FACULTY OF MEDICINE AND HEALTH SCIENCES

Using reprocessed public proteomic data to detect cell line specific protein patterns

Sam van Puyenbroeck

Student number: 01805006

Supervisor: Prof. Dr. Lennart Martens
Mentor: Tine Claeys

A dissertation submitted to Ghent University in partial fulfilment of the requirements for the degree of Master of Biomedical Sciences

Academic year: 2022 – 2023

GHENT UNIVERSITY

1. Preface

These past two years have been the most captivating, yet challenging phase of my university education. I would like to take this opportunity to express my gratitude to all the individuals who have contributed to my accomplishments and provided unwavering support during this journey.

First and foremost, I am deeply thankful to my mentor, Tine Claeys, whose invaluable feedback, guidance, and consistent support have played a pivotal role in the successful completion of this thesis. Dr. Robbin Bouwmeester, I extend my gratitude for your tips and feedback on the deep learning aspects of this project, along with Arne Lescrauwaet, who was always ready to have fruitful discussions on data science and deep learning topics. I would also like to express my appreciation to Alireza Nameni for your valuable suggestions on the poster design. I am grateful to all the members of CompOmics for their warm welcome and kindness, which have made my experience even more enriching. Special thanks to Prof. Dr. Lennart Martens for the feedback and discussions, as well as providing me with the opportunity to pursue my master's thesis within the CompOmics group.

Furthermore, I extend my thanks to Marjoke Le Roy for always being a patient listener to my ideas and frustrations related to the master thesis. Also, thank you to my parents for their unwavering faith in me and their constant support throughout these two years.

Finally, I would like to thank my jury, Prof. Dr. Kris Gevaert and Prof. Dr. Sven Degroeve for their time to read this thesis and preparing the defence.

2. Table of contents

3. Abstract

Cancer cell lines are widely used in cancer research as a model system to study the aberrant pathways that give rise to cancer and to test the efficacy of cancer treatments. In this project, 43 PRIDE-projects are reprocessed and combined to build a model capable of accurately classifying cell line groups which through feature importance analysis can provide insights into which proteins are most discriminative for a group of cell lines. To build such a model, for each pre-processing step consisting of: (i) normalisation; (ii) imputation; (iii) feature selection; and (iv) oversampling; several methods were implemented and evaluated. Additionally, the systematic difference in protein identifications due to sample preparation was explored and whether or not correlations in our dataset can be predictive of functional associations. Our findings suggest that by performing in-gel digestion a more hydrophobic part of the proteome is identified than compared to in-solution digestion. Additionally, we found pairwise protein Pearson correlations between protein abundances to have a predictive value for functional associations. Based on our evaluations of the pre-processing methods, the optimal pre-processing pipeline included (i) quantile normalisation; (ii) limit-of-detection imputation; (iii) an ensemble of feature selection methods; and (iv) SMOTE. Subsequently, a Logistic Ridge Regression was trained and reached 93.7% classification performance. A preliminary biological exploration of the model showed slight concordance with the annotations made by the Human Protein Atlas. By exploring the model further and leveraging the correlations within the dataset, more insights can be gained into what makes cell line groups unique.

4. Layman summary with societal impact

Cancer continuous to be a prominent global cause of mortality, underscoring the importance of early detection and effective treatment. However, due to the intrinsic heterogeneity of cancer across patients, the same treatment does not work for everyone and a highly personalised treatment strategy is necessary.

In the realm of cancer research, surrogate model systems known as cell lines are frequently used to study the effectiveness of various compounds in treating cancer. However, due to the fact that cell lines are intrinsically different from the cancer subtypes of patients, the translation of compound effectiveness on cell lines to the clinic is highly inefficient.

In our study, we have tried to capture the differences between cell lines. This approach can be beneficial in two ways. Firstly, it can help to understand the biological diversity between cell lines on a systematic level. As cell lines model cancer subtypes, this could help to identify new biomarkers able to distinguish between cancer subtypes and thus make patient treatment more precise and lower the devastating effects of unresponsive therapy. Additionally, our approach could match a patient’s cancer subtype to the most closely related cancer cell line. By capitalizing on this resource, researchers could conduct more precise assessments of the efficacy of potential treatments and interventions on cell lines that closely mirror the patient's specific cancer subtype. This can enhance the predictive value of preclinical studies and lower the high costs associated with developing unsuccessful drug candidates.

