencedirect.com/science/article/pii/S2772442522000235.

[34] Kyung Keun Yun, Sang Won Yoon, and Daehan Won. "Interpretable stock price forecasting model using genetic algorithm-machine learning regressions and best feature subset selection". In: *Expert Systems with Applications* 213 (2023), p. 118803. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2022.118803.

[35] M. Anand et al. "Deep learning and natural language processing in computation for offensive language detection in online social networks by feature selection and ensemble classification techniques". In: *Theoretical Computer Science* 943 (2023), pp. 203–218. ISSN: 0304-3975. DOI: https://doi.org/10.1016/j.tcs.2022.06.020. URL: https://www.sciencedirect.com/science/article/pii/S0304397522003887.

[36] Isabelle Guyon and André Elisseeff. "An Introduction to Variable and Feature Selection". In: *J. Mach. Learn. Res.* 3.null (Mar. 2003), pp. 1157–1182. ISSN: 1532-4435.

[37] Md. Alamgir Sarder, Md. Maniruzzaman, and Benojir Ahammed. "Feature Selection and Classification of Leukemia Cancer Using Machine Learning Techniques". In: 5 (July 2020), pp. 18–27. DOI: 10.11648/j.mlr.20200502.11.

[38] Hassan Eldeeb, Shota Amashukeli, and Radwa El Shawi. "An Empirical Analysis of Integrating Feature Extraction to Automated Machine Learning Pipeline". In: *Pattern Recognition. ICPR International Workshops and Challenges*. Ed. by Alberto Del Bimbo et al. Cham: Springer International Publishing, 2021, pp. 336–344.

[39] Rui Zhang et al. "Feature selection with multi-view data: A survey". In: *Information Fusion* 50 (2019), pp. 158–167. ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2018.11.019. URL: https://www.sciencedirect.com/science/article/pii/S1566253518303841.

[40] P. Campadelli et al. "Intrinsic Dimension Estimation: Relevant Techniques and a Benchmark Framework". In: *Mathematical Problems in Engineering* 2015 (Oct. 2015), p. 759567. ISSN: 1024-123X. DOI: 10.1155/2015/759567.

[41] Elena Facco et al. "Estimating the intrinsic dimension of datasets by a minimal neighborhood information". In: *Scientific Reports* 7.1 (Sept. 2017), p. 12140. ISSN: 2045-2322. DOI: 10.1038/s41598-017-11873-y.

[42] Michele Allegra et al. "Data segmentation based on the local intrinsic dimension". In: *Scientific Reports* 10.1 (Oct. 2020), p. 16449. ISSN: 2045-2322. DOI: 10.1038/s41598-020-72222-0.

[43] Girish Chandrashekar and Ferat Sahin. "A survey on feature selection methods". In: *Computers & Electrical Engineering* 40 (1 Jan. 2014), pp. 16–28. DOI: https://doi.org/10.1016/j.compeleceng.2013.11.024.

[44] Jun Chin Ang et al. "Supervised, Unsupervised, and Semi-Supervised Feature Selection: A Review on Gene Selection". In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 13.5 (2016), pp. 971–989. DOI: 10.1109/TCBB.2015.2478454.

[45] Roberto Battiti. "Using Mutual Information for Selecting Features in Supervised Neural Net Learning". In: *Neural Networks, IEEE Transactions on* 5 (Aug. 1994), pp. 537–550. DOI: 10.1109/72.298224.

[46] Leo Breiman. "Random Forests". In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 1573-0565. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324.

[47] Ke Yan and David Zhang. "Feature selection and analysis on correlated gas sensor data with recursive feature elimination". In: *Sensors and Actuators B: Chemical* 212 (2015), pp. 353–363. ISSN: 0925-4005. DOI: https://doi.org/10.1016/j.snb.2015.02.025. URL: https://www.sciencedirect.com/science/article/pii/S0925400515001872.

[48] Xuyang Yan et al. "An efficient unsupervised feature selection procedure through feature clustering". In: *Pattern Recognition Letters* 131 (2020), pp. 277–284. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2019.12.022. URL: https://www.sciencedirect.com/science/article/pii/S0167865519303976.

[49] Konstantin Hopf and Sascha Reifenrath. *Filter Methods for Feature Selection in Supervised Machine Learning Applications – Review and Benchmark*. 2021. arXiv: 2111.12140 [cs.LG].

[50] Xia Wu et al. "Supervised Feature Selection With Orthogonal Regression and Feature Weighting". In: *IEEE Transactions on Neural Networks and Learning Systems* 32.5 (2021), pp. 1831–1838. DOI: 10.1109/TNNLS.2020.2991336.

[51] Trevor Hastie and Robert Tibshirani. *Generalized additive models*. Wiley Online Library, 1990.

[52] Sebastián Maldonado, Richard Weber, and Jayanta Basak. "Simultaneous feature selection and classification using kernel-penalized support vector machines". In: *Information Sciences* 181.1 (2011), pp. 115–128. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2010.08.047.

[53] Sebastián Maldonado and Julio López. "Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification". In: *Applied Soft Computing* 67 (2018), pp. 94–105. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2018.02.051. URL: https://www.sciencedirect.com/science/article/pii/S1568494618301108.

[54] Yanfang Liu et al. "Robust neighborhood embedding for unsupervised feature selection". In: *Knowledge-Based Systems* 193 (2020), p. 105462. ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2019.105462. URL: https://doi.org/10.1016/j.knosys.2019.105462.

[55] Heyong Wang and Ming Hong. "Distance Variance Score: An Efficient Feature Selection Method in Text Classification". In: *Mathematical Problems in Engineering* 2015 (May 2015), p. 695720. ISSN: 1024-123X. DOI: 10.1155/2015/695720.

[56] Xiaofei He, Deng Cai, and Partha Niyogi. "Laplacian Score for Feature Selection". In: *Advances in Neural Information Processing Systems*. Ed. by Y. Weiss, B. Schölkopf, and J. Platt. Vol. 18. MIT Press, 2005. URL: https://proceedings.neurips.cc/paper_files/paper/2005/file/b5b03f06271f8917685d14cea7c6c50a-Paper.pdf.

[57] Deng Cai, Chiyuan Zhang, and Xiaofei He. "Unsupervised feature selection for multi-cluster data". In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '10. Washington, DC, USA: Association for Computing Machinery, 2010, pp. 333–342. ISBN: 9781450300551. DOI: 10.1145/1835804.1835848. URL: https://doi.org/10.1145/1835804.1835848.

[58] Christos Boutsidis, Petros Drineas, and Michael W Mahoney. "Unsupervised Feature Selection for the k-means Clustering Problem". In: *Advances in Neural Information Processing Systems*. Ed. by Y. Bengio et al. Vol. 22. Curran Associates, Inc., 2009. URL: https://proceedings.neurips.cc/paper_files/paper/2009/file/c51ce410c124a10e0db5e4b97fc2af39-Paper.pdf.

[59] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. "Estimating mutual information". In: *Phys. Rev. E* 69 (6 June 2004), p. 066138. DOI: https://doi.org/10.1103/PhysRevE.69.066138. URL: https://link.aps.org/doi/10.1103/PhysRevE.69.066138.

