Biologically informed neural network for subphenotype classification in septic AKI

Erik Hartman

Master's thesis in Biomedical Engineering

Supervisors

Aaron Scott
Johan Malmström
Christian Antfolk

LUND UNIVERSITY

LTH
FACULTY OF ENGINEERING

Department of Biomedical Engineering

Contents

Abstract

Sepsis is a life threatening condition where the body's reaction to an infection results in a dysregulated immune response - ultimately causing damage to tissues and organs. The syndrome is diverse, both in underlying biology, disease manifestation and severity, and is therefore divided into endotypes and further into subphenotypes. Further understanding of the biological pathways of the various sepsis types is required in order to develop targeted diagnostic and therapeutic tools necessary to combat the disease. In this thesis, the plasma proteome of patients suffering from two subphenotypes of septic acute kidney injury with varying severity were analyzed. The proteomic data was combined with the Reactome pathway database, and leveraged to generate and train a biologically informed neural network in classifying the two subphenotypes. The network was able to distinguish between the subphenotypes, achieving an accuracy of $98.2 \pm 0.02\%$ when created with four hidden layers. The informed nature of the network allows for introspection into the network's decision making - allowing us to utilize feature importance values to interpret which proteins and biological pathways the network deemed important for classification. Ultimately, this identified several biomarkers for the subphenotypes including apolipoproteins, histones and known inflammatory markers such as CD14 and osteopontin. The algorithm generating the biologically informed network was generalized and is publicly available as a Python package: https://github.com/InfectionMedicineProteomics/BINN.

Lay summary

Utilizing machine learning to understand sepsis

Sepsis is one of the deadliest syndromes in modern time, with little to no effective therapies available. In this project, machine learning was utilized to gain insight into the underlying biology of sepsis - a necessary step in finding novel and effective treatments and diagnostic tools.

Sepsis is a syndrome (collection of symptoms) which is responsible for $\sim$ 20% of global deaths each year. It is extremely diverse and complex, rendering it difficult to both diagnose and treat. Recently, researchers have classified various types of sepsis and recognized that unique therapies are required for the different types. However, before creating treatments and diagnostic tools, we need to understand the underlying biology of the various types - which is easier said than done.

In this project, machine learning was utilized to understand the biology of a specific type of sepsis which is characterized by damage to the kidney (referred to as septic acute kidney injury or AKI). Machine learning is a way to make a machine find patterns in complex data - and sepsis as a disease can be seen as a complex mixture of biological molecules. Finding the pattern in this soup of molecules is therefore a task fit for machine learning, and could help us understand the disease.

An algorithm named a *biologically informed neural network* - that is: a machine learning algorithm (specifically a neural network) that reflects the underlying biology of the disease (hence biologically informed) was devised. Creating such an algorithm solves the black box problem in machine learning, which states that it is impossible

to understand what a machine learning algorithm is doing when it is solving a problem. It also allows for introspection into the algorithm and understand what parts of the biology it finds important and interesting when analyzing specific types of sepsis.

The algorithm allowed for the finding of proteins and biological pathways which were important in classification of septic acute injury of different severity. It was generalized to be compatible with any type of condition, disease or syndrome. Further, the algorithm is available in a public repository, and anyone can now create a biologically informed neural network with one line of code. Therefore, it can be utilized in other experiments to not only progress in the development of treatments and diagnostics of sepsis - but also other diseases.

Preface

This thesis project has been conducted at the Biomedical Center at Lund University - specifically at the infection medicine proteomics lab headed by Prof. Johan Malmström. I would like to thank everyone at BMC D13 for their support during the entirety of my master thesis project. Additionally, I'd like to thank Aaron Scott and Johan Malmström for their patience, insight and encouragement - without which this project wouldn't have been possible.

1 Background

1.1 Sepsis

Sepsis is a life threatening condition where the body's response to an infection results in injury to tissues and organs, formally defined by the Third International Consensus as: *"organ dysfunction caused by a dysregulated host response to infection"* [1]. In 2017, 49.8 million cases of sepsis resulted in 11 million deaths worldwide - representing 19.7% of global deaths [2]. That same year, sepsis was recognized as a global health priority by the WHO, urging member states to action, with a priority goal of *"developing national policy and processes to improve the prevention, diagnosis, and treatment of sepsis"* [3]. International collaborations, such as the Global Sepsis Alliance [4] and the European Sepsis Alliance [5], are funding research and development in order to combat the syndrome. This united effort has resulted in accelerated data gathering and research in the area of sepsis diagnostics and prevention.

During sepsis, pathogen associated molecular pattern-derived or damage-associated molecular patterns (DAMPs) initiate an excessive inflammatory response after binding to receptors such as toll-like receptors (TLRs) on immune cells, resulting in the upregulation of both inflammatory and anti-inflammatory pathways [6]. The pathophysiology of sepsis is the result of a complex interaction between the various parts of these inflammatory pathways and pathological molecules, leading to a multifaceted disruption of the regulation of the immune system, which, in a healthy state, is finely tuned [7]. The host response to sepsis varies greatly and may result in symptoms ranging from mild to very severe, including coagulopathy (impaired blood-clotting) [8], acute respiratory distress syndrome, [9], acute kidney injury [10], septic shock and death.

The varied disease manifestations resulted in a weak consensus on the definition of sepsis, which led to the differentiation of distinct sepsis endotypes, largely defined by gene expression and distinct biological pathways [11]. Furthermore, the endotypes can be divided into subphenotypes of varying severity and manifestations [12], [13]. Analysis of the various types of sepsis has motivated large cohort studies, where stratification using biological and clinical markers have been successful in early discrimination of some sepsis endotypes and subphenotypes. One such study is the FINNAKI study, a prospective observational study including 2901 patients where the incidence, risk factors and outcome of AKI were monitored [14]. AKI is characterized by a reduced glomerular filtration rate (GFR), leading to fluid and electrolyte-imbalances and is accompanied by a high mortality rate ($\sim 40\%$) [14]. Two subphenotypes of AKI of varying severity have been identified in the FINNAKI cohort by latent class analysis based on comorbidities, clinical data and biomarkers. The more severe subphenotype was found to be characterized by an increase in inflammatory and endothelial injury markers and associated with a lower chance of renal recovery and increased mortality [15].

Advances in areas such as transcriptomics, metabolomics and proteomics have yielded insight into some of the mechanisms underlying sepsis and the different endotypes, although the complexity of the syndrome leaves much to be wanted in terms of diagnostics, treatment and further understanding of the mechanisms of disease. Currently, there are no targeted therapies for sepsis, and further understanding of endotype-specific therapeutic targets and biological pathways is needed to ensure the success of future clinical trials [16]. This motivates the development and implementation of novel methodologies which are capable of incorporating vast amounts of information to unravel the complexity of the biological pathways involved in the pathogenesis of sepsis. Approaches utilizing machine learning have emerged as good candidates in similar fields of research due to their ability to capture complex patterns in high dimensional data, making them suitable for the analysis of biological systems.

1.2 Machine learning

Machine learning is a set of methods whereby data is leveraged to tune the predictive performance of a model on certain tasks and is commonly seen as a subset of artificial intelligence. The term “learning” is regularly mistakenly associated with the human-like characteristic of acquiring general and transferable knowledge, suggesting that the field of machine learning is set out to create human-like machines with human-like intelligence. This is not the case, as machine learning is simply a set of algorithms that optimize performance by tuning parameters and does so by minimizing *loss* defined by some *loss function*. The core goal of a machine learning model is to, based on some set of data drawn from an unknown probability distribution, build a model of the space of occurrences that is accurate enough to be able to predict the outcome of new occurrences. The model’s ability to correctly identify new data points is central, and corresponds to a suitable complexity of the hypothesis proposed based on the given data. Often, the high dimensionality of the space and the comparatively smaller number of samples makes the problem of generating a suitable hypothesis difficult (often referred to as the *curse of dimensionality*). The process of generating a hypothesis from a given set of data points is often referred to as *training* the model [17].

A data point is represented as a vector of *features*, the nature of which highly influences the performance of the model. Poorly chosen or expressed features may result in a poor representation of the data, and a large part of the workflow when working with machine learning models is dedicated to feature selection and feature engineering to counter this [18]. Common practices in feature engineering include normalization, encoding, scaling and dimensionality reduction. A feature vector can be viewed as a point in feature space, which represents all possible combinations of features.

Machine learning methods are typically divided into three categories based on the nature of the model's inputs and outputs: *supervised learning, unsupervised learning* and *reinforcement learning.* This work mainly utilizes methods of supervised learning, where the training input consists of both example inputs and desired outputs (also referred to as *label*), and the goal of the model is to generate a rule which maps the input to the appropriate output. In unsupervised learning there are no labels, and only the features of a data point are known. A common method of unsupervised learning is data clustering where unlabeled data points are labeled based on some metric (often euclidean distance) and method (such as Ward minimum variance method [19]).

Although current machine learning methods show no signs of acquiring anything like human intelligence, novel techniques and advances in model architecture do seem to blur the boundaries between human-like intelligence and elementary predictive performances [20]. One may therefore divide machine learning techniques into classical machine learning algorithms, and those characterized by the use of modern techniques such as *neural networks* (often denoted as *deep learning*).

1.2.1 Classical methods

Classical machine learning algorithms were introduced in the 1950's and are mostly based on statistics and probabilistic reasoning [21]. There are several classical ML methods which achieve high accuracy while being simple, making them attractive for numerous applications.

Support vector machines

A support vector machine (SVM) is a robust non-probabilistic binary classifier, which separates the classes by dividing the feature-space with a hyperplane. New occurrences are mapped onto the space and classified based on the spatial situation relative to the hyperplane. The goal is to find the hyperplane which maximizes the distance between the hyperplane and the points of both classes [22]. There are various kernels which may be used for non-linear classifications, the most used one being the radial basis function (RBF) [23].[1]

A hyperplane, H, can be written as:

$$H : \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} - b = 0$$

where $\boldsymbol{w}$ is the normal vector to the hyperplane.

If we have two classes with labels $(1, -1)$, which are *linearly separable*, we can select two parallel hyperplanes which separate these classes so that the distance between these hyperplanes is maximized. The two hyperplanes can be written as:

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} - b = 1$$

and

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x} - b = -1$$

The distance between the planes are $2/||\boldsymbol{w}||$, so maximizing this distance can be achieved by minimizing $\boldsymbol{w}$, while keeping data-points on the correct side of the margin. This results in the optimiation problem of minimizing $||\boldsymbol{w}||$ subject to $y_i(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} - b) \geq 1$.

[1] The Iris flower dataset https://en.wikipedia.org/wiki/Iris_flower_data_set was used for all classical ML visualizations.

To separate classes that are not completely linearly separable (which is often the case), we define the *hinge loss function*:

$$max(0, 1 - y_i(\boldsymbol{w^T}\boldsymbol{x}_i - b))$$

This function is zero if a point x_i lies on the correct side of the margin, and proportional to the distance if on the wrong size of the margin. Large errors are therefore penalized greatly. We now get a new optimization problem, where the goal is to minimize:

$$\lambda||\boldsymbol{w}||^2 + (\frac{1}{n}\sum_{i=1}^{n} max(0, 1 - y_i(\boldsymbol{w^T}\boldsymbol{x}_i - b)))$$

where λ is a trade-off parameter between the margin-size and correct classifications.

Tree based algorithms

Tree-based algorithms use decision trees where features are represented as nodes in a tree-like structure. The connections (branches) between the nodes are based on different sets of feature values. Decisions are represented as leaves, i.e., terminal nodes. When making a classification, the tree is traversed according to which set each feature belongs to. Eventually, a leaf is reached, and a classification is made. Generating a tree entails creating a tree-architecture where data is separated based on feature cut-offs which are deemed to stratify the data optimally. To decide which order to place nodes, and which feature and cut-off to subset the data on, *entropy* (disorder) is used to calculate the *information gain*.