5. Introduction

Since the end of the 20th century, the idea behind treating cancer patients with fit-for-all medication has changed radically due to the proven, limited efficacy of many compounds[1]. The main cause for the variable results of these compounds is the heterogenous nature of tumours between patients, which require a more targeted approach to be effective. This targeted approach, often referred to as precision medicine, will first stratify patients using staging systems, histopathology and multigene expression assays[2,3]. These screening methods indicate the tumour type of the patient which can differ in proliferation rate, invasion capabilities and resistance to certain therapeutics of the cells. The classification will then be used to determine the most optimal treatment strategy. In the case of breast cancer, treatment options are currently chosen based on molecular subtypes defined by varying expression patterns of specific hormone receptors and epidermal growth factors[3]. The patient stratification captures the so-called intertumoral heterogeneity between patient groups and, although far from comprehensive, has been able to improve prognostics and treatment efficacy[1]. However, large biological diversity is also apparent within individual tumours and is one of the most important causes of treatment resistance[2]. The phenomenon of intratumoral diversity is caused by genetic instability of cancer cells, which give rise to subclones with diverse mutations. The resulting cellular diversity within a tumour is further enhanced by complex interactions with the microenvironment and changes in microenvironmental factors[2,3]. Furthermore, cancer cells can acquire stem cell-like properties and have been demonstrated to promote intratumor heterogeneity[2]. These characteristics of cancer cells demonstrate the need for highly individualised treatment schemes guided by more clinical prognostic and therapeutic biomarkers to stratify patients and tumour subtypes. However, this will require a better and more detailed understanding of the biological differences between tumour types and their response to specific therapeutic approaches. For this purpose, model systems such as cancer cell lines are often used[4].

Cancer cell line models are the most used cancer model to study cancer biology and the efficacy of therapeutic effects. These models offer several advantages over primary cells, primarily due to their ease of culturing and ability to provide an unlimited source of biological material for high-throughput screening experiments[4]. However, despite their widespread use as cancer model, it is important to note that the *in vivo* microenvironment of a tumour contains various complex factors that cannot be fully replicated by an *in vitro* environment. Indeed, the *in vivo* microenvironment is very complex and is composed of many different kinds of cells, such as fibroblasts, immune, endothelial and other tissue specific cells which are embedded in the extracellular matrix[5]. Furthermore, tumour growth significantly impacts the surrounding healthy microenvironment. Immune cells infiltrate the stroma surrounding the cancer cells, creating a chronically inflammatory region. Together with the inflammatory cells, the malignant cells send pro-angiogenic signals to nearby endothelial cells, stimulating the formation of a new vascular network in the tumour microenvironment. Th resulting increased blood flow provides nutrients and oxygen necessary for growth and a means to dispose waste. Nonetheless, the core of the tumour remains inadequately vascularised, inducing the formation of a necrotic core. Necrotic cell death further promotes inflammation and the proliferation of neighbouring viable cells by the release of bioactive regulatory factors[6]. In these conditions of chronic inflammation, myofibroblasts further contribute to cancer cell proliferation, angiogenesis, invasion and metastasis[5]. Unsurprisingly, this highly complex and specific tumour microenvironment cannot be easily replicated *in vitro*. Indeed, cell lines are grown in an artificially composed medium optimised for each cell line for cell growth, proliferation and viability, resulting in culture conditions that are a crude simplification of the *in vivo* microenvironment[7]. Due to this large discrepancy, few tumour cells can adequately adapt to cell culture conditions and the ones that do are established as cell lines, meaning the full spectrum of tumour cells is not completely represented by cell lines.

Although the creation of cell lines is unpredictable and time-consuming, since the inception of the first cell line in the 1950's, numerous cell lines have been established[4]. The emergence of high-throughput technologies spurred large collaborative efforts to systematically analyse vast cell line panels ranging between 60 to 1000 cell lines representing 36 cancer types[8,9]. This has allowed to summarise in part the large cancer heterogeneity from a cancer model perspective with high-throughput technologies such as next-generation sequencing[10]. In one of these studies, Barretina et al analysed 947 human cancer cell lines with genomic and transcriptomic technologies, revealing that cell lines are somewhat representative of their respective primary cancer subtypes on the genetic level. Although this encourages translation of cell line biology to tumour samples, some precautions should be taken when selecting cell lines to study a specific cancer type.