[60] Lars St»hle and Svante Wold. "Analysis of variance (ANOVA)". In: *Chemometrics and Intelligent Laboratory Systems* 6.4 (1989), pp. 259–272. ISSN: 0169-7439. DOI: https://doi.org/10.1016/0169-7439(89)80095-4.

[61] Hussein Almuallim, Thomas G Dietterich, et al. "Learning With Many Irrelevant Features." In: *AAAI*. Vol. 91. Citeseer. 1991, pp. 547–552.

[62] Ryan J. Urbanowicz et al. "Relief-based feature selection: Introduction and review". In: *Journal of Biomedical Informatics* 85 (2018), pp. 189–203. ISSN: 1532-0464. DOI: https://doi.org/10.1016/j.jbi.2018.07.014.

[63] Kenji Kira and Larry A Rendell. "A practical approach to feature selection". In: *Machine learning proceedings 1992*. Elsevier, 1992, pp. 249–256.

[64] Zeinab Noroozi, Azam Orooji, and Leila Erfannia. "Analyzing the impact of feature selection methods on machine learning algorithms for heart disease prediction". In: *Scientific Reports* 13.1 (Dec. 2023), p. 22588. ISSN: 2045-2322. DOI: 10.1038/s41598-023-49962-w. URL: https://doi.org/10.1038/s41598-023-49962-w.

[65] Shahadat Uddin and Haohui Lu. "Dataset meta-level and statistical features affect machine learning performance". In: *Scientific Reports* 14.1 (Jan. 2024), p. 1670. ISSN: 2045-2322. DOI: 10.1038/s41598-024-51825-x. URL: https://doi.org/10.1038/s41598-024-51825-x.

[66] Jundong Li et al. "Feature Selection: A Data Perspective". In: *ACM Comput. Surv.* 50.6 (Dec. 2017). ISSN: 0360-0300. DOI: 10.1145/3136625. URL: https://doi.org/10.1145/3136625.

[67] Anand Sharma, Chen Liu, and Misaki Ozawa. *Selecting Relevant Structural Features for Glassy Dynamics by Information Imbalance*. 2024. arXiv: 2408.12705 [cond-mat.soft]. URL: https://arxiv.org/abs/2408.12705.

[68] The DADApy Authors. *Distance-based Analysis of DAta-manifolds in python (DADApy)*. Accessed on March 28, 2024. 2021. URL: https://dadapy.readthedocs.io/en/latest/index.html.

[69] Kui Yu et al. "Causality-Based Feature Selection: Methods and Evaluations". In: *ACM Comput. Surv.* 53.5 (Sept. 2020). ISSN: 0360-0300. DOI: https://doi.org/10.1145/3409382.

[70] Daniela M. Witten and Robert Tibshirani. "Covariance-Regularized Regression and Classification for high Dimensional Problems". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 71.3 (Feb. 2009), pp. 615–636. ISSN: 1369-7412. DOI: 10.1111/j.1467-9868.2009.00699.x. eprint: https://academic.oup.com/jrsssb/article-pdf/71/3/615/49686350/jrsssb_71_3_615.pdf. URL: https://doi.org/10.1111/j.1467-9868.2009.00699.x.

[71] Pradeep Ravikumar et al. “Sparse Additive Models”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 71.5 (Oct. 2009), pp. 1009–1030. ISSN: 1369-7412. DOI: 10.1111/j.1467-9868.2009.00718.x. URL: https://doi.org/10.1111/j.1467-9868.2009.00718.x.

[72] Shiyun Xu et al. “Sparse Neural Additive Model: Interpretable Deep Learning with Feature Selection via Group Sparsity”. In: *Machine Learning and Knowledge Discovery in Databases: Research Track*. Ed. by Danai Koutra et al. Cham: Springer Nature Switzerland, 2023, pp. 343–359.

[73] Rok Blagus and Lara Lusa. “Class prediction for high-dimensional class-imbalanced data.” In: *BMC bioinformatics* 11 (2010). DOI: https://doi.org/10.1186/1471-2105-11-523.

[74] David Furcy and Sven Koenig. “Limited Discrepancy Beam Search”. In: *Proceedings of the 19th International Joint Conference on Artificial Intelligence*. IJCAI’05. Edinburgh, Scotland: Morgan Kaufmann Publishers Inc., 2005, pp. 125–131.

[75] Jiadong Zhang et al. “Multi-label learning with Relief-based label-specific feature selection”. In: *Applied Intelligence* 53.15 (2023), pp. 18517–18530. DOI: 10.1007/s10489-022-04350-1. URL: https://doi.org/10.1007/s10489-022-04350-1.

[76] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830. DOI: https://doi.org/10.48550/arXiv.1201.0490.

[77] Thomas M Stulnig. “C-reactive protein, fibrinogen, and cardiovascular risk”. In: *New England Journal of Medicine* 368.1 (2013), pp. 84–86. DOI: https://doi.org/10.1056/NEJMc1213688.

[78] Gordon DO Lowe, Ann Rumley, and Ian J Mackie. “Plasma fibrinogen”. In: *Annals of Clinical Biochemistry* 41.6 (2004), pp. 430–440. DOI: https://doi.org/10.1258/0004563042466884. URL: https://doi.org/10.1258/0004563042466884.

[79] Giuseppe Lippi, Carl J. Lavie, and Fabian Sanchis-Gomar. “Cardiac troponin I in patients with coronavirus disease 2019 (COVID-19): Evidence from a meta-analysis”. In: *Progress in Cardiovascular Diseases* 63.3 (2020), pp. 390–391. ISSN: 0033-0620. DOI: https://doi.org/10.1016/j.pcad.2020.03.001. URL: https://www.sciencedirect.com/science/article/pii/S0033062020300554.

[80] Tao Chen et al. "Clinical characteristics of 113 deceased patients with coronavirus disease 2019: retrospective study". In: *BMJ* 368 (2020). DOI: https://doi.org/10.1136/bmj.m1091. URL: https://www.bmj.com/content/368/bmj.m1091.

[81] Shaobo Shi et al. "Association of Cardiac Injury With Mortality in Hospitalized Patients With COVID-19 in Wuhan, China". In: *JAMA Cardiology* 5.7 (July 2020), pp. 802–810. ISSN: 2380-6583. DOI: https://doi.org/10.1001/jamacardio.2020.0950. URL: https://doi.org/10.1001/jamacardio.2020.0950.

[82] Tao Guo et al. "Cardiovascular Implications of Fatal Outcomes of Patients With Coronavirus Disease 2019 (COVID-19)". In: *JAMA Cardiology* 5.7 (July 2020), pp. 811–818. ISSN: 2380-6583. DOI: https://doi.org/10.1001/jamacardio.2020.1017. URL: https://doi.org/10.1001/jamacardio.2020.1017.

[83] Nasrin Amiri-Dashatan et al. "Increased inflammatory markers correlate with liver damage and predict severe COVID-19: a systematic review and meta-analysis". In: *Gastroenterology and Hepatology from Bed to Bench* 13.4 (Sept. 2020), pp. 282–291. DOI: https://doi.org/10.22037/ghfbb.v13i4.2038. URL: https://journals.sbmu.ac.ir/ghfbb/index.php/ghfbb/article/view/2038.