Entropy, E, of a state, S, is defined as:

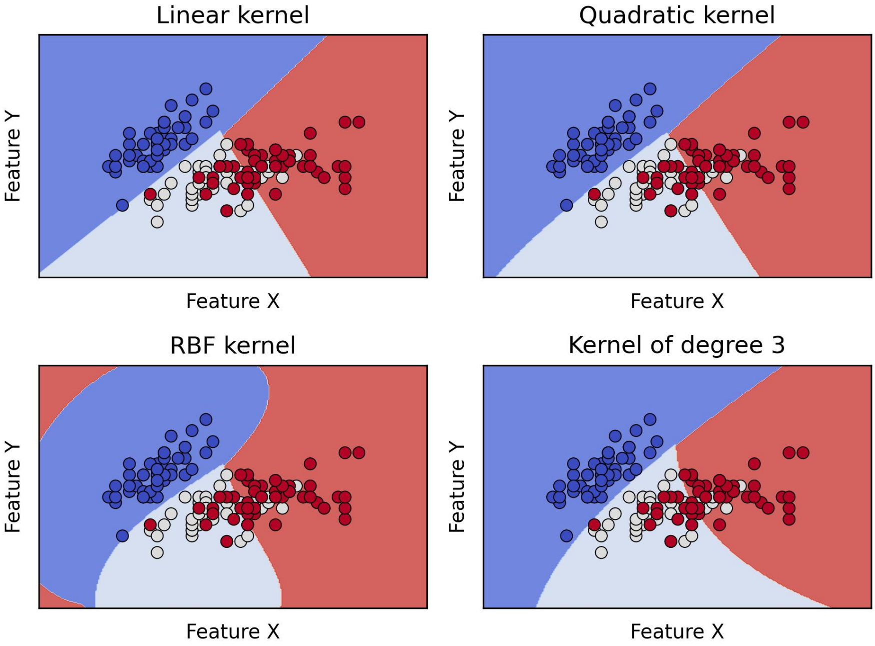

Figure 1.1: Fitting an SVM to data consisting of three classes (red, blue, white) with different kernels results in different divisions of the feature-space. The linear kernel is limited to linear divisions of the space, whereas the radial basis function (RBF) kernel can divide the space to better fit the data.

$$E(S) = \sum_{i=1}^{n} -p_i log_2 p_i$$

where p_i is the probability of an event of state S. We can calculate the entropy of multiple features:

$$E(S, X) = \sum_{c \in X} P(c)E(c)$$

where X is the selected feature. Thereafter, the information gain, I, can be calculated as a decrease in entropy after splitting the data on the given feature value:

$$I(S, X) = E(S) - E(S, X)$$

There are several modern and widely used implementations of tree-based learning algorithms such as random forests [24] and they all differ slightly. Furthermore, tree-based models are often used in *boosting* - where "weak learners" (algorithms that only do slightly better than random) are used in ensembles to generate a good prediction model. Gradient boosters utilize gradient descent to generate weak learners that complement each other well. XGBoost [25] and LightGBM [26] are examples of efficient gradient boosted machines utilizing tree-based models.

k-nearest neighbours

The k-nearest neighbors (k-NN) algorithm is a theoretically simple algorithm, whereby points are classified in the feature space based on their proximity to other labeled data points. Training constitutes mapping data to the feature space, and classification is conducted by counting which label is most frequent amongst the k nearest neighboring data points [27].

Let d be some distance metric (e.g., Euclidian) and k a defined positive integer. The algorithm of classifying a data-point, x, with k-NN can then summarized as follows:

1. $D \leftarrow d(x, x_i)$ for $i = 1, ..., n$
2. Sort D in increasing order.
3. $D \leftarrow D(1, ..., k)$
4. Let K_i define the number of data-points belonging to class i among D.

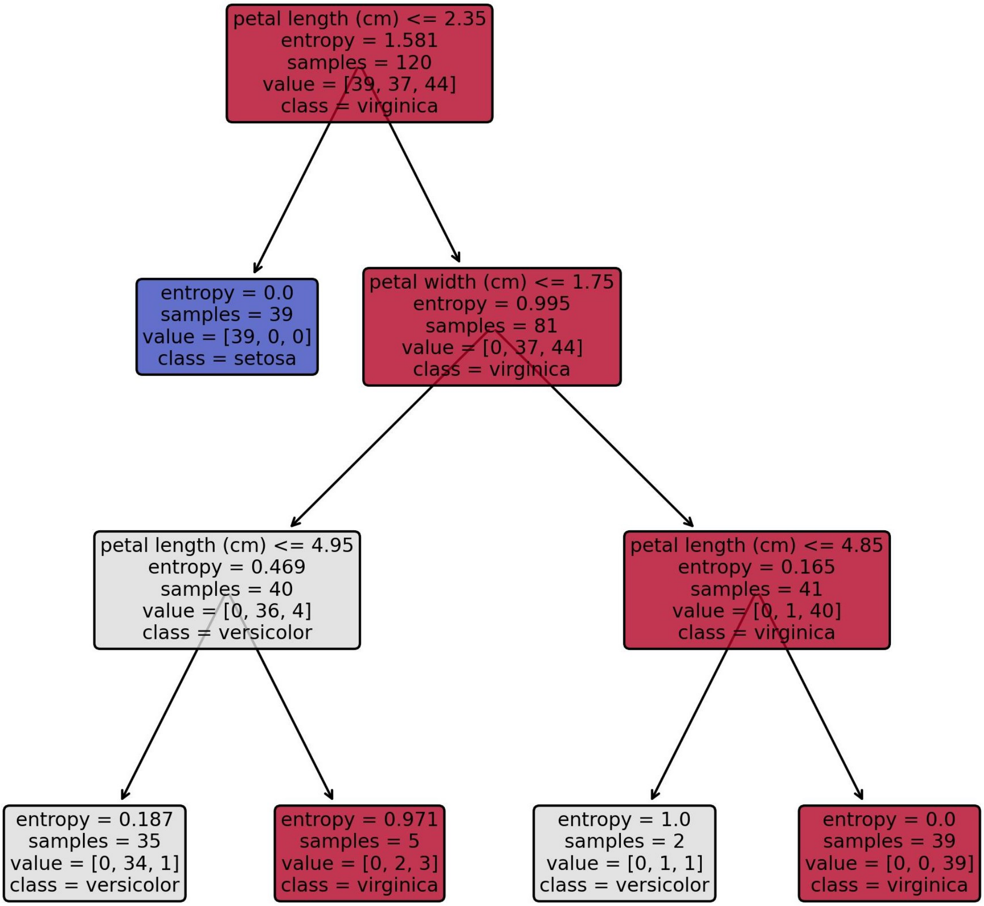

Figure 1.2: A trained decision tree where leaf nodes are colored according to the classification of that node. Each node subsets the samples based on features of the dataset. Traversing the tree (starting at the top) classifies a given data-point.

5. Assign x to class $max(K_i)$

1.2.2 Neural networks

The fundamental concept behind neural networks was identified by Donald Hebb in 1949, who proposed that neuronal connections strengthen with use: *"Cells that fire together, wire together"* [28].

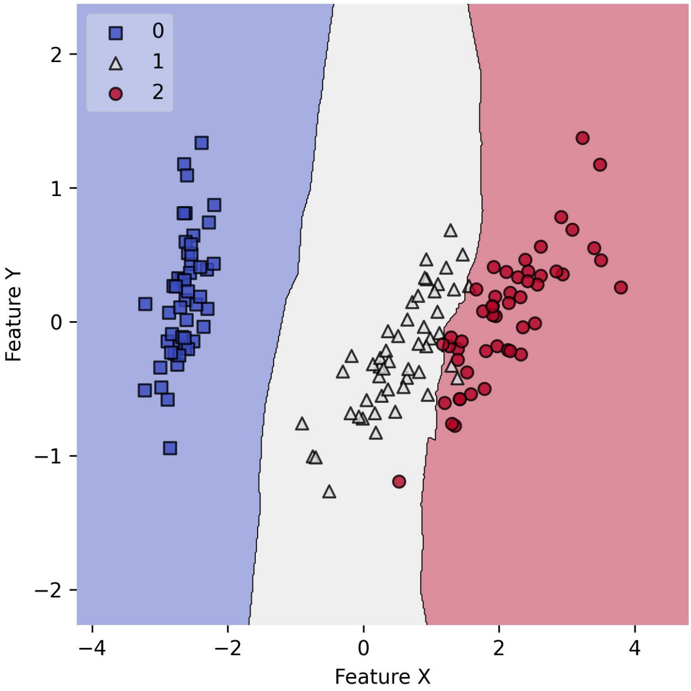

Figure 1.3: The decision boundaries of a k-NN algorithm. Three classes (red, white, blue) are placed onto the feature-space. The decision boundaries specify what a new data-point would be classified as if placed in the feature space at a given position.

The first computational Hebbian network was created in 1954, resulting in the advent of machine learning with neural networks [29]. A typical neural network is characterized by connected nodes (artificial neurons) arranged in layers, where real values are transmitted as signals. The connective edge (synapse) between nodes applies a weight (multiplicative factor) and a bias (additive factor) to the signal, thus applying a linear transformation to the input. These parameters are altered when training the neural network. Nodes then apply some function (activation function) to compute its output. Performing sequential transformations to the input through several layers results in an output, which may e.g., be a classification decision.

$$y = f(\sum_i x_i w_i + b)$$

where y - output, f - activation function, x - feature vector, w - weight and b - bias.

Training the neural network means tuning the weights and biases of the network so that a given input results in the wanted output when passed through the network. The process of tuning the parameters is done through back-propagation [30]. In back-propagation, the gradient of the loss function with respect to each weight is calculated. The gradients can then be used to optimize the network as we would like to tune the weights in the opposite direction of the gradient, since this represents the steepest downward direction of the loss landscape, in which we seek to find the minimum. The algorithm used to tune the weights using the gradients is referred to as the optimizer.

The typical architecture of a neural network consists of an input layer, several hidden (intermediary) layers, followed by an output layer with dense connections (every node in a given layer is connected to every node in the subsequent layer). However, many variations of this architecture exist and may include skip connections (as in a ResNet) [31], or convolutional layers as in most neural networks designed for image analysis-related tasks [32]. The combinatorial nature of several densely connected hidden layers leads to a large number of trainable parameters as exemplified in the language model GPT-3 which contains 175 billion parameters [33], or in BLOOM containing 176 billion trainable parameters. The large number of parameters make the networks timely and resource-intensive to train.

Neural networks have outperformed most of the classical ML techniques and are in use in many applications, prominently in visual classification tasks such as autonomous driving. However, they suffer from a lacking interpretability which hinders their use in areas such as health care due to a lack of trust from both patients and officials. The set of parameters in a trained model aren't understandable by humans and often fail to reflect real phenomena, and large neural networks are therefore often denoted as *black boxes*. Furthermore, several neural networks have faced criticism due to their focus on unimportant features and their feebleness when faced with obstructed or noisy data, making them unsuitable for several real-world applications [34].

1.3 Explainable artificial intelligence and Shapley values

In an attempt to illuminate these "black box" models, a new subfield focusing on the interpretability of artificial intelligence has come forth - explainable AI (XAI). XAI includes methods that enable the interpretation of models so that they may be more readily implemented in real-world scenarios. The understanding of a model's reasoning when generating an output may also help gain insight into solutions of the problem which it is set out to solve by e.g., highlighting key features or decisions made when making a classification [35].

Knowing which features are deemed "important" for a model's decision-making is necessary when interpreting it. From a game theoretic perspective, one may consider the feature importance as the marginal contribution of the feature (player), after considering each combination of features (coalitions) when making a correct classification (desired outcome). It can be shown that the set of marginal contributions with certain desirable properties is unique. This set of solutions is called the Shapley values and is named after Lloyd Shapley, who won the Nobel Prize in economics for the concept in 2012. Shapley values are expensive to calculate, since one has to average the contribution of every player across all different coalitions of players, resulting in 2^n combinations. In practice, the values are estimated using a subset of all possible coalitions.

Variants of Shapley values have been introduced and applied to ML models, one being Shapley Additive Explanations (SHAP) [36]. In SHAP we want to simplify our model f using a simplified *explanation* model g. We simplify our feature vector x to x', and introduce a mapping function $h_x(x') = x$. SHAP uses the linear explanation model:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i$$

where z is a *coalition vector* $\in \{0,1\}^M$ and ϕ_i the feature importance for feature x_i. Three properties (*local accuracy, missingness* and *consistency*) are given to define ϕ:

$$\mathbf{1:}\; f(x) = g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i x'_i$$

$$\mathbf{2:}\; x'_i = 0 \Rightarrow \phi_i = 0$$

$$\mathbf{3:}\; f'_x(z') - f'_x(z'i) \geq f_x(z') - f_x(z'i) \Rightarrow \phi_i(f', x) \geq \phi_i(f, x)$$

Then, ϕ is uniquely defined and are the same as the Shapley values. Although other feature attribution methods exist, such as LIME [37] and DeepLIFT [38], SHAP provides a unified framework applicable to various ML models with greater correspondence to human intuition than contemporary methods.

1.4 Mass spectrometry protemoics

The ability to study the proteomic content of samples is realized by the *mass spectrometer* (MS) [39]. MS is an instrument that measures the mass to charge ratio (m/z) of molecules in a sample. The ratio, alongside spectral libraries, and alternatively some prior knowledge about the content of the sample, is used to infer the constituents of the molecules and thereby identify them. There are several workflows for identifying the content of biological samples depending on the nature of the sample and the experiment. However, all bottom-up proteomic workflows begin with *digestion* - where the proteins in a biological sample are digested by sequence-specific or unspecific proteases (commonly trypsin). The resulting peptides are separated through liquid chromatography and then ionized (various ionization techniques exist such as electrospray ionization (ESI) and matrix assisted laser desorption ionization (MALDI). The ionized peptides are then separated by their m/z using an electric or magnetic field. In order to identify these peptides, another MS is performed in tandem (MS/MS), where peptides are fragmented and their resulting fragments ions m/z are analyzed. The second fragmentation and separation enable the distinction between peptides of similar m/z. The resulting spectra is then analyzed by some software, and the peptide content quantified by integrating the precursor-ions signal peaks [39].