Firstly, the annotations of cell line subtypes do not consistently align with their molecular subtypes. This discrepancy highlights the need for meticulous molecular characterization of cell lines when selecting them for studying a specific type of cancer[11]. For example, IGROV1 was seen to more closely resemble an endometrioid-like cancer profile than the high-grade serous ovarian cancer (HGSOC) it was previously presumed to be. This is based on a comparison of genome-wide copy-number changes and mutations between HGSOC tumours and ovarian cancer cell lines, which are combined to calculate a score of cell line suitability. Similarly, A2780 and SKOV3 did not compare well to HGSOC tumour samples, yet are the most used cell lines to study HGSOC. In contrast, the cell line models that compared best with HGSOC tumour samples are used the least, showing the cell line selection procedure is not routinely based on molecular characterisation. Secondly, it is important to consider that a cell line represents a heterogeneous population of genetically diverse cells[12]. In laboratory settings, when studying specific aspects of biology, cell lines are often transfected with plasmids. Subsequently, only the successfully transfected cells are selected, creating a genetic bottleneck event that reduces the genetic heterogeneity within the cell line population[13]. Further expansion of the selected cells can result in a cell line population that is genotypically distinct from the original cell line population. Similarly, alterations in culture conditions or exposure to compounds can lead to the same phenomenon[12]. These observations underscore the critical significance of thorough molecular characterization even within a single cell line, to ensure reproducibility in research. Thirdly, as previous points only considered the genetic component, it is crucial to acknowledge the limitations with solely relying on this type of characterisation. This limitation becomes evident through a comprehensive cell line classification study that incorporated multi-omics data[14]. The study revealed that the top contributing features for cancer subtype clustering predominantly originated from transcriptomics and proteomics data. By taking these three points together, it is evident that careful molecular profiling of cancer cell lines on multiple levels is of high importance.

An additional benefit of characterising cancer cell lines, apart from guiding model selection, is that key molecular signatures specific for cancer cell lines themselves, and thus indirectly the cancer subtype it is modelling, can be identified. These signatures encompass crucial biomarkers that have the potential to guide clinical decision-making and facilitating the personalised treatment schemes for patients. As stated above, first successful systematic and large-scale studies to identify biomarkers were focused on genotype-phenotype associations through genomic analyses[15]. Partly due to these studies, over 37,730 short nucleotide variants (SNV) are currently found associated to some disease[16]. Furthermore, these biomarkers not only allow to provide a measure of individual disease risk, they can also provide insights into disease biology by linking the identified genes to their function. However, inferring clear functional and mechanistic explanations from genomic data has proven to be less effective[15,16].

Although the potential of genomic data is undeniable, the findings are difficult to link to their biological and functional basis directly, which is essential for implementing the results in drug discovery pipelines and biomarker selection[16,17]. One of the main difficulties arises from the fact

that association does not imply causation, as it is the proteins, not the genes, that ultimately dictate function. Focusing solely on genomics often overlooks multiple layers of information essential for understanding biological mechanisms. Furthermore, it is crucial to recognize that most associations identified in genomic studies do not directly contribute to disease presentation. Instead, they may exert their influence downstream through functional connections until a limited set of core disease genes are modulated to an extent that leads to disease manifestation[18]. In fact, the omnigenic model posits that due to the highly interconnected nature of genes through regulatory networks, all expressed genes with regulatory variants contributing to pathogenesis can have a small yet significant impact on the dysregulation of core disease genes. Due to the large proportion of contributing genes, their total contribution of indirect effects on the core genes is relatively large[18]. Distinguishing the specific core genes that lie at the basis of the disease from possibly associated genes remains a major challenge that will require a more holistic approach characterising not only cancer subtypes based on genomics data, but also integrating data on more downstream levels, closer to the biological function.

To overcome these challenges, the fields of proteomics, interactomics and the study of post-translational modifications (PTM) are necessary[19]. These disciplines offer crucial insights into the functional aspects of biology that are often essential for drug targeting and can address the limitations of focusing solely on upstream data. While it is possible to infer protein levels from upstream data types, the relationship between different layers of expression is far more intricate than suggested by the traditional biological dogma[20]. This dogma assumes a one-to-one relationship between genes, their transcripts and their resulting proteins. However, the abundance of proteins can be impacted by various events. One example is ubiquitination of protein products that results in proteasome-mediated degradation which can be altered by specific protein interactions. Large scale experiments profiling both the proteome and transcriptome of 59 and up to 949 cell lines have highlighted this discrepancy[9,21-23]. These studies showed transcript levels to be in part poorly correlated with protein abundance in cell lines[9,21-24]. In the NCI-60 cell line panel study, 40.7% of the 3171 protein-transcript correlations were non-significantly correlated, including TP53[9]. Additionally, poorly correlated protein-transcripts included known subunits of protein complexes and proteins strongly linked to protein synthesis and degradation, which was stated to show the utility of proteomics to capture post-transcriptional regulation.