[84] Jafar Tanha et al. "Boosting methods for multi-class imbalanced data classification: an experimental review". In: *Journal of Big Data* 7 (2020), pp. 1–47.

[85] Qianmu Li et al. "Multiclass imbalanced learning with one-versus-one decomposition and spectral clustering". In: *Expert Systems with Applications* 147 (2020), p. 113152. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2019.113152.

[86] Saqib Ejaz Awan et al. "Imputation of missing data with class imbalance using conditional generative adversarial networks". In: *Neurocomputing* 453 (2021), pp. 164–171. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2021.04.010. URL: https://www.sciencedirect.com/science/article/pii/S0925231221005282.

[87] Martina Fabris et al. "Cytokines from Bench to Bedside: A Retrospective Study Identifies a Definite Panel of Biomarkers to Early Assess the Risk of Negative Outcome in COVID-19 Patients". In: *International Journal of Molecular Sciences* 23.9 (2022). ISSN: 1422-0067. DOI: https://doi.org/10.3390/ijms23094830. URL: https://www.mdpi.com/1422-0067/23/9/4830.

[88] National Center for Immunization and Respiratory Diseases (NCIRD), Division of Viral Diseases. *People with certain medical conditions*. https://www.cdc.gov/coronavirus/2019-ncov/need-extra-precautions/people-with-medical-conditions.html. Accessed: 2022-03-18. (Visited on 03/18/2022).

[89] A. Rogier T. Donders et al. "Review: A gentle introduction to imputation of missing values". In: *Journal of Clinical Epidemiology* 59.10 (2006), pp. 1087–1091. ISSN: 0895-4356. DOI: https://doi.org/10.1016/j.jclinepi.2006.01.014. URL: https://www.sciencedirect.com/science/article/pii/S0895435606001971.

[90] Christine Schlömmer, Anna Brandtner, and Mirjam Bachler. "Antithrombin and its role in host defense and inflammation". In: *International Journal of Molecular Sciences* 22.8 (2021), p. 4283. DOI: https://doi.org/10.3390/ijms22084283.

[91] Thomas Marjot et al. "COVID-19 and liver disease: Mechanistic and clinical perspectives". In: *Nature Reviews Gastroenterology & Hepatology* 18.5 (2021), pp. 348–364. DOI: https://doi.org/10.1038/s41575-021-00426-4.

[92] Dinesh Jothimani et al. "COVID-19 and the liver". In: *Journal of Hepatology* 73.5 (2020), pp. 1231–1240. ISSN: 0168-8278. DOI: https://doi.org/10.1016/j.jhep.2020.06.006. URL: https://www.sciencedirect.com/science/article/pii/S0168827820303779.

[93] Jonathan Coddington, Robert K Colwell, and Jonathan A Coddington. "Estimating terrestrial biodiversity through extrapolation". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* (1994).

[94] Tom Fisk. *Aerial Photography of Green-leafed Trees*. [Online; accessed October 18, 2024]. 2019. URL: https://www.pexels.com/photo/aerial-photography-of-green-leafed-trees-2739666/.

[95] Hans ter Steege et al. "Biased-corrected richness estimates for the Amazonian tree flora". In: *Scientific Reports* 10.1 (June 2020), p. 10130. ISSN: 2045-2322. DOI: 10.1038/s41598-020-66686-3. URL: https://doi.org/10.1038/s41598-020-66686-3.

[96] Malin Rivers et al. *European Red List of Trees*. IUCN Red List of Threatened Species – Regional Assessment. United Kingdom: International Union for Conservation of Nature and Natural Resources (IUCN), Aug. 2019. ISBN: 978-2-8317-1986-3. DOI: 10.2305/IUCN.CH.2019.ERL.1.en.

[97] C Nobre et al. *Science panel for the Amazon: Amazon Assessment Report 2021: executive summary*. 2021. DOI: 10.55161/RWSX6527.

[98] Marizilda Cruppe. *Tropical forests around the world are already seeing a rise in temperature, new study warns*. [Online; accessed October 18, 2024]. 2019. URL: https://www.paraterraboa.com/meio-ambiente/florestas-tropicais-em-todo-o-mundo-ja-registram-aumento-de-temperatura-alerta-novo-estudo/.

[99] Copernicus Atmosphere Monitoring Service (CAMS). *Copernicus: Pantanal and Amazon wildfires saw their worst wildfires in almost two decades*. 2024. URL: https://atmosphere.copernicus.eu/copernicus-pantanal-and-amazon-wildfires-saw-their-worst-wildfires-almost-two-decades (visited on 10/18/2024).

[100] La Nación. *"Se hará más denso a medida que transcurra el día": el humo llegó a la ciudad de Buenos Aires y la alerta se amplía a 15 distritos*. 2024. URL: https://www.lanacion.com.ar/sociedad/se-hara-mas-denso-a-medida-que-transcurra-el-dia-el-humo-ya-llego-a-la-ciudad-de-buenos-aires-nid09092024/ (visited on 10/18/2024).

[101] Bernardo M. Flores et al. "Critical transitions in the Amazon forest system". In: *Nature* 626.7999 (Feb. 2024), pp. 555–564. ISSN: 1476-4687. DOI: 10.1038/s41586-023-06970-0. URL: https://doi.org/10.1038/s41586-023-06970-0.

[102] Arie Staal et al. "Feedback between drought and deforestation in the Amazon". In: *Environmental Research Letters* 15.4 (Apr. 2020), p. 044024. DOI: 10.1088/1748-9326/ab738e. URL: https://dx.doi.org/10.1088/1748-9326/ab738e.

[103] Alessandro Chiarucci, Giovanni Bacaro, and Samuel M Scheiner. "Old and new challenges in using species diversity for assessing biodiversity". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 366 (1576 2011), pp. 2426–2437. DOI: 10.1098/rstb.2011.0065. URL: https://about.jstor.org/terms.

[104] R. A. Fisher, A. Steven Corbet, and C. B. Williams. "The Relation Between the Number of Species and the Number of Individuals in a Random Sample of an Animal Population". In: *Journal of Animal Ecology* 12.1 (May 1943), p. 42. DOI: 10.2307/1411.

[105] Hans Ter Steege et al. "Mapping density, diversity and species-richness of the Amazon tree flora". In: *Communications Biology 2023 6:1* 6 (1 Nov. 2023), pp. 1–14. ISSN: 2399-3642. DOI: 10.1038/s42003-023-05514-6. URL: https://www.nature.com/articles/s42003-023-05514-6.

[106] Claude Elwood Shannon. "A Mathematical Theory of Communication". In: *The Bell System Technical Journal* 27 (1948), pp. 379–423. URL: http://plan9.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf (visited on 04/22/2003).

[107] E. H. SIMPSON. "Measurement of Diversity". In: *Nature* 163.4148 (Apr. 1949), pp. 688–688. ISSN: 1476-4687. DOI: 10.1038/163688a0. URL: https://doi.org/10.1038/163688a0.