MS/MS-workflows are commonly grouped into: *data-dependent acquisition* (DDA), *data-independent acquisition* (DIA) and *selected reaction monitoring* (SRM) where DDA can be used for *discovery proteomics* methods, and SRM is a *targeted method*. DIA falls in between the two methods, and is often used for targeted methods but may also be used in discovery proteomics. In targeted methods, there is a search for a set of predetermined molecules in the sample, and the goal is to quantify these [40]. In discovery proteomics, the search is unbiased, and the goal is to discover and quantify all proteins that are present in the sample. Although DDA and DIA are both used in unbiased discovery proteomics, they differ in several aspects. In DDA, certain selected intense peptides from the first MS survey scan are analyzed by MS/MS, whereas in DIA, the mass range is divided into mass windows, in which all precursors in a given mass window are subjected to MS/MS. Multiple windows are sequentially employed until the complete mass spectra is analyzed. The cycle time and window-size need to be tuned to achieve good quantification[41]. Since many peptides are fragmented together in an m/z-window, the resulting MS/MS spectra in DIA analyses are complex, and require computationally expensive deconvolution. Therefore, a targeted DIA approach is utilized alongside a spectral library generated by DDA. This allows for the superior quantification and peptide discovery DIA entails while reducing the complexity of the deconvolution [42].

1.5 Machine learning and sepsis

ML has been applied in means of finding novel tools for early diagnostics and treatments of sepsis [43]. Most of the methods for early sepsis predictions utilize electronic medical records (EMR), which largely consist of unstructured data such as clinical notes or images, but also include data such as vitals, patient data and investigative metrics (e.g., white blood cell count) [44]. Some of these models, such as SERA [44], outperform physicians in predicting sepsis risk and may act as support to clinical decision making. However, there is still an aversion towards their implementation in a hospital setting due to both a lack of trust and clinical studies needed to assess patient relevant outcomes [45]. Although the proteomic profile of sepsis has been analyzed [46], [47], [48], proteomics data remains underutilized in classifying and stratifying sepsis, and the proteomic profile of sepsis endotypes and subphenotypes is poorly understood.

1.6 Reactome

DIA and DDA MS/MS results in vast proteomic datasets, containing tens of thousands of peptides mapping to several thousand proteins. Interpreting the data is often difficult, but facilitated by databases and tools such as UniProt [49], Ensembl [50] and Reactome [51]. The Reactome pathway database is central to this project, and contains molecular details of biological processes, where proteins are linked to molecular function and their physiological context. Each subsequent level in a pathway can be seen as a further abstraction of the previous, finally resulting in high-level categories such as “Disease”, “Apoptosis”, “Signaling” and “Metabolism”.

1.7 Biologically informed neural network

The idea of using the Reactome pathway database to generate a *biologically informed neural network* (BINN) was realized by Van Allen et al. [52]. They introduced a method to linearize the human Reactome graph, where-after it could be converted to a neural network. Their BINN utilized gene-related data to stratify prostate cancer, after which the trained network was interpreted. Since each node and connection is annotated the network is easily explainable, and they were able to find, and verify *in vivo*, novel molecular alterations which were important in predicting advanced disease.

2 Introduction

Sepsis is a diverse and complex syndrome associated with detrimental outcomes. There is a great need for targeted therapeutics and methods of early diagnosis, which are unavailable today. It has become apparent that the path towards effective diagnostic tools and targeted therapeutics of sepsis require: 1. a distinction and deeper understanding of the various endotypes and subphenotypes, and 2. an approach that incorporates the vast amount of information necessary to capture and untangle the complexity of the different types [16]. The ability to capture a large portion of the proteomic environment in tissues using a mass spectrometer allows for the analysis of large quantities of biological data. However, many contemporary methods utilizing proteomic data select few individual proteins based on their level of differential expression, and subject them to further studies. This low-throughput approach fails to incorporate all the available data, and therefore may miss important factors, the additive effect of which are bound to have implications on their study. Furthermore, this type of experiment fails to capture the overarching structure which constitutes biological pathways and processes, which are key in understanding a given condition.

The goal of this thesis was to implement a data-driven approach to investigate how the complete proteomic profiles of two subphenotypes of septic AKI differ - and whether their profiles convey information about the underlying biology of the two conditions. Prior to computational analysis, a proteomic dataset was generated by analyzing the blood plasma of patients suffering from septic AKI with DIA mass spectrometry, and stratified to the two subphenotypes [15]. The informatic analysis is divided into three main methodologies: firstly, a rudimentary analysis of the dataset and the differentially expressed precursors and proteins was performed. Thereafter, general machine learning methods were utilized to stratify the subphenotypes, showcasing that it indeed is viable to use proteomic data for subphenotype stratification in a machine learning setting.

The main emphasis of this thesis lies in the creation and interpretation of a biologically informed neural network. We present an algorithm that allows for the generation of a sparse network given an input dataset (in this case the proteomes) and a directed graph (in this case the Reactome pathway database [51]), in which the connections are defined by the given graph. This generates a completely annotated network, which allows for introspection. Furthermore, SHAP values were used to interpret the network by estimating the feature importance among its nodes and layers. Thereby, insight was gained into which biological phenomena are reflected in the proteomic data and are important for the classification. This workflow utilizes a data-driven deep learning approach to gain insight into the underlying biology of the given condition. The algorithm was generalized and is applicable to any type of disease, syndrome or condition, and is packaged and available in a public repository: https://github.com/InfectionMedicineProteomics/BINN.

3 Methods

The ultimate goal of this project is to utilize a data-driven machine learning approach to stratify two subphenotypes of septic AKI from proteomic data. To do so, the general characteristics of the data was analyzed and the dataset subject to classification using classical ML methods. Thereafter, a generalized algorithm of creating a biologically informed neural network was created, and used to create a sparse, informed neural network. The network was then introspected and interpreted, after which insights gained were used to cluster the dataset. The methodology is therefore presented according to the following structure.

1. Firstly, the dataset is presented (section 3.1).

2. Thereafter, the data is processed and its general characteristics investigated (section 3.2).

3. Thirdly, the data is subjected to classification with classical ML methods (section 3.3).

4. The BINN is then generated, evaluated and interpreted (section 3.4).

5. Lastly, clustering was performed on the dataset utilizing insights gained from the BINN (section 3.5).

3.1 Dataset and acquisition

The dataset used in this thesis is derived from the FINNAKI study [14] where patients with sepsis were screened for acute kidney injury (AKI). Two subphenotypes (henceforth referred to as subphenotype 1 and 2) of AKI with different clinical outcomes have been identified in the dataset, which were used as classification labels [15]. The two subphenotypes differ in severity, where subphenotype 2 is more severe - resulting in lower probability of renal recovery and an increase in mortality. The final dataset is a combination of two datasets: a longitudinal dataset, where plasma samples were taken from 23 patients at 5 different timepoints, and a subtypes dataset where the subtypes were determined for 141 plasma samples. In total there are 263 unique samples. Out of those, 197 could be stratified to one of the two subphenotypes (subphenotype 1: 74, subphenotype 2: 123) and 66 were unclassified (figure 3.1A). The two datasets were separately processed with the same OpenSWATH [42] workflow. The longitudinal dataset contained less peptides (figure 3.1B) and proteins than the subtype dataset (subtypes: 1090 proteins, longitudinal: 604 proteins. The final dataset was a result of the intersection between the two datasets and contains 563 unique proteins (figure 3.1C). The discarded proteins generally contained few peptides and a higher Q-value (false discovery rate-adjusted p-value) than included proteins (figure 3.1D).

3.2 Data processing and general characteristics

Prior to merging the datasets, the data is filtered and normalized separately. The peptide lists were filtered on Q-value to keep peptides with Q-value ≤ 0.01. Data was normalized using the sample mean, and by using a retention time-mean sliding window filter. Thereafter, the data can be merged to a single peptide list. When

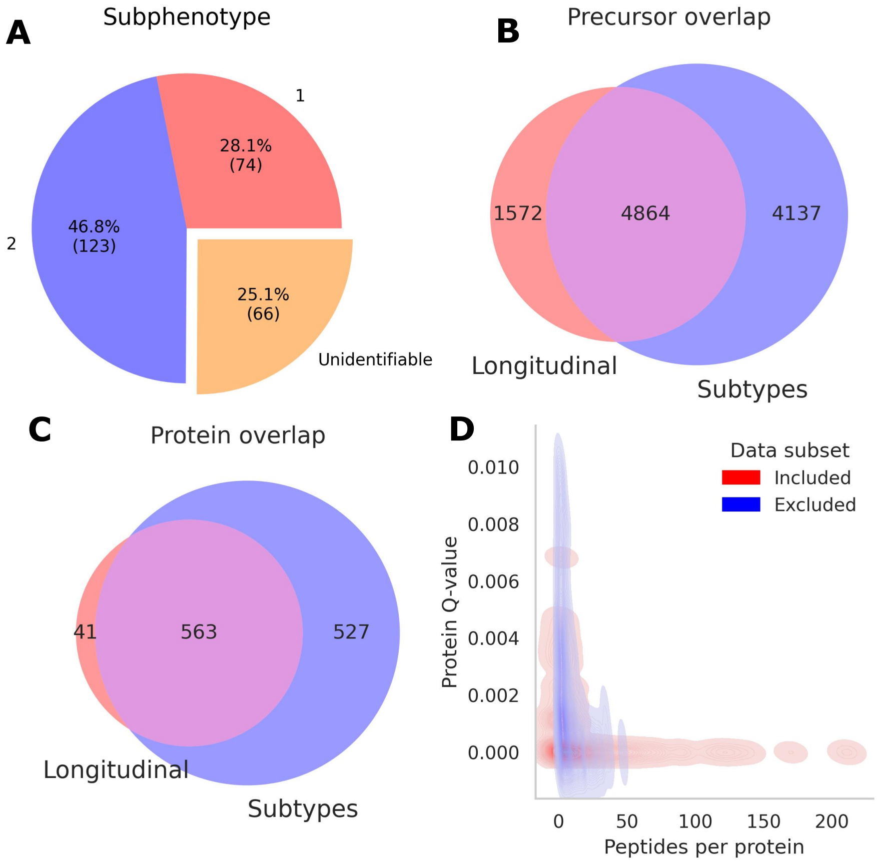

Figure 3.1: A) Pie-chart of number of subtypes. The final dataset is the intersection of two datasets: *longitudinal* and *subtypes*. There are 197 labeled samples, out of which 74 are of subphenotype 1 and 123 of subphenotype 2. B) Venn-diagram of number of the precursor overlap in the two datasets. 75% of the precursors in the longitudinal dataset is included by the subtypes dataset. C) Venn-diagram of the protein-overlap in the two datasets. 93% of the proteins in the longitudinal dataset is included in the subtypes dataset. D) Kernel density estimation (Gaussian kernel) of the protein Q-value and the number of peptider per protein. The 568 proteins which are discarded from the dataset generally have few peptides and a higher Q-value than the included proteins.

merging, the retention time was kept from the subtypes dataset. Protein abundances were quantified using the MaxLFQ-method, in which pair-wise peptide and protein ratios are used to determine the protein quantity in each sample [53]. This results in a protein matrix, containing the protein abundance for each protein in each sample. The matrix is scaled by removing the mean and scaling to unit variance:

$$z = \frac{x - \mu}{s}$$

The transformed protein abundances were used as features for the coming machine learning applications.

The general characteristics of the dataset were analyzed in various ways. Firstly, differential expressions on precursor and protein level were evaluated using linear regression. In differential expression with linear regression, lines are fitted to the data points and the differences between the parameters of the resulting linear functions between groups are statistically evaluated. Methods from the in-house DPKS-package [1] were used for quantization, normalization and differential expression.

When summarized absolute measurements were evaluated, such as the number of peptides per protein in each group, values were corrected by the imbalance in number of samples per group.

3.3 Classical machine learning

Four classifiers: SVM (RBF-kernel), k-NN, RF, XGBoost, LightGBM, were trained using k-fold cross validation ($k = 5$). In k-fold cross validation, the data is divided into k subsets. During each fold, subset n is used for validation and all other k-1 subsets used for training. After each fold, n is incremented until all the folds have been used for validation. This technique utilizes the complete

[1] https://github.com/InfectionMedicineProteomics/DPKS

dataset, and allows for the use of statistical measures for inter-model comparisons, as k models may be trained and evaluated. The accuracy, sensitivity, and specificity of the various methods were evaluated using the area under the receiver operator characteristic (ROC) curve (AUC) and true positive/true negative rates.

3.4 Biologically informed neural network (BINN)

The following subsections describes how the BINN is generated (section 3.4.1), evaluated (section 3.4.2) and interpreted (section 3.4.3).