Another advantage of using proteomics instead of transcriptomics is that correlations between protein products can link proteins by function, more so than transcriptomics can as the latter is additionally biased by mRNA co-expression based on chromosomal proximity[25]. Naturally, previously mentioned core genes could be picked up by identifying proteins strongly correlating with many other proteins. In the CCLE proteomic study, protein correlations with important epithelial and mesenchymal markers, EPCAM and vimentin, were explored[21]. Half of the identified proteins were correlated either positively or negatively, and only half of these could be reproduced based on transcriptomics data. Additionally, EPCAM and vimentin were strongly associated with the first two components describing the variability in a protein expression dataset of 949 cancer cell lines[22], further corroborating and highlighting the potential of picking up the protein that affects a very large part of the proteome with proteomics. The same study explored the potential of identifying protein interactions through protein expression correlations[22]. Higher correlations indicated known interactions more[26]. These findings support the notion that protein expression is altered at a level downstream of gene expression and can capture crucial biological information not apparent in genomics and transcriptomics data.

From the previously mentioned large-scale proteomic cell line studies, cell lines originating from the same tissue clustered closely together, suggesting that tissue-specific characteristics are retained in cell lines. Furthermore, one study could indicate wrong annotations based on

proteomic data[23]. MDA-MB-435, previously believed to be a breast cancer cell line, showed more similarities with melanoma cell lines after performing unsupervised clustering. This finding proves to be correct because MDA-MB-435 is derived from the M14 melanoma cell line and thus wrongly annotated. Additionally, IGROV1 did not cluster with ovarian cell lines and was believed to also be a melanoma cell line. However, transcriptomics data indicates IGROV1 more closely resembles endometrioid-like cells[11]. This shows that selecting cell lines as model for a type of cancer can be guided by annotated tissue-of-origin data, however there are cell lines that do not corroborate with this. Therefore, it is crucial to classify individual cell lines in a more detailed manner. This will require a larger number of measurements of the same cell line in order to be robust as currently only a limited number of repeated measurement of individual cell lines are made in the described studies. Indeed, the generation of large sample sizes in proteomics is still limited due to technical limitations.

The availability of large datasets in public repositories presents an opportunity to expand the scope of cell line proteome analysis by reanalysing and merging datasets. PRIDE, the largest freely accessible repository, holds at the time of writing 22590 proteomic datasets, whereof 3150 of type cell culture and Homo sapiens[27]. With a continuous influx of at least 2500 dataset submissions each year since 2018, PRIDE's repository is steadily growing. These datasets encompass raw mass spectrometry (MS) data, enabling studies that involve reanalysing the raw data with more finetuned algorithms and integrating many projects together[24,28]. Although such studies are increasingly common and have demonstrated the added value of such endeavours, there are certain challenges associated with this approach that are inherent to bottom-up MS-based proteomics. In the following sections, these challenges will be discussed in a chronological manner starting from sample preparation to data analysis.