[108] Anne Chao. "Nonparametric Estimation of the Number of Classes in a Population". In: *Scandinavian Journal of Statistics* 11.4 (1984), pp. 265–270. ISSN: 03036898, 14679469. URL: http://www.jstor.org/stable/4615964 (visited on 10/19/2024).

[109] Anne Chao. "Estimating the Population Size for Capture-Recapture Data with Unequal Catchability". In: *Biometrics* 43 (Jan. 1988), pp. 783–91. DOI: 10.2307/2531532.

[110] Linda Seggi, Raffaella Trabucco, and Stefano Martellos. "Valorization of Historical Natural History Collections Through Digitization: The Algarium Vatova–Schiffner". In: *Plants* 13.20 (2024). ISSN: 2223-7747. DOI: 10.3390/plants13202901. URL: https://www.mdpi.com/2223-7747/13/20/2901.

[111] Stefano Martellos et al. "Lichens and air quality: a new citizen science approach". In: *ARPHA Proceedings* 6 (2024), pp. 37–42. DOI: 10.3897/ap.e126131. eprint: https://doi.org/10.3897/ap.e126131. URL: https://doi.org/10.3897/ap.e126131.

[112] James P. Darby, James R. Kermode, and Gábor Csányi. "Compressing local atomic neighbourhood descriptors". In: *npj Computational Materials* 8.1 (Aug. 2022), p. 166. ISSN: 2057-3960. DOI: 10.1038/s41524-022-00847-y. URL: https://doi.org/10.1038/s41524-022-00847-y.

[113] Paul Shannon et al. "Cytoscape: a software environment for integrated models of biomolecular interaction networks". In: *Genome research* 13.11 (2003), pp. 2498–2504.

[114] Feiping Nie, Jing Li, and Xuelong Li. "Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification". In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. IJCAI'16. New York, New York, USA: AAAI Press, 2016, pp. 1881–1887. ISBN: 9781577357704.

[115] Vittorio Del Tatto et al. "Robust inference of causality in high-dimensional dynamical processes from the Information Imbalance of distance ranks". In: *Proceedings of the National Academy of Sciences* 121.19 (2024), e2317256121. DOI: 10.1073/pnas.2317256121. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2317256121. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2317256121.

[116] Jörg Behler and Michele Parrinello. "Generalized neural-network representation of high-dimensional potential-energy surfaces." In: *Physical review letters* 98 14 (2007), p. 146401. URL: https://api.semanticscholar.org/CorpusID:37065565.

[117] Albert P. Bartók, Risi Kondor, and Gábor Csányi. "On representing chemical environments". In: *Phys. Rev. B* 87 (18 May 2013), p. 184115. DOI: 10.1103/PhysRevB.87.184115. URL: https://link.aps.org/doi/10.1103/PhysRevB.87.184115.

[118] Sandip De et al. "Comparing molecules and solids across structural and alchemical space". In: *Phys. Chem. Chem. Phys.* 18 (20 2016), pp. 13754–13769. DOI: 10.1039/C6CP00415F. URL: http://dx.doi.org/10.1039/C6CP00415F.

[119] Kevin Beyer et al. "When Is "Nearest Neighbor" Meaningful?" In: *Database Theory — ICDT'99*. Ed. by Catriel Beeri and Peter Buneman. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 217–235.

[120] Alexander Hinneburg, Charu Aggarwal, and Daniel Keim. "What is the Nearest Neighbor in High Dimensional Spaces?" In: *First publ. in: Proc. of the 26th Internat. Conference on Very Large Databases, Cairo, Egypt, 2000, pp. 506-515* 671675 (Oct. 2000).

[121] Yoshimasa Tsuruoka, Jun'ichi Tsujii, and Sophia Ananiadou. "Stochastic Gradient Descent Training for L1-regularized Log-linear Models with Cumulative Penalty". In: *ACL and AFNLP* (2009), pp. 477–485.

[122] Francis Bach et al. "Optimization with Sparsity-Inducing Penalties". In: *Foundations and Trends in Machine Learning* 4 (1 2011), pp. 1–106. ISSN: 1935-8237. DOI: 10.1561/2200000015.

[123] Hui Zou and Trevor Hastie. "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2 (2005), pp. 301–320. DOI: https://doi.org/10.1111/j.1467-9868.2005.00503.x. URL: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2005.00503.x.

[124] Bob Carpenter. *Lazy Sparse Stochastic Gradient Descent for Regularized Mutlinomial Logistic Regression*. Alias-i, Inc., 2008. URL: https://lingpipe.files.wordpress.com/2008/04/lazysgdregression.pdf.

[125] Gareth A. Tribello et al. "PLUMED 2: New feathers for an old bird". In: *Computer Physics Communications* 185.2 (2014), pp. 604–613. ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2013.09.018. URL: https://www.sciencedirect.com/science/article/pii/S0010465513003196.

[126] Gerald Lippert, Jürg Hutter, and Michele Parrinello. "The Gaussian and augmented-plane-wave density functional method for ab initio molecular dynamics simulations". In: *Theoretical Chemistry Accounts* 103 (1999), pp. 124–140. DOI: 10.1007/s002140050523. URL: https://api.semanticscholar.org/CorpusID:124305820.

[127] Lauri Himanen et al. "DScribe: Library of descriptors for machine learning in materials science". In: *Computer Physics Communications* 247 (2020), p. 106949. ISSN: 0010-4655. DOI: 10.1016/j.cpc.2019.106949. URL: https://doi.org/10.1016/j.cpc.2019.106949.

[128] Jarno Laakso et al. "Updates to the DScribe library: New descriptors and derivatives". In: *The Journal of Chemical Physics* 158.23 (2023).

[129] M. D. Springer and W. E. Thompson. "The Distribution of Products of Beta, Gamma and Gaussian Random Variables". In: *SIAM Journal on Applied Mathematics* 18.4 (1970), pp. 721–737. URL: http://www.jstor.org/stable/2099424.

[130] Ryan J. Urbanowicz et al. "Benchmarking relief-based feature selection methods for bioinformatics data mining". In: *Journal of Biomedical Informatics* 85 (2018), pp. 168–188. DOI: https://doi.org/10.1016/j.jbi.2018.07.015. URL: https://www.sciencedirect.com/science/article/pii/S1532046418301412.

[131] Newton Spolaôr et al. "ReliefF for Multi-label Feature Selection". In: *2013 Brazilian Conference on Intelligent Systems*. 2013, pp. 6–11. DOI: 10.1109/BRACIS.2013.10.

[132] Matteo Carli and Alessandro Laio. "Statistically unbiased free energy estimates from biased simulations". In: *Molecular Physics* 119.19-20 (2021), e1899323. DOI: 10.1080/00268976.2021.1899323.

[133] Shinya Honda et al. "Crystal Structure of a Ten-Amino Acid Protein". In: *Journal of the American Chemical Society* 130.46 (2008), pp. 15327–15331. DOI: 10.1021/ja8030533.