3.4.1 Generating the BINN

The BINN architecture is automatically generated from the union of all the proteins present in the proteomic dataset and the Reactome pathway database. An example of a path in the Reactome pathway database can be seen in figure 3.2.

The BINN is generated using the following algorithm:

1. Subset the Reactome pathways database (directed graph) using the union of proteins by recursively adding the parental pathway, starting at the protein level, until the highest level of nodes is reached.

2. Generate a network from the subsetted pathways and add an output node connected to the highest level of nodes. The number of output nodes correspond to the number of classes the network is set to predict, in our case 2 for the subphenotypes.

3. Starting at the output node, traverse the network backwards

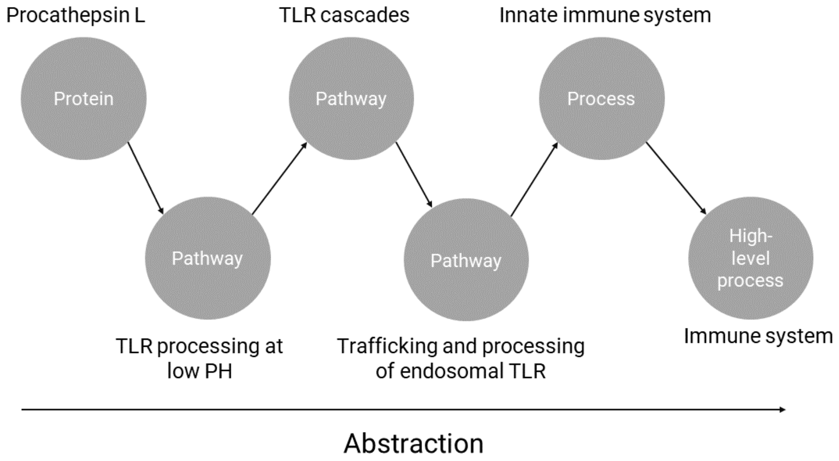

Figure 3.2: Example of a path in the Reactome pathway database. A protein (Procathepsin L) is mapped to biological pathways/processes with increasing level of abstraction. Each node can be seen as a sub-process of the following node.

for N layers If reaching a terminal node before N layers have been reached - add a copy of the previous node. This implies that the path depth $\leq N + 1$.

4. Remove nodes which have not been traversed.
5. Finally, connect proteins to the final corresponding terminal nodes.

The resulting architecture can be translated to a neural network by pruning the connections of each layer using a weight mask corresponding to the connectivity matrices of the respective layer. The output for a node is therefore:

$$y = f((MW)^T x + b)$$

where M is the masking matrix and W the weight matrix. The neural network was implemented in PyTorch - a common machine learning framework in Python. The network is sparse, containing trainable parameters in the thousands, as compared to 10^5 to 10^6 - which would be the case for densely connected networks with similar structure. A summation of the workflow can be seen in figure 3.3. Out of 563 proteins in the original dataset, after filtering and subsetting on the proteins present in the Reactome database, 446 proteins were left and used as input features. At the time of writing, the downloaded Reactome pathway database contained 2603 edges, of which 1856 (71.3%) were included in the network subsetted on the proteomic dataset.

Each hidden layer is followed by the hyperbolic tangent activation function:

$$tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$

The connections between hidden layers are intersected by a dropout layer, which randomly nullifies 20% of connections between layers.

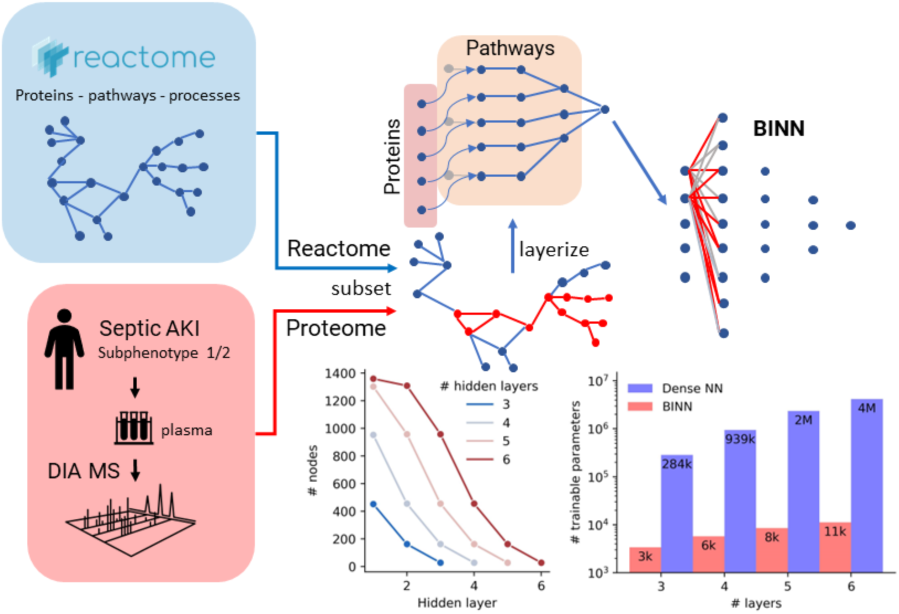

Figure 3.3: The BINN is generated by subsetting and linearizing the Reactome pathway database. The sparse graph is translated to a PyTorch framework. The sparsity of the connections reduces the number of trainable parameters in the network 100-fold to 1000-fold. Increasing the number of layers in the network naturally increases the number of trainable parameters, although the number of parameters is still very low compared to contemporary deep neural networks.

Using dropout is a common regularization technique implemented to reduce over-fitting [54]. Batch normalization was also applied after each hidden layer, to reduce the risk of vanishing or exploding gradients [55]. When training, the network seeks to reduce the cross-entropy loss function using an Adam optimizer. The cross entropy function is defined as:

$$Loss = -\sum_{i=1}^{n} t_i log(p_i)$$

where t_i is the truth label and p_i the probability for the *i:th* class. The loss function is weighted to account for the class imbalances in the dataset. Learning rate is initialized at 10^{-4}, and reduced tenfold when reaching a plateau in the loss landscape. Weight decay ($L2$-regularization) is applied as another means of reducing the risk of over-fitting. Parameters were initialized in accordance with Xavier uniform initialization for fast conversion [56]. A summation of the BINN architecture can be seen in figure 3.4.

It should be of note that the Reactome pathway graph has to be manipulated to generate the layered structure necessary for a sequential neural network. Therefore, as outlined in the algorithm above, nodes may have to be removed or inserted to fit the desired structure. The possible scenarios regarding path length and desired pathway length are outline in figure 3.5.

3.4.2 Evaluating the BINN

All performance evaluations were conducted with k-fold cross validation ($k = 5$). Models with different numbers of hidden layers ($3 \leq$ hidden layers ≤ 6) were generated using the aforementioned algorithm and evaluated. To investigate how dependent the model is on data, models were trained on data-subsets of varying sizes. All models were trained for 100 epochs. Where the number of layers is not explicitly stated, 4 layers were used to generate the model. This

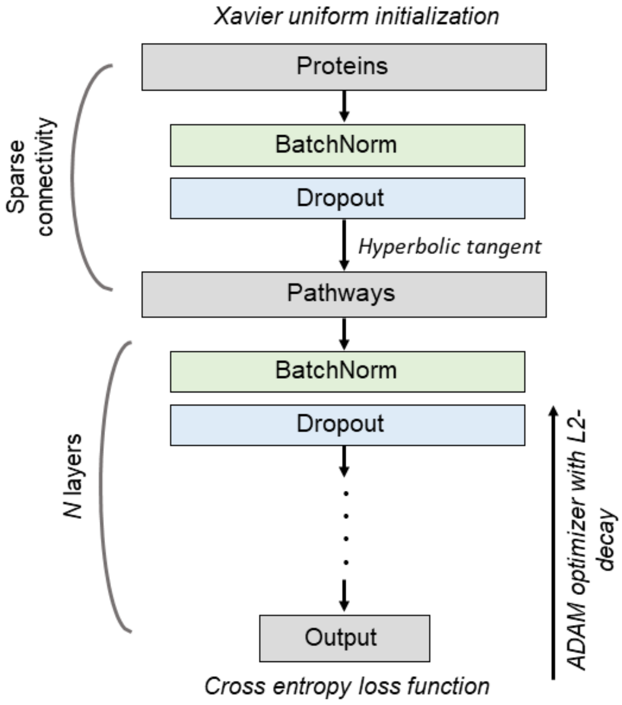

Figure 3.4: A visualization of the BINN architecture. Parameters are initialized with Xavier uniform initialization. Layers are connected sparsely as per defined by the Reactome database. Batch normalization and dropout layers follow each hidden layer. A hyperbolic tangent function is used as activation function. The number of hidden layers (N) is user defined. An ADAM optimizer with $L2$-regularization is used to minimize the cross entropy loss function in training.

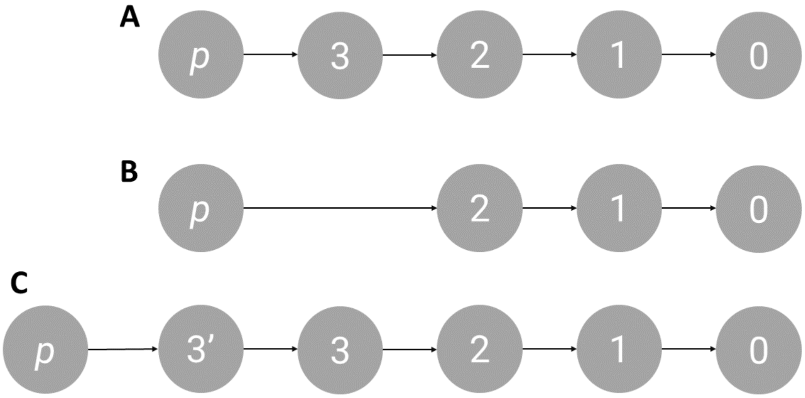

Figure 3.5: The possible scenarios of which path length versus desired network length, two of which require some manipulation of the database graph. Here, p denotes a protein, and the numbers in each node corresponds to the length from the terminal (most abstract) node. A) The path length is exactly the desired length. In such a scenario, no graph-manipulation is made. B) The desired length is shorter than the length in the database. In such a scenario, N nodes are kept and the final p-node is attached to *node(N)*. C) In the final scenario, the desired length is longer than the path length in the database. If so, a copy - in this case $3'$, is made of node 3, and the protein is attached to the copy.

was chosen to minimize the number of copies introduced during the generation of the BINN (72 copies for 4 hidden layers as compared to 367 copies for 5, see supplementary 6.1).

3.4.3 Interpreting the BINN

SHAP values were used to explain the contribution of each node in the network. The values were estimated using Deep SHAP - a combination of DeepLift and SHAP values as implemented in the SHAP python package. Deep SHAP utilizes a subset of the dataset to establish the expected outcome of the model, and then evaluates the outcome of an instance by comparing it to the expected outcome. Therefore, it is important that the class distribution of the background and evaluation data is held equal. 70% of the data was used as background and the remaining 30% to generate SHAP values. Since evaluation of the feature importance is done separately from evaluation of metrics related to accuracy, the interpreted network was trained on the entire dataset. The marginal contribution of a single node to each class was calculated by computing the mean of the absolute importance for each evaluation instance:

$$s = \frac{\sum_i^N |S_i|}{N}$$

where s - normalized SHAP value, S_i - SHAP value for instance i and N - number of evaluations.

Contribution was normalized between layers, to remove inter-layer discrepancies (assuming that each layer contributes equally to the prediction).

3.5 Clustering

To verify that the most important features identified by Deep SHAP are indeed significant features for classification, hierarchical clustering was employed on the ten proteins deemed most important. Agglomerative clustering was conducted using the euclidean distance and the Ward minimum variance method. The reduced feature vectors were also projected to two dimensions using a Uniform Manifold Projection and Approximation (UMAP) for visualization of erroneous classification.

3.6 Implementation

All code was implemented in Python 3.9.13. The MS-data was processed using an in-house package (DPKS)[2]. Sci-kit learn and PyTorch were used for used the implementation of classical machine learning methods and neural networks respectively.

[2] https://github.com/InfectionMedicineProteomics/DPKS

4 Results

4.1 Dataset

The general characteristics of the dataset were analyzed and differential expression performed between the two subphenotypes. This demonstrated that the dataset is homogeneous, showing little to no significant difference in precursor intensity distribution (figure 4.1A), protein abundance distribution (figure 4.1B), or peptides per protein distribution (figure 4.1C), indicating that no bias was introduced during sample preparation or was present in the sample prior to preparation.

However, several precursors and proteins are differentially expressed (figure 4.1D), inferring that the content of the samples differ. In total, there are 554 unique proteins in the dataset, out of which 77 proteins were considered differentially expressed (subphenotype 1: 32, subphenotype 2: 45). Differential expression was defined by:

$$-\log_{10}(p - value) \geq 1.3$$

and

$$max(P_2/P_1, P_1/P_2) \geq 2$$

where P_i is the protein or precursor abundance.