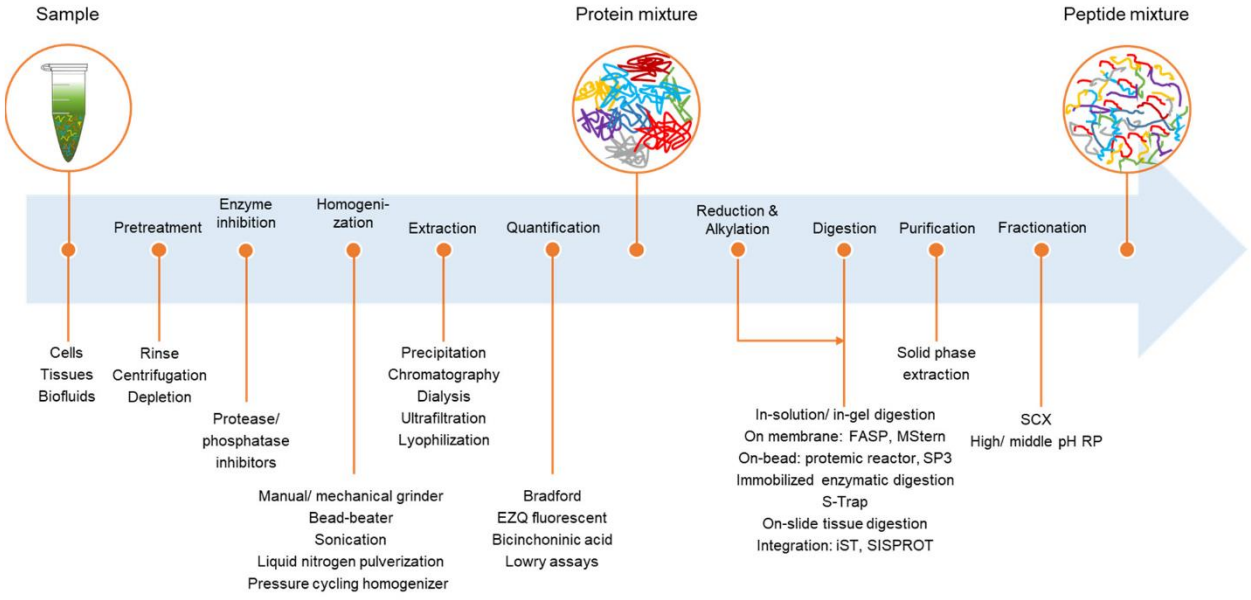


Figure I-1: Overview of sample preparation workflow in bottom-up proteomics[29].

In a typical MS-proteomics experiment (as depicted in figure I-1), several essential steps are involved[29]. Initially, cell lysates are generated by lysing cells. To prevent any potential endogenous enzymatic interference from lysate components, inhibitory measures are implemented. Lysation can be achieved through physical methods such as sonication and homogenization, or chemical means involving chaotropic agents like urea and guanidine hydrochloride, as well as detergents like sodium dodecyl sulfate and sodium deoxycholate.

Detergents also aid in protein precipitation[30,31]. Protein precipitation, coupled with decontamination, is crucial for obtaining high protein concentrations and eliminating non-protein substances such as detergents, buffers, DNA, or lipids, which would otherwise interfere with protein digestion, liquid chromatography (LC)-based separation, and introduce less clear spectra during MS analysis[32]. Multiple methods are available for removing specific contaminants, tailored to the nature of the contaminants involved. Once the sample is purified from contaminants, proteins are denatured to enhance proteolysis efficiency. This involves the reduction of disulfide bonds between cysteine residues, which contribute to the protein's secondary structure, followed by alkylation to prevent cystine reformation. The subsequent step involves enzymatic digestion of proteins into peptides, with the most commonly used enzymes being trypsin and LysC.

On each of the aforementioned steps, many adaptations exist and considering that many combinations of these steps are possible, this results in a wide range of possible sample preparation workflows. Broadly speaking, based on the environment the above mentioned steps are performed in, these approaches can be classified into in-solution, filter-, gel- and bead-based methods. The selection of specific sample preparation method can have an impact on the proteins that are identified with MS. This has been studied by Varnavides et al in the HeLa cell line, which showed a qualitative difference of the proteomes extracted attributable to buffer and precipitation methods used when performing Principal Component Analysis (PCA)[31]. This can be caused by changes in trypsin-based peptide cleavage efficiency, with detergents showing higher and chaotropic agents lower efficiencies. Additionally, only 61.6% of protein identifications was observed across all methods. Although it is true each sample preparation workflow has their own strengths and weaknesses to extract and purify a peptide mixture, the lack of protein identification overlap is not solely attributable to differences in sample handling. Instead the lack of overlap is in part a direct consequence of how the LC-MS/MS instrument works under data-dependant acquisition (DDA) settings. To understand the underlying factors contributing to this issue, the LC-MS/MS workflow is first briefly summarised.