[134] Fabio Pietrucci and Alessandro Laio. "A Collective Variable for the Efficient Exploration of Protein Beta-Sheet Structures: Application to SH3 and GB1". In: *Journal of Chemical Theory and Computation* 5.9 (2009). PMID: 26616604, pp. 2197–2201. DOI: 10.1021/ct900202f.

[135] Alex Rodriguez et al. "Computing the Free Energy without Collective Variables". In: *Journal of Chemical Theory and Computation* 14.3 (2018), pp. 1206–1215. DOI: 10.1021/acs.jctc.7b00916.

[136] Keri A. McKiernan, Brooke E. Husic, and Vijay S. Pande. "Modeling the mechanism of CLN025 beta-hairpin formation". In: *Journal of Chemical Physics* 147.10 (2017). DOI: 10.1063/1.4993207. URL: https://doi.org/10.1063/1.4993207.

[137] Maria d'Errico et al. "Automatic topography of high-dimensional data sets by non-parametric density peak clustering". In: *Information Sciences* 560 (2021), pp. 476–492. DOI: 10.1016/j.ins.2021.01.010.

[138] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. "Introduction to Information Retrieval". In: Cambridge, England: Cambridge University Press, 2008. Chap. Flat Clustering, pp. 349–375.

[139] Jörg Behler. "Atom-centered symmetry functions for constructing high-dimensional neural network potentials." In: *The Journal of chemical physics* 134 7 (2011), p. 074106. URL: https://api.semanticscholar.org/CorpusID:20222855.

[140] Bingqing Cheng et al. "Ab initio thermodynamics of liquid and solid water". In: *Proceedings of the National Academy of Sciences* 116 (2018), pp. 1110–1115. URL: https://api.semanticscholar.org/CorpusID:54077466.

[141] John A. Keith et al. "Combining Machine Learning and Computational Chemistry for Predictive Insights Into Chemical Systems". In: *Chemical Reviews* 121.16 (2021). PMID: 34232033, pp. 9816–9872. DOI: 10.1021/acs.chemrev.1c00107.

[142] Andreas Singraber, Jörg Behler, and Christoph Dellago. "Library-Based LAMMPS Implementation of High-Dimensional Neural Network Potentials". In: *Journal of Chemical Theory and Computation* 15.3 (2019). PMID: 30677296, pp. 1827–1840. DOI: 10.1021/acs.jctc.8b00770. eprint: https://doi.org/10.1021/acs.jctc.8b00770. URL: https://doi.org/10.1021/acs.jctc.8b00770.

[143] Andreas Singraber et al. "Parallel Multistream Training of High-Dimensional Neural Network Potentials". In: *Journal of Chemical Theory and Computation* 15.5 (2019). PMID: 30995035, pp. 3075–3092. DOI: 10.1021/acs.jctc.8b01092. eprint: https://doi.org/10.1021/acs.jctc.8b01092. URL: https://doi.org/10.1021/acs.jctc.8b01092.

[144] Felix Musil et al. "Physics-Inspired Structural Representations for Molecules and Materials". In: *Chemical Reviews* 121.16 (Aug. 2021), pp. 9759–9815. ISSN: 0009-2665. DOI: 10.1021/acs.chemrev.1c00021. URL: https://doi.org/10.1021/acs.chemrev.1c00021.

[145] Sandip De et al. "Comparing molecules and solids across structural and alchemical space". In: *Phys. Chem. Chem. Phys.* 18 (20 2016), pp. 13754–13769. DOI: 10.1039/C6CP00415F. URL: http://dx.doi.org/10.1039/C6CP00415F.

[146] Claudio Zeni et al. "Compact atomic descriptors enable accurate predictions via linear models". In: *The Journal of Chemical Physics* 154.22 (June 2021). ISSN: 1089-7690. DOI: 10.1063/5.0052961. URL: http://dx.doi.org/10.1063/5.0052961.

[147] Claudio Zeni et al. "Exploring the robust extrapolation of high-dimensional machine learning potentials". In: *Phys. Rev. B* 105 (16 Apr. 2022), p. 165141. DOI: 10.1103/PhysRevB.105.165141. URL: https://link.aps.org/doi/10.1103/PhysRevB.105.165141.

[148] Huamin Ren et al. "Greedy vs. L1 convex optimization in sparse coding: comparative study in abnormal event detection". In: vol. 37. International Conference on Machine Learning 2015 ; Conference date: 01-06-2015. MIT Press, 2015.

[149] Marwa El Halabi and Stefanie Jegelka. *Optimal approximation for unconstrained non-submodular minimization*. 2022. arXiv: 1905.12145 [cs.LG].

[150] Jamelle Watson-Daniels et al. "Multi-Target Multiplicity: Flexibility and Fairness in Target Specification under Resource Constraints". In: *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '23. Chicago, IL, USA: Association for Computing Machinery, 2023, pp. 297–311. ISBN: 9798400701924. DOI: 10.1145/3593013.3593998. URL: https://doi.org/10.1145/3593013.3593998.

[151] Blanca Ceballos, María Teresa Lamata, and David A. Pelta. "A comparative analysis of multi-criteria decision-making methods". In: *Progress in Artificial Intelligence* 5.4 (Nov. 2016), pp. 315–322. ISSN: 2192-6360. DOI: 10.1007/s13748-016-0093-1. URL: https://doi.org/10.1007/s13748-016-0093-1.

[152] Siamak Kheybari, Fariba Mahdi Rezaie, and Hadis Farazmand. "Analytic network process: An overview of applications". In: *Applied Mathematics and Computation* 367 (2020), p. 124780. ISSN: 0096-3003. DOI: https://doi.org/10.1016/j.amc.2019.124780. URL: https://www.sciencedirect.com/science/article/pii/S0096300319307726.

[153] Smitha Milli, Luca Belli, and Moritz Hardt. "From Optimizing Engagement to Measuring Value". In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '21. Virtual Event, Canada: Association for Computing Machinery, 2021, pp. 714–722. ISBN: 9781450383097. DOI: 10.1145/3442188.3445933. URL: https://doi.org/10.1145/3442188.3445933.

[154] Laurens van der Maaten, Eric O. Postma, and Jaap van den Herik. "Dimensionality Reduction: A Comparative Review". In: 2008. URL: https://api.semanticscholar.org/CorpusID:12051918.

[155] Aldo Glielmo et al. "Unsupervised Learning Methods for Molecular Simulation Data". In: *Chemical Reviews* 121.16 (2021). PMID: 33945269, pp. 9722–9758. DOI: 10.1021/acs.chemrev.0c01195.

[156] Aurélien Bellet, Amaury Habrard, and Marc Sebban. "Metric Learning". In: *Synthesis Lectures on Artificial Intelligence and Machine Learning* 9 (Jan. 2015), pp. 1–151. DOI: 10.2200/S00626ED1V01Y201501AIM030.

[157] B. Mishra et al. "Low-Rank Optimization with Trace Norm Penalty". In: *SIAM Journal on Optimization* 23.4 (2013), pp. 2124–2149. DOI: 10.1137/110859646. eprint: https://doi.org/10.1137/110859646. URL: https://doi.org/10.1137/110859646.