The number of precursors per sample (subphenotype 1: 3417 ± 243, subphenotype 2: 3509 ± 326, figure 4.2B) and proteins per sample (subphenotype 1: 330 ± 31, subphenotype 2: 360 ± 38, figure 4.2F) do not differ between subphenotypes.

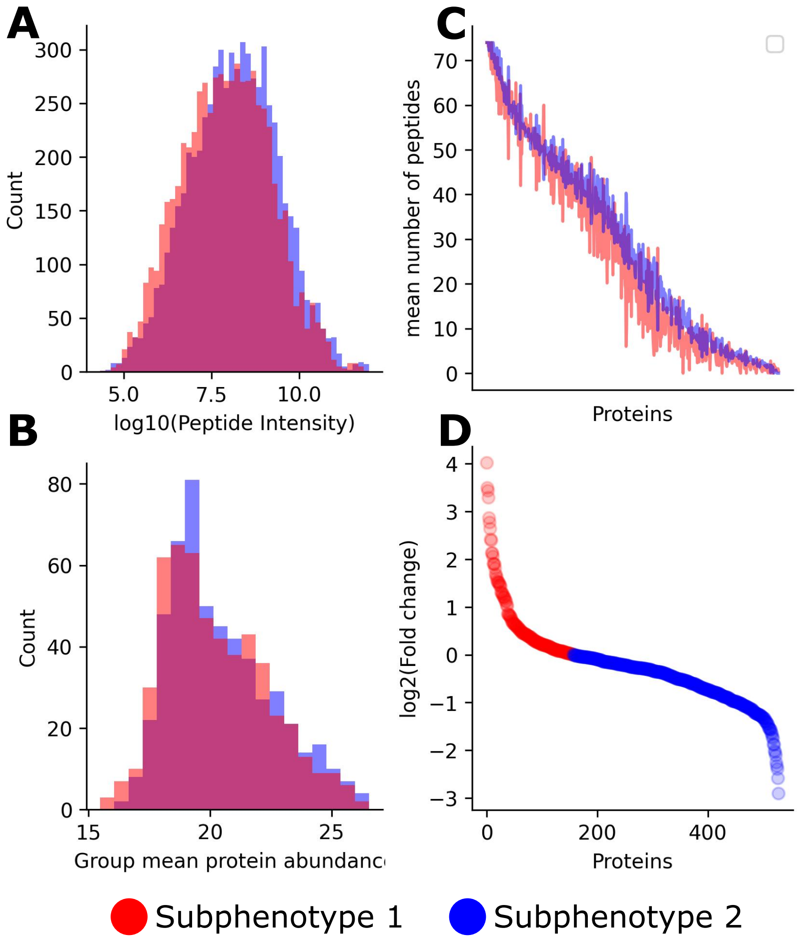

Figure 4.1: A) The precursor intensity follows similar distributions for both subphenotypes - although slightly shifted to more abundant precursors for subphenotype 2. B) After normalization and quantification, the protein abundances are similar for both groups. C) Proteins were ranked based on the mean number of precursors mapping to them. Individual proteins contain a similar number of precursors in both groups (corrected by the number of samples per group). D) Points were ranked according to their $log_2(Foldchange)$, and colored depending on their situation relative to 0. Subphenotype 2 contains more abundant proteins than subphenotype 1 (defined by the fold change of each protein).

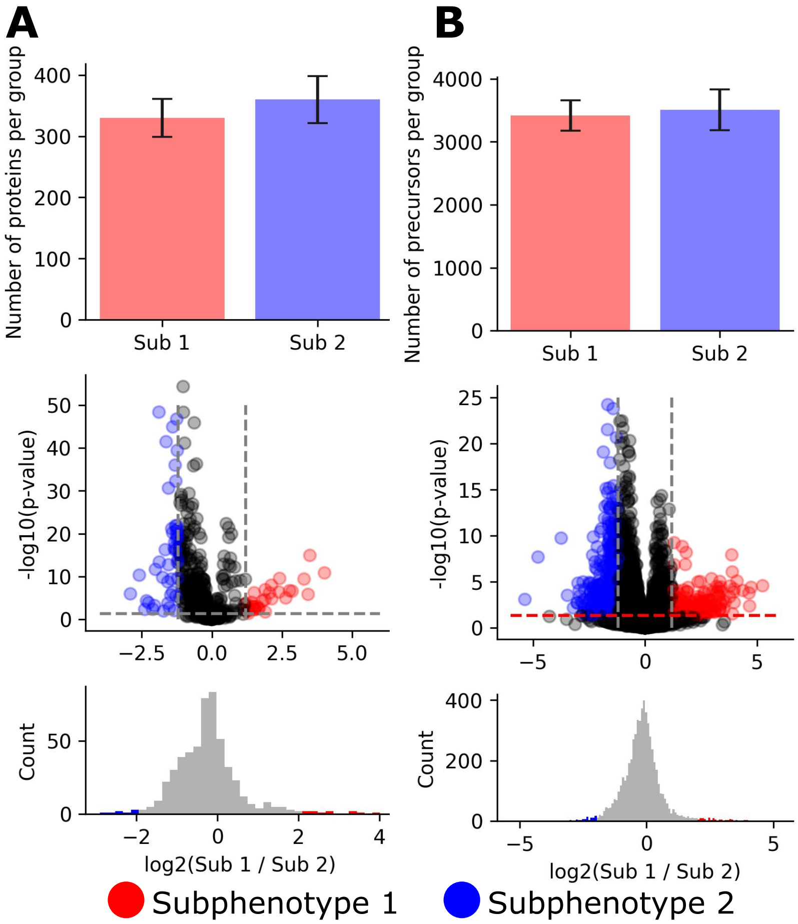

Figure 4.2: Number of precursors and peptides in the samples, and volcano plots where differentially expressed proteins are colored according to subphenotype. A) Subphenotype 1 and 2 contain a similar number of proteins per group (upper panel), although subphenotype 2 contains more differentially expressed proteins than subphenotype 1 (lower panels). B) Similarly, the two the samples of the two subphenotypes do not differ in number of precursors (upper panel), although, subphenotype 2 contains more differentially expressed precursors than subphenotype 1 (lower panels).

This indicate that although the nature of the samples from the subphenotypes (overarching structure) do not differ, the *content* of the samples do. The following machine learning methods are therefore not likely to train on *noise* which is induced by bias, but will instead discriminate between the subphenotypes based on the qualitative differences in proteomic content.

4.2 Classical machine learning

Four classifiers: SVM (RBF-kernel), k-NN, RF, XGBoost, LightGBM, were evaluated using k-fold cross validation ($k = 5$), using the scaled quantified protein abundance as features. The resulting confusion matrices and ROC-curves can be seen in figure 4.3. The SVM performed best (AUC: 0.96 ± 0.02), followed by XGBoost (AUC: 0.91 ± 0.04) (figure 4.3B). All models achieved a true positive and true negative rate of $\geq 70\%$ (figure 4.3A).

4.3 BINN

A BINN was constructed with 4 layers and evaluated using k-fold cross validation ($k = 5$). The confusion matrix, ROC-curve and validation loss/accuracy during training can be seen in 4.4. Training was conducted over 100 epochs. Generally, validation loss plateaued after 50 epochs, and $\geq 90\%$ accuracy was reached after 30 epochs. It is equally effective in predicting both classes, achieving a true positive and true negative rate of over 90%, and an AUC of 0.96 ± 0.03 (figure 4.4A) - thereby outperforming the classical ML algorithms. The number of hidden layers in the BINN had little to no effect on model accuracy (figure 4.4B). The BINN also proved efficient when trained on a low number of datapoints - achieving $\geq 80\%$ accuracy when trained on 20% of the data (figure 9B).

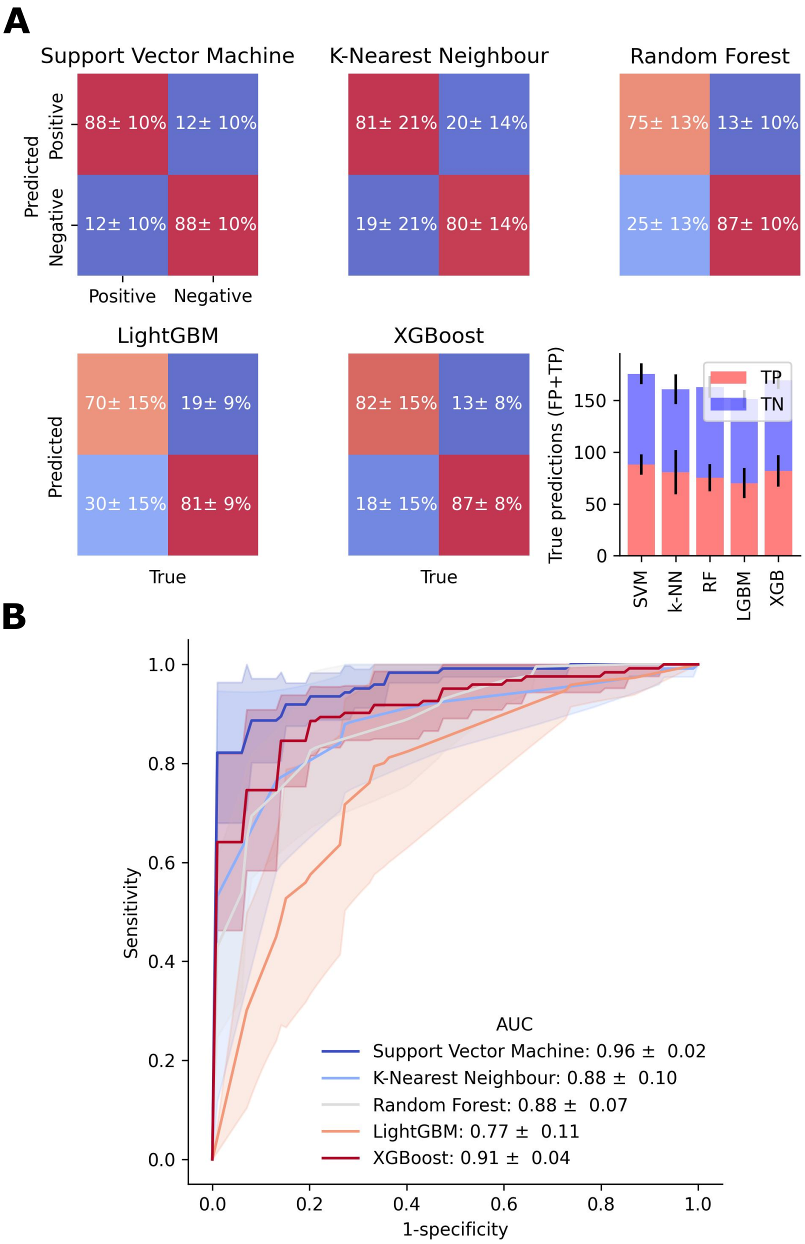

Figure 4.3: A) The confusion matrices show the rates as percentages of true and false classifications. Rates in the descending diagonal are the rates of predictions which correspond to the true class. The SVM achieved the highest true positive and true negative rate of all the models. B) An ROC-curve for all models. Approaching the upper left corner implies a perfect model. The SVM performed the best, with an AUC-score of 0.96 ± 0.02. All models except LightGBM received a mean AUC-score of 0.88 and above.