In the LC part, the peptide mixture is separated based on physicochemical properties to minimise the number of different peptide species being ionised, fragmented and detected per time unit and thus reducing the spectral complexity in the MS/MS part[33]. The eluting peptides are ionised and a first mass spectrum (MS1) is acquired by the first mass analyser. The spectrum consists of peaks of different intensities separated by the distinct mass over charges (m/z) of the ionised peptide species in the eluent, also called precursor ions. In DDA, a predefined number of the largest peaks are selected to be fragmented in a collision cell, producing fragment ions which generates a new spectrum in the second mass analyser, the MS2 spectrum. The sequence of the peptide is inferred from the MS2 spectrum with the help of a search engine (detailed in the next paragraph). Two major components in this workflow contribute to the lack of overlap across LC-MS/MS runs: (i) the stochastic nature of precursor ion selection for fragmentation, resulting from the co-elution of multiple peptides in a single MS1 scan, leading to different highest intensity precursor ions across runs, and (ii) inaccurate peptide identifications and unidentified spectra. The first component is directly related to the instrument used. Indeed, reducing the sample complexity before introduction to the MS/MS and increasing MS1 scan speeds could decrease stochasticity in some sense[34]. However, sometimes more stochasticity is desired and enforced by choosing a different precursor ion selection method which excludes previously identified peptides[35]. At the cost of a reduced overlap between replicates, this stochastic sampling increases the probability of identifying peptides from low abundant proteins, which would otherwise remain unidentified. In cancer studies, where identifying peptide mutants is crucial as they often characterise the disease, stochastic sampling becomes particularly valuable. Mutated peptides are typically low in abundance and less likely to be selected for fragmentation. Moreover these peptides are often not included in the search space when matching peptides to spectra as will be discussed next[36].

To address these challenges, more targeted proteomics approaches have been developed, focusing on specific peptide precursor ions. However, such approaches do not provide a comprehensive view of the proteome as DDA does. Therefore, they are unsuitable for the purpose of this thesis, which aims to identify characteristic global protein patterns in cell lines.

When reanalysing public projects, the generated raw spectral data cannot be altered. Thus, variations in instrument type, resolution, scan speed, precursor ion selection (both in number and method), and other settings may introduce additional variability amongst the raw data files. It is only possible to address the second component attributing to the lack of overlap between projects: the inaccurate peptide identifications and unidentified spectra. This can be mitigated through the use of a more advanced search engine.

Search engines make meaning of the MS2 spectra by matching them with peptides[37]. In short, the experimental MS2 spectrum is compared to theoretical spectra of peptide ions which match the m/z of the MS1 peak. These spectra are generated by in silico digestion and fragmentation of a sequence database that consists of a large set of proteins (or peptides), often the annotated reference human proteome from UniProt[38], that can be present in the sample. Indeed, a specific peptide not present in the sequence database cannot be identified, such as peptide mutants resulting from cancer mutations as previously described. Because many peptides hold post-translational modifications, either in vivo or in vitro acquired, their MS1 m/z changes accordingly, also making them unidentifiable if these peptidoforms are not added to the search space. In open modification search engines such as ionbot[39], also possible amino acid modifications are included. This exponentially enlarges the search space, which increases the chance of finding false positive identifications, often represented as the false discovery rate (FDR)[40]. Therefore, open modification search engines require accurate peptide-to-spectrum match (PSM) scoring functions. Ionbot leverages LC retention time predictions of both modified and unmodified peptides, using DeepLC[41] and predictions of fragment ion intensities of the MS2 spectrum, using MS2PIP[42], to score PSMs. These predictions are data-driven and allow more accurate identifications of peptidoforms, while also allowing to identify more chimeric spectra. Chimeric spectra arise from co-eluting peptides, meaning one peak in the MS1 spectra originates from multiple peptide species and are often the cause of the unidentifiability of spectra. To allow public proteomic data repurposing studies, CompOmics has and still is reanalysing raw data from a subset of the PRIDE database, which enables the extraction of new patterns from the combination of data from numerous experiments.

The ultimate goal of a shotgun LC-MS/MS proteomics experiment is to capture a snapshot of the overall protein abundances of a sample. Indeed, peptides instead of proteins are identified, meaning peptide abundances are used as proxy for estimating protein abundance. However, care should be taken on the choice of peptides to use for quantification. Indeed, some proteins share the same peptide which can influence the quantification[43]. The methods developed to get an accurate estimation of abundance can be broadly subdivided in labelled and label-free methods.

Labelling methods offer the advantage to easily perform relative comparison of samples by multiplexing. Multiplexing is unique to labelled approaches and allows to simultaneously analyse multiple samples in one LC-MS/MS run. TMT or iTRAQ experiments involve labelling peptides from different conditions or sample types with isotopically diverse markers which are subsequently mixed together and loaded on the LC-MS/MS as one run. Fragmentation of the labelled peptides generates reporter ions that can be used to identify the corresponding sample and compare the abundance of the peptides between them. While labelled methods provide accurate relative protein abundance and low missing values within a single run, they are costly, require laborious sample preparation and carefully thought out experimental design and are

limited in terms of number of samples that can be compared[44]. Indeed, large scale analyses will require multiple TMT batches to be integrated. To control for variation between batches which presents itself as more MVs and quantification variability across batches compared to single batch TMT experiments, a common control sample is added to each batch which serves as normalisation reference. When repurposing projects, it is very unlikely that common reference samples are present to normalise on which makes integration difficult. In contrast, label free quantification methods allow comparability with unlimited samples, making them more suitable for large repurposing studies[33].