[158] Ji Eun Park et al. "A systematic review reporting quality of radiomics research in neuro-oncology: toward clinical utility and quality improvement using high-dimensional imaging features". In: *BMC Cancer* 20.1 (Jan. 2020), p. 29. ISSN: 1471-2407. DOI: 10.1186/s12885-019-6504-5. URL: https://doi.org/10.1186/s12885-019-6504-5.

[159] Guray Erus et al. "Learning high-dimensional image statistics for abnormality detection on medical images". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*. 2010, pp. 139–145. DOI: 10.1109/CVPRW.2010.5543141.

[160] Hongwen Zheng and Yanxia Zhang. "Feature selection for high-dimensional data in astronomy". In: *Advances in Space Research* 41.12 (2008), pp. 1960–1964. ISSN: 0273-1177. DOI: https://doi.org/10.1016/j.asr.2007.08.033. URL: https://www.sciencedirect.com/science/article/pii/S0273117707009325.

[161] Maryam Imani and Hassan Ghassemian. "Binary coding based feature extraction in remote sensing high dimensional data". In: *Information Sciences* 342 (2016), pp. 191–208. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2016.01.032. URL: https://www.sciencedirect.com/science/article/pii/S0020025516000438.

Appendix A

Appendix

Code for Categorical Information Imbalance

```
1
2  import scipy.spatial.distance as scidist
3  from scipy.stats import rankdata
4  import warnings
5
6  def return_dist_inds_scipy(X):
7      scdist=scidist.squareform(scidist.pdist(X))
8      distranks = np.argsort(scdist, axis=1)
9      zero_dists = np.sum(np.sort(scdist, axis=1)[:, 1:] <=
           1.01 * np.finfo(np.float32).eps)
10     if zero_dists > 0:
11         warnings.warn(
12             "There are points with neighbours at 0 distance, 
                   meaning the dataset probably has identical 
                   points.\n"
13             "This can cause problems in various routines.\nWe
                    suggest to either perform smearing of 
                   distances using\n"
14             "remove_zero_dists()\n"
15             "or remove identical points using\n"
16             "remove_identical_points()).")
17
18     return distranks
19
20 def imb_cont2cat(continous_vector, binary_vector): #2D arrays
       of shape (N,1), not (N,)
21     N = continous_vector.shape[0]
```

```
22      dist_inds = return_dist_inds_scipy(continous_vector)
23      NN = dist_inds[:,1]
24
25      # calculate a correction factor each of the classes of
            discrete values
26      binary_set = set(binary_vector[:,0]) # the [:,0] because
            the input is a 2D array...
27      correction_list = []
28      for i in binary_set:
29          binary_inds = np.where(binary_vector == i)[0]
30          p_alpha = len(binary_inds)/N # class probability
31          c = 1/(N*(1-p_alpha))
32          correction_list.append(c)
33
34      count = 0
35      for i in range(N):
36          c = correction_list[np.where(np.array(list(binary_set
                )) == binary_vector[i])[0][0]] # get correction
                factor for i's class
37          if binary_vector[i] == binary_vector[int(NN[i])]:
38              count += c*0 # in the same class rank = 0
39          else:
40              count += c*N # in different classes rank = N
41      return count/(N)
42
43
44  def imb_cat2cont(binary_vector, continous_vector):    #2D
        arrays of shape (N,1), not (N,)
45      N = continous_vector.shape[0]
46      cont_NNinds = return_dist_inds_scipy(continous_vector)
47      i_sum = []
48      i_min_sum = []
49      i_term = 0
50      for i in set(binary_vector[:,0]): # the [:,0] because the
            input is a 2D array.
51          binary_inds = np.where(binary_vector == i)[0]
52          summi = 0
53          number_items = len(binary_inds)
54          for j in binary_inds:
55              for k in binary_inds:
56                  if j != k:
57                      summi += (np.where(cont_NNinds[j] == k)
                            [0][0] + 1) # add the rank of j and k
```

```
                        in same class # plus one so the ranks
                        start from 1
58          i_sum.append(summi)
59          i_term +=  2*summi/((number_items)*(number_items)) #
               this is summi/minimum_ranksum_possible
60      i_term = i_term/len(set(binary_vector[:,0])) # average
           over all classes
61      i_term = i_term/N # bring in range [0:1]
62      return i_term, i_sum, i_min_sum
```

Comparison of Classic Information Imbalance and Categorical Information Imbalance

To evaluate the behavior of the classic and continuous-predicts-categorical Information Imbalance with more and more classes, consider the example of data points from Fig. 5.1. Because the classes are non-overlapping in the continuous feature (on the y-axis), the continuous feature (x-axis) should retain high information (measured by $\Delta_{con2cat}$) about the categorical feature, even when classes are added: Indeed, Fig. A.1 shows that even with 100 classes, we still find $\Delta_{con2cat} \approx 0.25$ (blue), and with fewer than 25 classes the relationship from continuous to categorical is evaluated as highly informative.

Because the categorical features here are also distributed across the continuous feature in a ordinal (sequential) manner, also the classic Information Imbalance should capture the informativeness of the continuous feature towards the categorical. In Fig. A.1, the classic Information Imbalance (orange) performs well, when the number of categorical classes exceeds roughly $\sqrt{N} = \sqrt{500} = 22$ classes. For 5,000 data points we observe the same, that with roughly $\sqrt{5000} = 71$ classes or more the classic implementation leads to good results. When the number of classes is lower than $\sqrt{N}$, then $\Delta_{con2cat}$ performs better, even though its dedicated use case is for non-ordinal data. Therefore, the current framework allows for working with categorical-ordinal data, but the type of data and number of classes have to be carefully considered.

Future development should aim at versions of the Information Imbalance that are dedicated to handle ordinal data, and possibly even converge in a common framework with the categorical Information Imbalance.

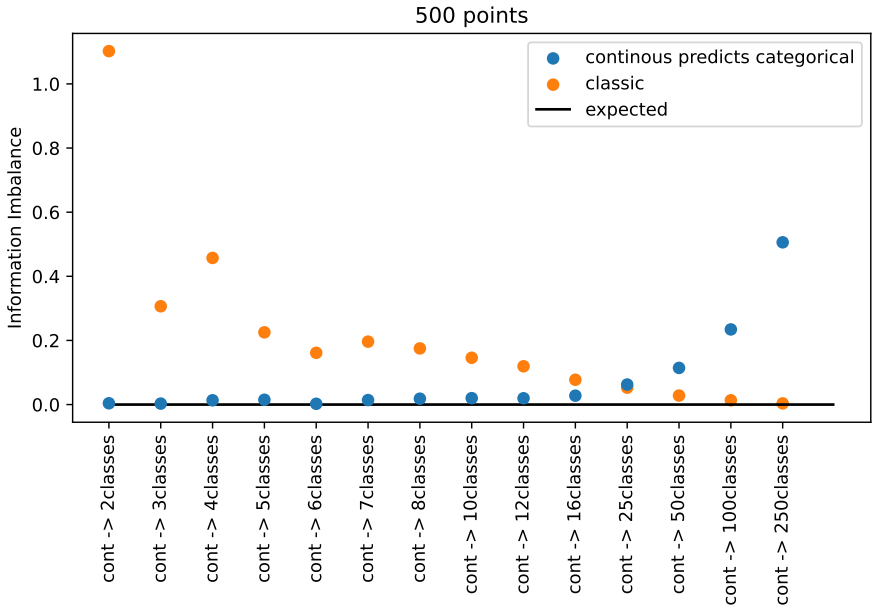


Figure A.1: The classic and continuous-to-categorical Information Imbalances for the example in Fig. 5.3 for 500 points.