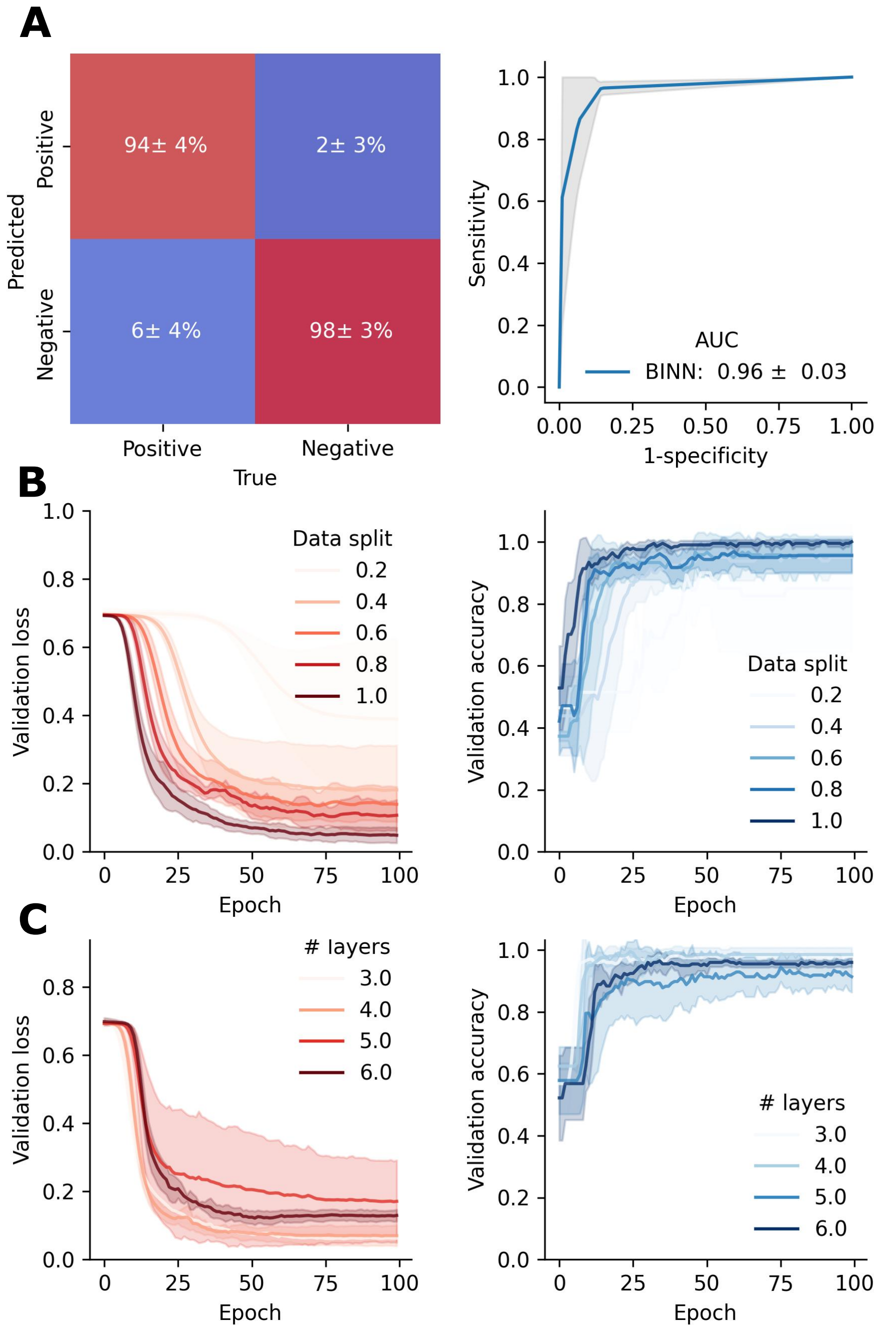

Figure 4.4: A) The BINN receives an AUC of 0.96 ± 0.03, a true positive rate of 94 ± 4% and a true negative rate of 98 ± 3%. B) Validation loss was decreased, and accuracy increased by incorporating more data, although the model achieved high accuracy ($\sim 90\%$) even when excluding 80% of the data. C) The number of layers included in the model had little effect on loss and accuracy.

4.4 Interpreting the network

The BINN achieved a high accuracy, inferring that some aspects of the proteomic content of the samples which are reflected in the trained network are telling of the discrepancies between the subphenotypes. SHAP values were therefore used to interpret the network to unveil the biological entities important for classification. The absolute feature importance for each node and each class is calculated and the total importance normalized across layers. The connectivity of the network is known (as it is per our design), and it is therefore possible to visualize how feature importance propagates through the network using a Sankey diagram (figure 4.5). The *flow* in the Sankey diagram is defined by the SHAP values and the network connectivity. The most important nodes in each layers are shown.

The ten proteins deemed most important were: apolipoprotein A1 (APOA1), apolipoprotein B (APOB), apolipoprotein A4 (APOA4), cluster of differentiation 14 (CD14), cathepsin Z (CATZ), actin beta (ACTB), histone H2A type 1-b (H2A1B), osteopontin (OSTP), collagen type 1 alpha 1 (C1A1) and histone H2 type 1–a (H2B1A). In the final layer (highest level), the top-ranking processes were: metabolism of proteins, immune system, disease, and metabolism. Amongst the intermediary layers, the innate immune system, neutrophil degranulation, and retinoid transport and transport stand out as important pathways.

4.5 Clustering

The top ten proteins defined by SHAP are visualized in a correlation plot in figure 4.6A. To validate the findings by SHAP, reduced feature vectors consisting only of the ten proteins deemed most important by SHAP value were used to cluster the dataset. Hierarchical clustering resulted in two distinct groups. Group member-

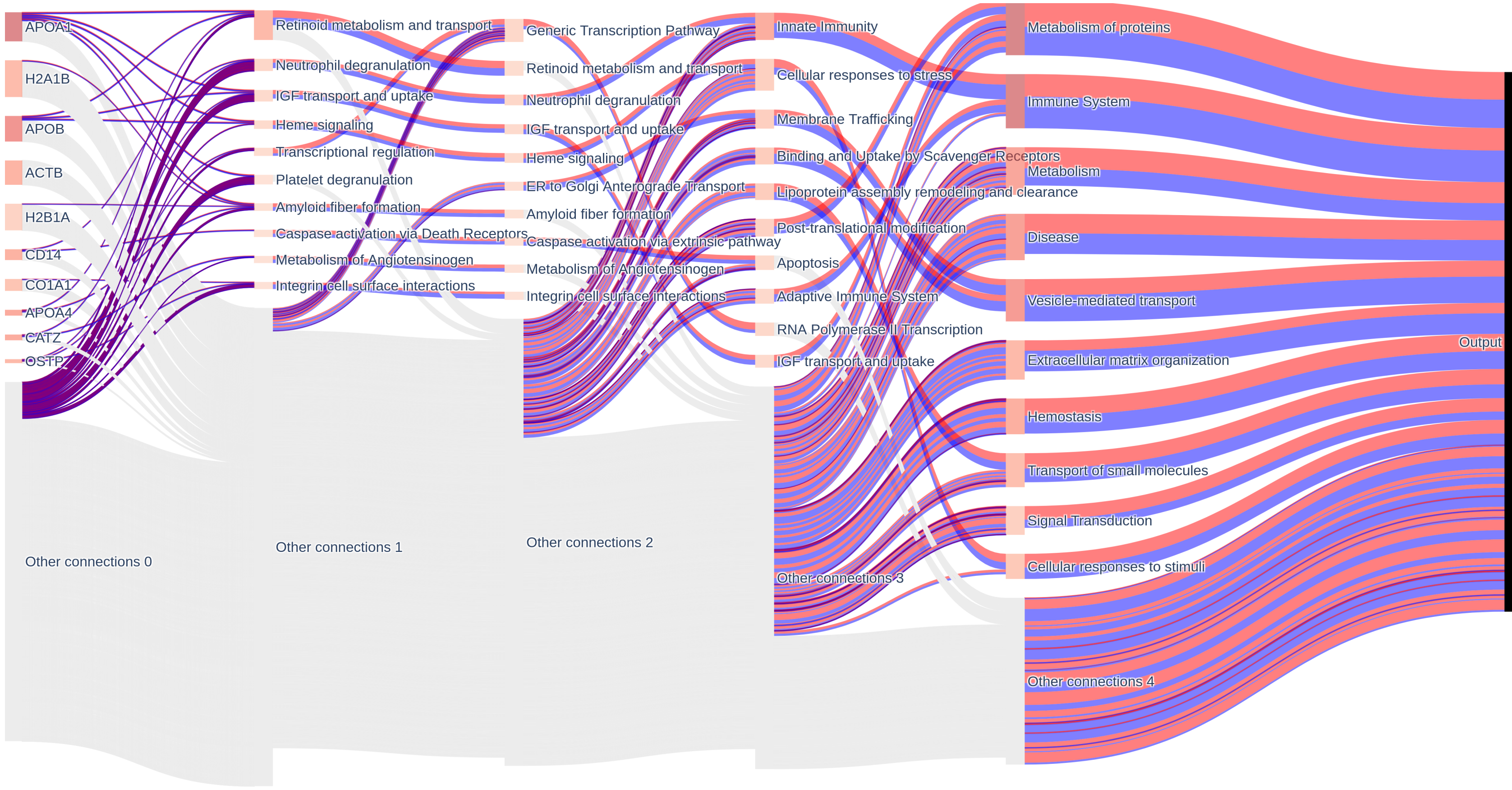
APOA1
H2A1B
APOB
ACTB
H2B1A
CD14
CO1A1
APOA4
CATZ
OSTP
Other connections 0
Retinoid metabolism and transport
Neutrophil degranulation
IGF transport and uptake
Heme signaling
Transcriptional regulation
Platelet degranulation
Amyloid fiber formation
Caspase activation via Death Receptors
Metabolism of Angiotensinogen
Integrin cell surface interactions
Other connections 1
Generic Transcription Pathway
Retinoid metabolism and transport
Neutrophil degranulation
IGF transport and uptake
Heme signaling
ER to Golgi Anterograde Transport
Amyloid fiber formation
Caspase activation via extrinsic pathway
Metabolism of Angiotensinogen
Integrin cell surface interactions
Other connections 2
Innate Immunity
Cellular responses to stress
Membrane Trafficking
Binding and Uptake by Scavenger Receptors
Lipoprotein assembly remodeling and clearance
Post-translational modification
Apoptosis
Adaptive Immune System
RNA Polymerase II Transcription
IGF transport and uptake
Other connections 3
Metabolism of proteins
Immune System
Metabolism
Disease
Vesicle-mediated transport
Extracellular matrix organization
Hemostasis
Transport of small molecules
Signal Transduction
Cellular responses to stimuli
Other connections 4
Output

Figure 4.5: A Sankey diagram visualizing the SHAP values for the nodes in each layer (lower-level layers to the left and higher level layers to the right). The output of each node is the SHAP values for the two subphenotypes (subphenotype 1 in red and subphenotype 2 in blue). Connections in the diagram are the connections present in the BINN.

ship corresponded to subphenotypes in 191/197 (97%) cases (figure 4.6B). Clustering using the top 100 most important proteins resulted in 134/197 (68%) correct classifications. Uniform Manifold Projection and Approximation (UMAP) is a dimensionality reduction technique and was used to project the feature vectors to two dimensions for visualization [56] (figure 4.6C).

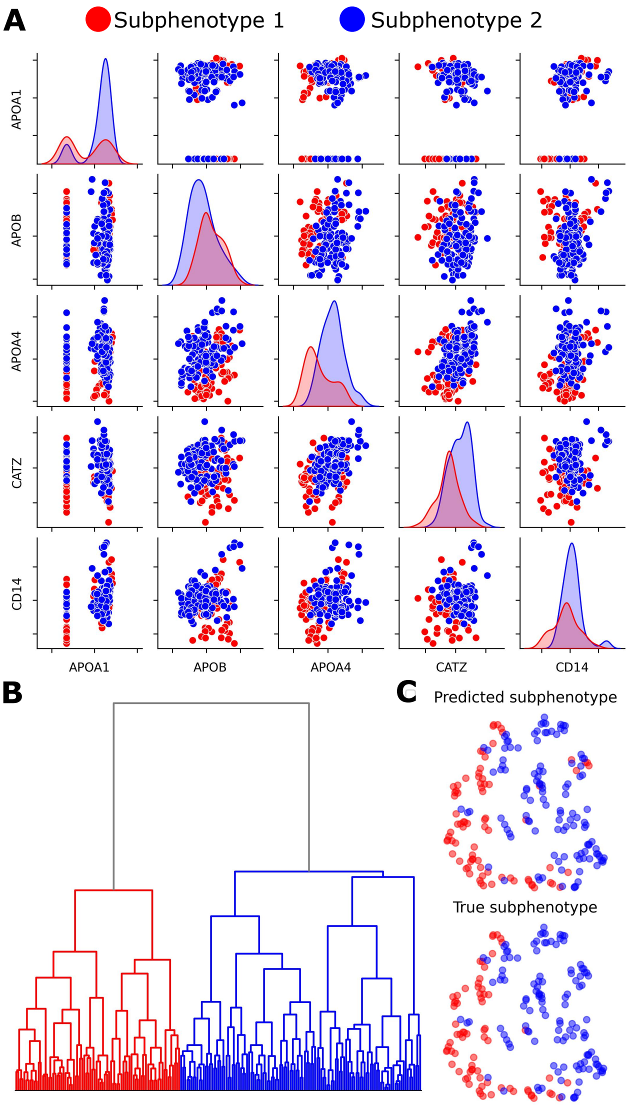

Figure 4.6: A) Pairwise correlation plots of the abundance of the 5 most important proteins. B) Resulting dendrogram after hierarchical clustering was performed using the Ward minimum variance method of the top 10 most important proteins. The data is clearly divided into two major groups, where 191/197 samples were correctly divided by subphenotype. C) UMAP projection of the reduced dataset using the top 10 most important proteins. The data points are colored based on predicted subphenotype (upper) and true subphenotype from the hierarchical clustering (lower).