Many methods for label-free quantification have been developed, which use different assumptions. They can be largely subdivided into two classes: (i) spectral counting and (ii) intensity-based quantification methods[33]. Spectral counting methods assume the precursor ions of more abundant peptides are more likely to be selected for fragmentation and thus generating more PSMs. Summing the observed spectra of the peptides that point to a protein is used as a proxy for protein abundance. Although a simple concept, several factors unrelated to protein abundance affect the number of peptides that can be identified and the way these are integrated in the quantification estimation differs between the spectral counting methods. In NSAF, the spectral counts are divided by the length of the protein, because larger proteins can generate more peptides[45]. Then, this value is further normalised using the sum of these previous values for the entire run, to allow the comparison of protein abundances across runs. Many more spectral counting methods exist, which differ from NSAF in their estimation of theoretical peptides a protein can generate. In contrast, intensity-based quantification methods use the MS1 signal intensity as proxy for peptide abundance by assuming the MS1-signal intensity correlates with ion concentration[43]. The acquired peptide abundances can be summarised in multiple ways to estimate protein abundance.

To be able to compare quantifications across samples and correct for sample loading differences, references are needed to scale the quantifications to the same unit. Indeed, reference proteins are often spiked-in at known concentrations and are used to derive a calibration curve[43]. Absolute protein abundances of all quantified proteins are subsequently inferred from this calibration curve. However, for reanalysing public data, the presence of spiked-in reference proteins are not guaranteed, meaning a protein standard free method is required[46]. Such methods exist and assume direct proportionality between estimated protein abundance and total protein amount[43]. This is achieved by dividing the calculated abundance value by the sum of all calculated abundance values by assuming that this sum is a proxy of total protein loaded.

Multiple methods exist for calculating protein abundances from LC-MS/MS raw data, each with their own assumptions and related dis- and advantages. When evaluating the most suitable quantification method, three criteria should be considered: (i) accuracy of the quantification, (ii) the ability to correctly identify proteins that are statistically significant altered in abundance and (iii) reproducibility among replicates[47]. While MS1 intensity-based method provide a non-discrete abundance value, and thus should be able to more precisely define peptide abundances, simpler quantification methods based on spectral counting such as NSAF, were deemed comparable in performance based on the three described metrics in a comparative study[47]. Therefore, NSAF is chosen as a suitable quantification method to use for a combinatorial analysis of proteomic projects

Besides the choice of quantification method, uncorrected technical variability often remains[48-50]. Therefore, normalisation is necessary to eliminate any technical bias, often called batch effects, that is still systematically present in the data. Similar to label-free quantitative methods, many normalisation methods with different assumptions on the underlying bias exist[50]. Median normalisation assumes samples are separated by a constant and corrects this by scaling the

samples to equal medians. Another very popular normalisation method, quantile normalisation, assumes every sample follows the same distribution and through a ranking based workflow (described in section 6.7.1) forces this on the dataset. ComBat is a more complex batch correction method, which is specifically designed to correct interlaboratory batch effects[51]. This method assumes that the factors responsible for batch effects affect many proteins in a similar way. By using an Empirical Bayes approach, the parameters used for standardisation are estimated on the data. A plethora of other normalisation methods exist which assume that the bias have a linear or non-linear relationship with the protein abundances which can be corrected by the use of regression models[50].

The choice of normalisation and its inherent assumptions, largely impact the downstream analysis. An objective decision of normalisation method can be based on several quantitative and qualitative metrics including Pearson correlation of technical replicates, boxplots, correlation plots, median versus standard deviation plots amongst others[48,50]. An evaluation of different normalisation methods based on these metrics showed that the optimal method is highly dataset dependant. Indeed, if most proteins are upregulated compared to another sample, median equalisation will not be optimal as its assumption is violated. Linear and non-linear regression methods also pose specific assumptions on the nature of the bias. In data repurposing studies, many projects are combined and the type of bias present is less clear to determine. Therefore, normalisation methods such as quantile normalisation could be more preferable due to their less strict assumptions about the underlying bias[50].