Logic of the Information Imbalance from categorical to continuous features

The heart of equation 5.3 describing $\Delta_{cat2con}$ comes from the following consideration:

For the classic Information Imbalance, its core could be understood as the factor describing how much worse than the optimum (wto) the actual sum of ranks is, the factor F_{wto}:

$$F_{wto} = \frac{\sum_{i=1}^{N} \sum_{j:r_{ij}^A=1}^{N} r_{ij}^B}{N}, \tag{A.1}$$

where the numerator captures the actual count of NN ranks in space B, and the denominator, N, captures the count of NN ranks in B if B had perfect information about A.

We can define a similar factor (of being worse than the optimum) for each cluster in the categorical-predicts-continuous case. If N_α points, including i, are in the same class k in the categorical space, then in the best case, all other points in that class will have distance ranks 1 to $(N_\alpha - 1)$ from point i, equaling the sum of the arithmetic series (the Gauss sum) between ranks $(N_\alpha - 1)$ and one: $S_\alpha^{min} = \frac{N_\alpha - 1}{2}(1 + (N_\alpha - 1)) = \frac{(N_\alpha - 1)N_\alpha}{2} \approx \frac{N_\alpha^2}{2}$. This is the minimum sum of ranks

possible in the continuous space, for each point i to the points of the same class. The class specific $F_{wto}^{class-\alpha}$ is:

$$F_{wto}^{class-\alpha} = \frac{\sum_{i=1,i\in\alpha}^{N_\alpha}\sum_{j=1,j\in\alpha}^{N_\alpha} r_{ij}^B}{S_\alpha^{min}} \approx \frac{2\sum_{i=1,i\in\alpha}^{N_\alpha}\sum_{j=1,j\in\alpha}^{N_\alpha} r_{ij}^B}{N_\alpha^2} \tag{A.2}$$

Eq. A.2 can be summed over all classes and divided by the number of classes, N_{class}, and finally normalized with $1/N$ to remain in the same interval between approximating zero (fully informative) and one (A is not informative about the ranks in B).

Choosing tuples by DII

The example plotted in Fig A.2 is a L_1-search of the 285 monomials of the ten Gaussian random variables as input, with ten of them scaled as ground truth. The figure shows how several different L_1 strengths lead to the same number of non-zero features with different features and/or weights. In these cases, the lowest DIIs per numbers of non-zero features should be selected.

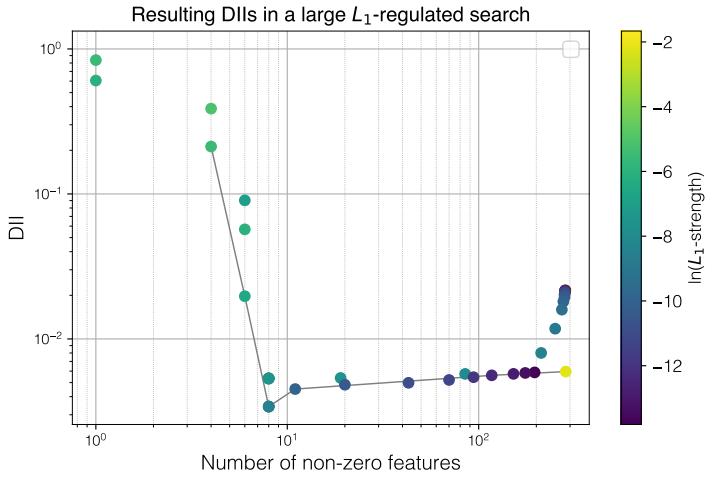


Figure A.2: The resulting DIIs for a L_1-search with many L_1 strengths plotted as a function of non-zero features on a log-log graph. Several different L_1 strengths can lead to the same number of non-zero features with different features and/or weights. The gray line corresponds to the DII represented in Fig. 1B III of the main paper.

Robustness test of DII selection features from peptide CLN025 MD trajectory

Four independent trajectory parts of the MD simulations of CLN025 are compared. The analyses show that independent parts lead to consistently chosen features (Fig. A.3) with consistent values of the DII (section on DII robustness, Fig. 6.6)

The first observation in Fig. A.3 is that the DII is not a submodular function of the number of features, meaning that the most informative single-feature, which is the anti-β-sheet content, is not part of the most informative duplets, triplets or quadruplets of features, whose main features always contain the principle components, and never the anti-β-sheet content. Secondly, the chosen features are mostly in good agreement across the four trajectory blocks. In certain instances, such as the most informative feature triplets, there are two competing features (RGYR and the PC residuals) which complement PC1 and PC2 and lead to similar DIIs. In the chosen sets of four features this conflict resolves and both of these features form part of very consistent quadruplets. Finally, in almost any feature tuple bigger than two features, PC2 is the most important feature - a non-trivial observation.