5 Discussion

Although the proteomic profile of sepsis has been investigated, this is - to our knowledge - the first time proteomic data has been used to predict sepsis subphenotypes. In this study, plasma samples from patients suffering from septic AKI of two different subphenotypes were analyzed with DIA-MS to generate a proteomic dataset. Classical machine learning methods were able to classify the samples well, as the SVM received an AUC score of ≤ 0.9 and other models a score of ≤ 0.8. There are several techniques that one could leverage to increase prediction accuracy of classical ML models, such as implementing ensemble voting [57]. However, the ability to accurately predict sepsis from proteomic data has limited real-world applications, as currently large-scale proteomic data rarely is available in clinical settings. Furthermore, methods utilizing clinical data alongside readily available biomarkers have yielded high accuracy and are more realizable than those utilizing proteomic data. Machine learning methods trained on proteomic data may, however, help us understand the important features of the proteomic data and thereby the nature of the disease itself, which is highly desirable. Additionally, they may contribute to the discovery of novel biomarkers and hypotheses which are of clinical relevance, as exemplified by Van Allen et al. [52]. Therefore, we constructed a BINN - a sparse, biologically informed neural network and used it to elucidate biologically important pathways when classifying the two subphenotypes of AKI.

The informed nature of the BINN has several advantages over densely connected neural networks. Firstly, the design can be seen as a means of intelligently pruning, resulting in a stark reduction in trainable parameters, allowing for a reduction in training time, datapoints, and demands on computing power, whilst maintaining a high prediction accuracy [58]. Secondly, since nodes and connections are annotated, it allows us to interpret the network to gain insight into its reasoning and thereby understand which biological pathways the network deemed important for classification. The BINN performed better than classical machine learning methods in stratifying the two subphenotypes, implying that some aspects of the true underlying biology is reflected in the network, the nature of which can be unveiled by interpreting it. This stays true for varying number of layers and when excluding a large portion of the dataset in training.

Shapley values allow us to utilize the informed architecture of the network to interpret it. On the protein level, apolipoproteins, histones and other inflammatory markers with known implications in sepsis were deemed important. Apolipoproteins play a role in lipid metabolism and have been connected to several diseases including sepsis [59]. Circulating histones are important during the progression of sepsis and have been identified both as possible biomarkers and therapeutic targets as they amplify the dysregulated immune response [60]. Other inflammatory markers with known relation to sepsis include CD14, cathepsin Z (CATZ) [61], osteopontin (OSTP) [62] and calreticulin (CALR) [63]. However, proteins with undocumented relations to sepsis were also highlighted in the BINN, such as actin beta (ACTB) and collagen type 1 alpha 1 (CO1A1), suggesting that these should be subject to further studies in connection with septic AKI. Reassuringly, the most important higher level biological pathways and processes include pathways linked to immunity, metabolism and disease, which is to be expected considering the nature of the dataset.

The quantities of the proteins identified by SHAP did differ slightly between samples, but interestingly, not to to the extent one might suspect. This demonstrates that the features most important for classification are *not the most differentially expressed proteins.* Therefore, the method complements many contemporary methods based on differential expression.

The BINN is completely dependent on the underlying Reactome pathway database, the proteomic dataset and the overlap between the two. Incompleteness in either part will affect the validity of conclusions drawn from the network. Fortunately, the proteomic dataset was largely covered by the pathway database, and the resulting network covered a large portion of the complete Reactome database. The reactome database sets an upper limit to the size of the network which I estimate is $\sim$ 8500 trainable parameters (without inducing too many copies). The complete BINN-algorithm isn't database specific, but can be generalized to any pathway database, such as Metascape [64].

The few data-points alongside the comparatively large number of features typify the curse of dimensionality described in section 1.2. To maximally utilize the available data-points, k-fold cross validation was implemented to evaluate all models, and no test-set was used for evaluation. This does not lessen the validity of any conclusions drawn in this project, although a final validation on a test set would be preferable given a large enough dataset. To reduce the risk of over-fitting, measures such as dropout, batch normalization and $L2$-regularization were implemented. None of the typical sign of a failure to generalize was exhibited in the BINNs, suggesting that the implemented measures were effective in averting over-fitting.

This work was an extension of the work by Van Allen et al. where the idea of a BINN consisting of biological pathways was first introduced [52]. Further development of the BINN is certainly possible. In this thesis, protein abundances were the sole input feature, however, one could imagine the incorporation of features such as protein modifications as well. Although this would increase the demands on the data-generation and pre-processing, such features could add to the information gained from interpreting the network. So far, the BINN has only been used for binary classification, however, the algorithm is compatible with multinomial classification tasks. Further studies including various sepsis endotypes and subphenotypes to train the same network would be of great interest. Naturally, the algorithm is not bound to any type of proteomic dataset, but can be applied to other conditions, diseases or syndromes, allowing for the investigation of a myriad of new research questions.

5.1 Implementation

The algorithm generating a sparse neural network from an edge file and an input is available as a package at:
https://github.com/InfectionMedicineProteomics/BINN

5.2 Sustainable development

Disease and ill-health is not sustainable - nor are toxic and ineffective treatments, as is emphasized in the third Sustainable Development Goal (SGD). An older, more fragile and larger population increases the global demand for sustainable and effective diagnostic strategies and treatments, which don't cause harm to the environment or the population. Although great strides have been taken in medical research and it's implementation in clinics, a growing resistance towards treatments such as antibiotics and our apparent ineptness in averting a viral pandemic highlights that much is yet to be accomplished before we've reached a satisfactory state of our health-care. Furthermore, the benefits of modern health-care are often only available to a minority of the global population, giving rise to a health-inequality which has to be combated before such a claim of success can be made.

The foundation of good health-care ultimately lies in the *understanding* of diseases and conditions - both on a molecular and on a grander scale. The goal of this project was to contribute to the contemporary understanding of one of the deadliest diseases in the world - sepsis, and consequentially, to contribute to the development of novel remedies and diagnostic tools. Although much is yet to be sought in regards to the understanding of the pathogenesis and complex dynamic which is sepsis, I believe methods such as this - which incorporates vast amounts of data to unravel the underlying biological processes involved, are key to furthering development.

It has become necessary to consider the environmental impact when utilizing deep learning approaches, as models closing in on trillions of parameters are energy-intensive to train, which has resulted in a questioning their sustainability. Smaller models can therefore be seen as more sustainable, and it is uplifting to see that a model with a few thousand parameters as our BINN can be intelligently designed to provide insight into complex syndromes.

Lastly, openness in the scientific community leads to a sustainable research environment. The act of sharing methods, code and datasets facilitates further studies, which is effort was made to make the methods presented in this project publicly available through a Python package.

6 Supplementary

6.1 Number of induced copies

As described above, specifying a desired network depth greater than the path-length in the given database will result in generating copies of nodes. The number of copies induced over number of desired layers can be seen in figure 6.1.

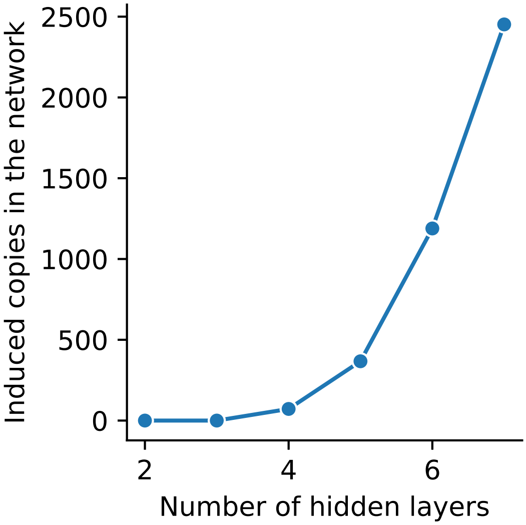

Figure 6.1: Number of induced copies grows exponentially over the number of hidden layers in the network.

Bibliography

[1] Singer M, Deutschman CS, Seymour CW, Shankar-Hari M, Annane D, Bauer M, Bellomo R, Bernard GR, Chiche JD, Coopersmith CM, Hotchkiss RS, Levy MM, Marshall JC, Martin GS, Opal SM, Rubenfeld GD, van der Poll T, Vincent JL and Angus DC. "The Third International Consensus Definitions for Sepsis and Septic Shock (Sepsis-3)". In: *JAMA* (2016). DOI: 10.1001/jama.2016.0287.

[2] Rudd KE, Johnson SC, Agesa KM, Shackelford KA, Tsoi D, Kievlan DR, Colombara DV, Ikuta KS, Kissoon N, Finfer S, Fleischmann-Struzek C, Machado FR, Reinhart KK, Rowan K, Seymour CW, Watson RS, West TE, Marinho F, Hay SI, Lozano R, Lopez AD, Angus DC, Murray CJL and Naghavi M. "Global, regional, and national sepsis incidence and mortality, 1990-2017: analysis for the Global Burden of Disease Study". In: *Lancet* (2020). DOI: 10.1016/S0140-6736(19)32989-7.

[3] Konrad Reinhart, M.D., Ron Daniels, M.D., Niranjan Kissoon, M.D., Flavia R. Machado, M.D., Ph.D., Raymond D. Schachter, L.L.B., Simon Finfer and M.D. "Recognizing Sepsis as a Global Health Priority". In: *N Engl J Med.* (2017). DOI: 10.1056/NEJMp1707170.

[4] *Global sepsis alliance.* URL: https://www.global-sepsis-alliance.org/.

[5] *European sepsis alliance.* URL: https://www.europeansepsisalliance.org/.

[6] Leligdowicz A and Matthay MA. “Heterogeneity in sepsis: new biological evidence with clinical applications”. In: *Crit Care.* (2019). DOI: 10.1186/s13054-019-2372-2.

[7] Jarczak D., Kluge S and Nierhaus A. “Sepsis—Pathophysiology and Therapeutic Concepts”. In: *Front. Med.* (2021). DOI: 10.3389/fmed.2021.628302.

[8] Saito S, Uchino S, Hayakawa M, Yamakawa K, Kudo D, Iizuka Y, Sanui M, Takimoto K, Mayumi T and Sasabuchi Y. “Epidemiology of disseminated intravascular coagulation in sepsis and validation of scoring systems”. In: *J Crit Care* (2019). DOI: 10.1016/j.jcrc.2018.11.009.

[9] Iscimen, Remzi MD; Cartin-Ceba, Rodrigo MD; Yilmaz, Murat MD; Khan, Hasrat MD; Hubmayr, Rolf D. MD; Afessa, Bekele MD; Gajic and Ognjen MD MSc. “Risk factors for the development of acute lung injury in patients with septic shock: An observational cohort study”. In: *Critical Care Medicine* (2008). DOI: 10.1097/CCM.0b013e31816fc2c0.

[10] Robert W. Schrier M.D. and Wei Wang M.D. “Risk factors for the development of acute lung injury in patients with septic shock: An observational cohort study”. In: *N Engl J Med* (2004). DOI: 10.1056/NEJMra032401.

[11] Arjun Baghela, Olga M. Pena, Amy H. Lee, Beverlie Baquir, Reza Falsafi, Andy An, Susan W. Farmer, Andrew Hurlburt, Alvaro Mondragon-Cardona, Juan Diego Rivera, Andrew Baker, Uriel Trahtemberg, Maryam Shojaei, Carlos Eduardo Jimenez-Canizales, Claudia C. dos Santos, Benjamin Tang, Hjalmar R. Bouma, Gabriela V. Cohen Freue and Robert E.W. Hancock. “Predicting sepsis severity at first clinical presentation: The role of endotypes and mechanistic signatures”. In: *eBioMedicine* (2021). DOI: https://doi.org/10.1016/j.ebiom.2021.103776.

[12] Gårdlund B, Dmitrieva NO, Pieper CF, Finfer S, Marshall JC and Taylor Thompson B. "Six subphenotypes in septic shock: Latent class analysis of the PROWESS Shock study." In: *J Crit Care.* (2018). DOI: 10.1016/j.jcrc.2018.06.012.

[13] Xu Z, Mao C, Su C, Zhang H, Siempos I, Torres LK, Pan D, Luo Y, Schenck EJ and Wang F. "Sepsis subphenotyping based on organ dysfunction trajectory". In: *J Crit Care.* (2022). DOI: 10.1186/s13054-022-04071-4.

[14] Nisula S., Kaukonen KM. and Vaara S.T. et al. "Incidence, risk factors and 90-day mortality of patients with acute kidney injury in Finnish intensive care units: the FINNAKI study". In: *Intensive Care Med* (2013). DOI: https://doi.org/10.1007/s00134-012-2796-5.

[15] Wiersema R., Jukarainen S. and Vaara S.T. et al. "Two subphenotypes of septic acute kidney injury are associated with different 90-day mortality and renal recovery". In: *Intensive Care Med* (2020). DOI: https://doi.org/10.1186/s13054-020-02866-x.

[16] Jack Varon and Rebecca M. Baron. "Sepsis endotypes: The early bird still gets the worm". In: *eBioMedicine* (2022). DOI: https://doi.org/10.1016/j.ebiom.2022.103832.

[17] Christopher M . Bishop. *Patter Recognition and Machine Learning*. 2006.

[18] Jeff Heaton. "An Empirical Analysis of Feature Engineering for Predictive Modeling". In: *arXiv* (2017). DOI: https://doi.org/10.48550/arXiv.1701.0785.