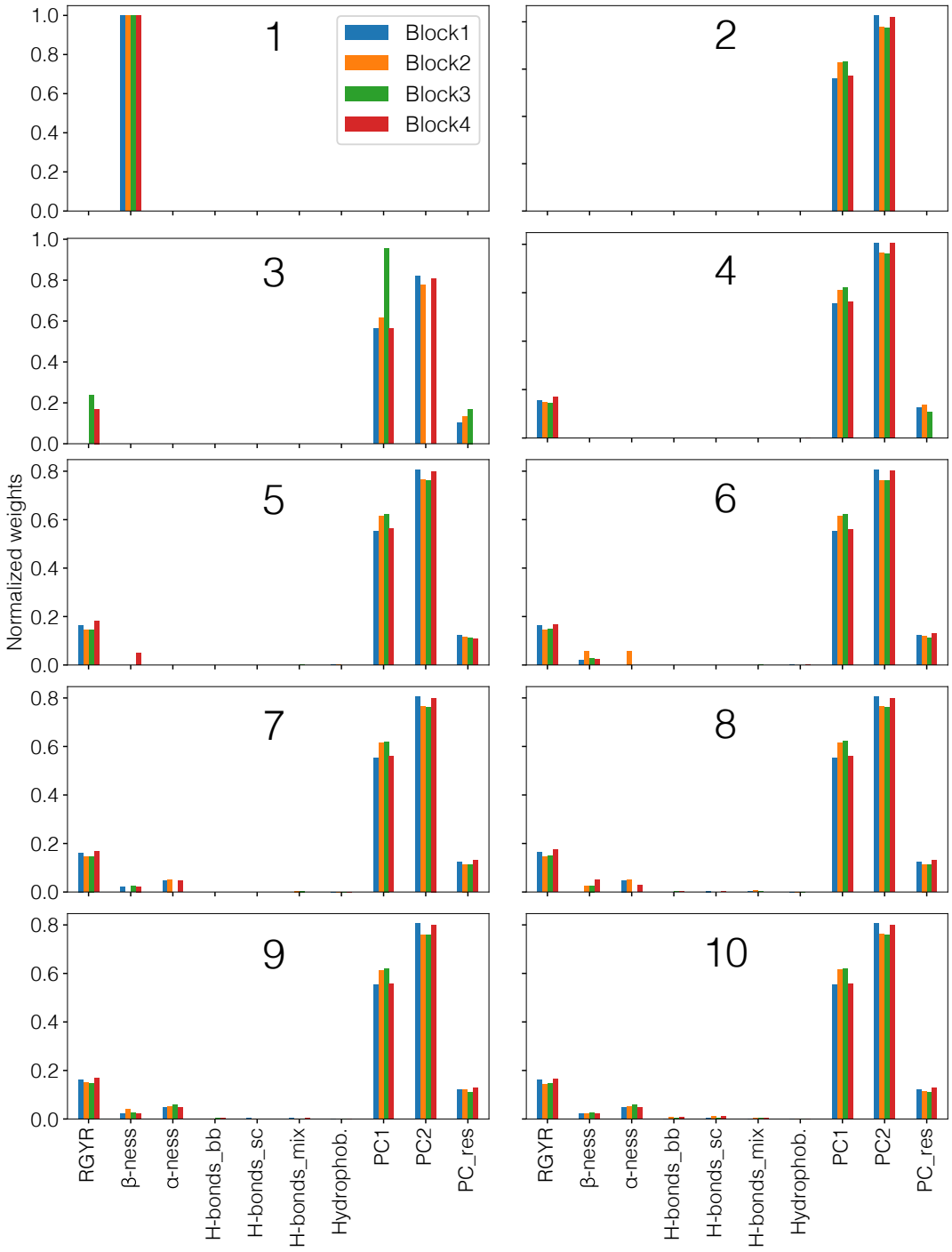


Figure A.3: The selected features and their relative weights for each number of selected non-zero features (large number in each plot) and each block, corresponding to Fig. 6.6. The ten input features are: Radius of gyration (RGYR); anti-β-sheet content (β-ness); α-helical content (α-ness); the number of hydrogen bonds in the backbone (H-bonds_bb), in the side chains (H-bonds_sc), and between the backbone and side chains (H-bonds_mix); the number of hydrophobic contacts (Hydroph.), principal component 1 (PC1); principal component 2 (PC2) and the principal component residuals (PC_res).

DII pseudocodes

We provide in Algorithm 1 a simple pseudocode for the optimization of the Differentiable Information Imbalance. The hyperparameters include the number of training epochs n_{epochs}, the starting learning rate η_0 (which is reduced with an exponential or cosine decay during the training, if `decaying_lr` is "exp" or "cos"), the strength of the L_1 regularization p and the softmax parameter λ_0. If λ_0 is not set by the user, the adaptive scheme is applied. Additionally, if no initial value $\boldsymbol{w}_0$ of the scaling weights is set, the algorithm automatically sets it to the inverse of the features' standard deviations. Similarly, if no starting learning rate is provided, the algorithm chooses a suitable one. The value of the α-component of the weight vector at epoch t is denoted by $\boldsymbol{w}_t^\alpha$ ($\alpha = 1, ..., D$).

In Algorithm 2 we describe the backward greedy approach for the search of the optimal subsets of D' features ($D' < D$). Here, the notation $\boldsymbol{w}^{(D')}$ is used to denote a weight vector with D' non-zero components, and the standard deviation of feature α is written as std^α. In each optimization, setting a component of the initial weight vector $\boldsymbol{w}_0$ to zero is equivalent to removing the corresponding feature from space A, as the derivative of the DII with respect to w^α is equal to zero throughout the entire training by setting $w_0^\alpha = 0$ (see Eq. (6) in the main text).

Algorithm 1: Pseudocode for the optimization of $DII\left(d^A(\boldsymbol{w}) \to d^B\right)$.

Parameters: n_{epochs}, η_0, $\boldsymbol{w}_0$, λ_0
Compute rank matrix in space B: r_{ij}^B;
$t = 0$; `/* epoch index */`
Compute and save starting DII: $DII_0 = DII\left(d^A(\boldsymbol{w}_0) \to d^B\right)$;
while $t < n_{\text{epochs}}$ **do**
  Compute $\boldsymbol{w}_t$-dependent distances in space A: $d_{ij}^A(\boldsymbol{w}_t)$;
  Compute softmax coefficients: $c_{ij}(\lambda, d^A(\boldsymbol{w}_t))$;
  if λ_0 **is** `None` **then**
    Compute λ_t given the current distances: $\lambda_t = \lambda_t(d_{ij}^A(\boldsymbol{w}_t))$; `/* adaptive lambda */`
  else
    $\lambda_t = \lambda_0$;
  end
  if `decaying_lr` **is** `True` **then**
    Compute η_t according to chosen schedule ; `/* set learning rate */`
  else
    $\eta_t = \eta_0$;
  end
  Compute gradient of DII: $\nabla_{\boldsymbol{w}_t} DII\left(d^A(\boldsymbol{w}_t) \to d^B\right)$;
  $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \nabla_{\boldsymbol{w}_t} DII\left(d^A(\boldsymbol{w}_t) \to d^B\right)$; `/* gradient descent step */`
  if $p \neq 0$ **then**
    for $\alpha = 1, ..., D$ **do**
      if $w_{t+1}^\alpha > 0$ **then**
        $w_{t+1}^\alpha = \max(0, w_{t+1}^\alpha - \eta_t p)$; `/* L1 regularization step */`
      else
        $w_{t+1}^\alpha = |\min(0, w_{t+1}^\alpha + \eta_t p)|$;
      end
    end
  end
  Compute and save DII with new weights:
  $DII_{t+1} = DII\left(d^A(\boldsymbol{w}_{t+1}) \to d^B\right)$;
  $t = t + 1$;
end
Result: DII_t, $\boldsymbol{w}_t$ for $t = 0, ..., n_{\text{epochs}}$

Algorithm 2: Pseudocode for the backward greedy optimization the DII.

$D' = D$; `/* number of non-zero components */`

$\boldsymbol{w}_0 = (1/\text{std}^1, 1/\text{std}^2, ..., 1/\text{std}^D)$; `/* initialize starting weight vector */`

while $D' > 0$ **do**

  Optimize $DII(\boldsymbol{w}_0)$ according to Alg. (1) and extract optimized weights $\hat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} DII$;

  Save the optimal D' non-zero weights: $\hat{\boldsymbol{w}}^{(D')} = \hat{\boldsymbol{w}}$;

  Set the smallest among the D' non-zero weights to zero: $\min \hat{w} = 0$;

  Set the new initial weight vector: $\boldsymbol{w}_0 = \hat{\boldsymbol{w}}$;

  $D' = D' - 1$; `/* reduce the dimensionality by 1 */`

end

Result: $\hat{\boldsymbol{w}}^{(D')}$ for $D' = 1, ..., D$