[19] Joe H. Ward Jr. "Hierarchical Grouping to Optimize an Objective Function". In: *Journal of the American Statistical Association* (1963). DOI: 10.1080/01621459.1963.10500845.

[20] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar and Nando de Freitas. "A Generalist Agent". In: *arXiv* (2022). DOI: arXiv:2205.06175.

[21] A. L. Samuel. "Some Studies in Machine Learning Using the Game of Checkers". In: *BM Journal of Research and Development* (1959). DOI: 10.1147/rd.33.0210.

[22] Theodoros Evgeniou and Massimiliano Pontil. *Support Vector Machines: Theory and Applications*. 2005.

[23] Karl Thurnhofer-Hemsi, Ezequiel López-Rubio, Miguel A. Molina-Cabello and Kayvan Najarian. "Radial basis function kernel optimization for Support Vector Machine classifiers". In: *arXiv* (2007). DOI: arXiv:2007.08233.

[24] Breiman L. "Random Forests". In: *Machine Learning* (2001). DOI: https://doi.org/10.1023/A:1010933404324.

[25] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016). DOI: https://doi.org/10.1145/2939672.2939785.

[26] Guolin Kem, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye and Tie-Yan Liu. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". In: *31st Conference on Neural Information Processing Systems (NIPS 2017)* (2017).

[27] K. Taunk, S. De, S. Verma and A. Swetapadma. "A Brief Review of Nearest Neighbor Algorithm for Learning and Classification". In: *2019 International Conference on Intelligent Computing and Control Systems (ICCS)* (2019). DOI: 10.1109/ICCS45141.2019.9065747.

[28] D. O. Hebb. *The organization of behavior; a neuropsychological theory*. Wiley, 1949.

[29] B. Farley and W. Clark. “Simulation of self-organizing systems by digital computer”. In: *Transactions of the IRE Professional Group on Information Theory* (1954). DOI: 10.1109/TIT.1954.1057468.

[30] Rumelhart D., Hinton G. and Williams R. “Learning representations by back-propagating errors”. In: *Nature* (1986). DOI: https://doi.org/10.1038/323533a0.

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *arXiv* (2015). DOI: arXiv:1512.03385.

[32] Yamashita R.and Nishio M. and Do R.K.G. et al. “Convolutional neural networks: an overview and application in radiology”. In: *Insights Imaging* (2018). DOI: https://doi.org/10.1007/s13244-018-0639-9.

[33] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever and Dario Amodei. “Language Models are Few-Shot Learners”. In: *34th Conference on Neural Information Processing Systems (NeurIPS 2020)* (2020).

[34] Ian J. Goodfellow, Jonathon Shlens and Christian Szegedy. “Explaining and Harnessing Adversarial Examples”. In: *arXiv* (2014). DOI: https://doi.org/10.48550/arXiv.1412.6572.

[35] P. Jonathon Phillips Carina A. Hahn Peter C. Fontana Amy N. Yates Kristen Greene David A. Broniatowski Mark A. Przybocki. “Four Principles of Explainable Artificial Intelligence”. In: *NISTIR* (2020). DOI: https://doi.org/10.6028/NIST.IR.8312-draft.

[36] Scott M. Lundberg and Su-In Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems 30 (NIPS 2017)* (2017). DOI: https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html.

[37] Marco Tulio Ribeiro, Sameer Singh and Carlos Guestrin. “”Why Should I Trust You?”: Explaining the Predictions of Any Classifier”. In: *arXiv)* (2016). DOI: https://doi.org/10.48550/arXiv.1602.04938.

[38] Avanti Shrikumar, Peyton Greenside and Anshul Kundaje. “Learning Important Features Through Propagating Activation Differences”. In: *arXiv)* (2017). DOI: https://doi.org/10.48550/arXiv.1704.02685.

[39] Aebersold R. and Mann M. “Mass-spectrometric exploration of proteome structure and function”. In: *Nature* (2016). DOI: https://doi.org/10.1038/nature19949.

[40] Picotti P. and Aebersold R. “Selected reaction monitoring–based proteomics: workflows, potential, pitfalls and future directions.” In: *Nat Methods* (2012). DOI: https://doi.org/10.1038/nmeth.2015.

[41] Doerr A. “DIA mass spectrometry”. In: *Nat Methods* (2015). DOI: https://doi.org/10.1038/nmeth.3234.

[42] Röst H., Rosenberger G. and Navarro P. et al. “OpenSWATH enables automated, targeted analysis of data-independent acquisition MS data.” In: *Nat Biotechnol* (2014). DOI: https://doi.org/10.1038/nbt.2841.

[43] Giacobbe DR, Signori A, Del Puente F, Mora S, Carmisciano L, Briano F, Vena A, Ball L, Robba C, Pelosi P, Giacomini M and Bassetti M. "Early Detection of Sepsis With Machine Learning Techniques: A Brief Clinical Perspective". In: *Front Med* (2021). DOI: 10.3389/fmed.2021.617486.

[44] Goh K.H., Wang L. and Yeow A.Y.K. et al. "Artificial intelligence in sepsis early prediction and diagnosis using unstructured data in healthcare". In: *Nat Commun* (2021). DOI: https://doi.org/10.1038/s41467-021-20910-4.

[45] Fleuren L.M., Klausch T.L.T. and Zwager C.L. et al. "Machine learning for the prediction of sepsis: a systematic review and meta-analysis of diagnostic test accuracy". In: *Intensive Care Med* (2020). DOI: https://doi.org/10.1007/s00134-019-05872-y.

[46] Malmström E., Kilsgård O. and Hauri S. et al. "Large-scale inference of protein tissue origin in gram-positive sepsis plasma using quantitative targeted proteomics". In: *Nat Commun* (2016). DOI: https://doi.org/10.1038/ncomms10261.

[47] Pimienta G, Heithoff DM, Rosa-Campos A, Tran M, Esko JD, Mahan MJ, Marth JD and Smith JW. "Plasma Proteome Signature of Sepsis: a Functionally Connected Protein Network". In: *Epub* (2019). DOI: 10.1002/pmic.201800389.

[48] Hayashi N, Yamaguchi S, Rodenburg F, Ying Wong S, Ujimoto K, Miki T and et al. "Multiple biomarkers of sepsis identified by novel time-lapse proteomics of patient serum". In: *PLoS ONE* (2019). DOI: https://doi.org/10.1371/journal.pone.0222403.

[49] The UniProt Consortium. "UniProt: the universal protein knowledgebase in 2021". In: *Nucleic Acids Research* (2021). DOI: https://doi.org/10.1093/nar/gkaa1100.

[50] Fiona Cunningham, James E Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Olanrewaju Austine-Orimoloye, Andrey G Azov, If Barnes, Ruth Bennett, Andrew Berry, Jyothish Bhai, Alexandra Bignell, Konstantinos Billis, Sanjay Boddu, Lucy Brooks, Mehrnaz Charkhchi, Carla Cummins, Luca Da Rin Fioretto, Claire Davidson, Kamalkumar Dodiya, Sarah Donaldson, Bilal El Houdaigui, Tamara El Naboulsi, Reham Fatima, Carlos Garcia Giron, Thiago Genez, Jose Gonzalez Martinez, Cristina Guijarro-Clarke, Arthur Gymer, Matthew Hardy, Zoe Hollis, Thibaut Hourlier, Toby Hunt, Thomas Juettemann, Vinay Kaikala, Mike Kay, Ilias Lavidas, Tuan Le, Diana Lemos, José Carlos Marugán, Shamika Mohanan, Aleena Mushtaq, Marc Naven, Denye N Ogeh, Anne Parker, Andrew Parton, Malcolm Perry, Ivana Piližota, Irina Prosovetskaia, Manoj Pandian Sakthivel, Ahamed Imran Abdul Salam, Bianca M Schmitt, Helen Schuilenburg, Dan Sheppard, José G Pérez-Silva, William Stark, Emily Steed, Kyösti Sutinen, Ranjit Sukumaran, Dulika Sumathipala, Marie-Marthe Suner, Michal Szpak, Anja Thormann, Francesca Floriana Tricomi, David Urbina-Gómez, Andres Veidenberg, Thomas A Walsh, Brandon Walts, Natalie Willhoft, Andrea Winterbottom, Elizabeth Wass, Marc Chakiachvili, Bethany Flint, Adam Frankish, Stefano Giorgetti, Leanne Haggerty, Sarah E Hunt, Garth R IIsley, Jane E Loveland, Fergal J Martin, Benjamin Moore, Jonathan M Mudge, Matthieu Muffato, Emily Perry, Magali Ruffier, John Tate, David Thybert, Stephen J Trevanion, Sarah Dyer, Peter W Harrison, Kevin L Howe, Andrew D Yates, Daniel R Zerbino and Paul Flicek. "Ensembl". In: *Nucleic Acids Research* (2022). DOI: https://doi.org/10.1093/nar/gkab1049.

[51] Marc Gillespie, Bijay Jassal, Ralf Stephan, Marija Milacic, Karen Rothfels, Andrea Senff-Ribeiro, Johannes Griss, Cristoffer Sevilla, Lisa Matthews, Chuqiao Gong, Chuan Deng, Thawfeek Varusai, Eliot Ragueneau, Yusra Haider, Bruce May,

Veronica Shamovsky, Joel Weiser, Timothy Brunson, Nasim Sanati, Liam Beckman, Xiang Shao, Antonio Fabregat, Konstantinos Sidiropoulos, Julieth Murillo, Guilherme Viteri, Justin Cook, Solomon Shorser, Gary Bader, Emek Demir, Chris Sander, Robin Haw, Guanming Wu, Lincoln Stein, Henning Hermjakob and Peter D'Eustachio. "The reactome pathway knowledgebase 2022". In: *Nucleic Acids Research* (2022). DOI: https://doi.org/10.1093/nar/gkab1028.

[52] Elmarakeby H.A., Hwang J. and Arafeh R. et al. "Biologically informed deep neural network for prostate cancer discovery". In: *Nature* (2021). DOI: https://doi.org/10.1038/s41586-021-03922-4.

[53] Cox J, Hein MY, Luber CA, Paron I, Nagaraj N and Mann M. "Accurate proteome-wide label-free quantification by delayed normalization and maximal peptide ratio extraction, termed MaxLFQ." In: *Mol Cell Proteomics* (2014). DOI: 10.1074/mcp.M113.031591.

[54] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *arXiv* (2015). DOI: https://doi.org/10.48550/arXiv.1502.03167.

[55] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *AISTATS* (2010). DOI: https://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf.

[56] Leland McInnes, John Healy and James Melville. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction". In: *arXiv* (2018). DOI: https://doi.org/10.48550/arXiv.1802.03426.

[57] Eric Bax. "Selecting a number of voters for a voting ensemble". In: *arXiv* (2021). DOI: https://arxiv.org/pdf/2104.11833.pdf.

[58] Jonathan Frankle and Michael Carbin. “The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks”. In: *arXiv* (2018). DOI: https://arxiv.org/abs/1803.03635.

[59] Barlage S., Gnewuch C. and Liebisch G. et al. “Changes in HDL-associated apolipoproteins relate to mortality in human sepsis and correlate to monocyte and platelet activation”. In: *Intensive Care Med* (2009). DOI: https://doi.org/10.1007/s00134-009-1609-y.

[60] Li Y, Wan D, Luo X, Song T, Wang Y, Yu Q, Jiang L, Liao R, Zhao W and Su B. “Circulating Histones in Sepsis: Potential Outcome Predictors and Therapeutic Targets”. In: *Front. Immunol.* (2021). DOI: 10.3389/fimmu.2021.650184.

[61] Campden RI, Warren AL, Greene CJ, Chiriboga JA, Arnold CR, Aggarwal D, McKenna N, Sandall CF, MacDonald JA and Yates RM. “Extracellular cathepsin Z signals through the 5 integrin and augments NLRP3 inflammasome activation”. In: *J Biol Chem* (2021). DOI: 10.1016/j.jbc.2021.101459.

[62] Castello LM, Baldrighi M, Molinari L, Salmi L, Cantaluppi V, Vaschetto R, Zunino G, Quaglia M, Bellan M, Gavelli F, Navalesi P, Avanzi GC and Chiocchetti A. “The Role of Osteopontin as a Diagnostic and Prognostic Biomarker in Sepsis and Septic Shock”. In: *Cells* (2019). DOI: 10.3390/cells8020174.

[63] Xu Z, Yang Y, Zhou J, Huang Y, Wang Y, Zhang Y, Lan Y, Liang J, Liu X, Zhong N, Li Y and Mao P. “Role of Plasma Calreticulin in the Prediction of Severity in Septic Patients”. In: *Dis Markers* (2019). DOI: 10.1155/2019/8792640.

[64] Zhou Y., Zhou B. and Pache L. et al. “Metascape provides a biologist-oriented resource for the analysis of systems-level datasets”. In: *Nat Commun* (2019). DOI: https://doi.org/10.1038/s41467-019-09234-6.