Normalisation can only reduce technical bias which presents itself in differences between abundance values. However, in proteomics data, many values are simply missing and this needs to be handled. Indeed, proteomic data, severely more so than transcriptomics data, is affected by many missing values, typically ranging between 20-50 % on the peptide level[44,52]. The cause of missing values in label-free proteomics is diverse: (i) stochastic peak-picking characteristic for DDA as described above[34,35], (ii) sample preparation biases that unintentionally filters out some proteins or peptides[30,31], (iii) misidentified or unidentified spectra[39], (iv) undetectable peptides due to low abundance[40] and (v) proteins biologically not present in the sample if samples of different origins are combined in the data analysis pipeline. These types of missing values are bundled in the literature in two large groups: missing completely at random (MCAR) and missing not at random (MNAR)[52]. MCAR missing values are characterised by their true random missingness, such as causes i and iii, whereas MNAR missing values are related to an underlying characteristic which in proteomics is low abundance or true missingness of a protein. If these missing values are left unattended, downstream analysis such as machine learning modelling is not possible[53].

To meet the challenge of missing values, imputation methods were developed. These methods fill in the missing values by a number inferred from the data or chosen by the data analyst. Several types of imputation methods exist and according to Webb-Robertson et al, can be largely subdivided in three categories[54]. The first is single-digit replacement, such as mean/median or zero-imputation. The zero-imputation method is an example of an MNAR-imputation strategy and can be refined by sampling from a left-shifted gaussian distribution, simulating protein abundance measurements that fall below the detection threshold. The second category imputes missing values based on local similarity in the dataset. The most well-known examples are k-nearest neighbour (KNN) imputation and local least-squares imputation, which assume similar peptide or protein intensity profiles in the dataset are biologically explainable and can be leveraged for imputation[54]. A final category of imputation methods use dimensionality reduction to capture global structure in the data and iteratively reconstruct missing values based on the reduced dimensions[55]. Indeed, imputation is a large field on its own and with many methods available, the selection of the most suitable method is not straightforward. Nonetheless, based on the ratio of

MNAR versus MCAR in the dataset, some imputation methods perform better[52]. In a study from Lazar et al, datasets were simulated to have missing values of MNAR and MCAR types in predefined proportions[52]. Several imputation methods were evaluated based on the accuracy of imputing the correct value. Generally, KNN and dimensionality reduction imputation strategies perform better than MNAR methods when the MNAR ratio is below 70%[52,54]. However, it is noted it is more important to choose the most adequate method for the type of missing values than choosing the method itself, as the wrong method can be detrimental for downstream analysis[52,54].

After the crucial steps of normalization and missing value imputation, a large matrix of thousands of protein features with hundreds of samples is retained. In machine learning-based classification workflows, a model is trained to accurately learn to recognise distinct patterns of protein expression, i.e., features that can differentiate one type of sample from another[53]. However, most shotgun proteomic datasets identify thousands of proteins in only a limited number of samples, making the set of features an order of magnitude larger than available training examples. This is commonly referred to as the 'curse of dimensionality'. This phenomenon poses significant challenges for machine learning algorithms, making models overly complex and prone to overfitting[56]. When models are overfit, small and insignificant changes in features, typically resultants of bias are conceived as important by the model. In severe cases, the individual samples the model is trained on are memorized, leading to very inaccurate predictions when confronted with unseen yet biologically identical samples. This is also called a lack of generalisability. To combat this issue, the dimensionality must be reduced to only retain the most informative features which are able to classify samples accurately[56]. Therefore, many techniques, called feature selection methods, have been developed to reduce the feature space and improve generalisability.

Feature selection methods aim to only keep the most useful information in the dataset and drop the rest. The methods can be largely subdivided in two main categories: feature subset selection (FSS) and feature extraction (FE)[56]. FSS-methods aim to select a subset of the original features, which can be achieved in several ways. The most simple methods, filter methods, use statistical tests to evaluate the most informative features in relation to the label either feature by feature (univariate) or by taking interactions between groups of features into account (multivariate). In more complex approaches, a learning algorithm is used to determine which subset of features produces the most accurate results by iterating over the possible feature subsets[56]. Indeed, in large dimensions, the number of available subsets is tremendous, making them computationally inefficient. Therefore, other strategies to select subsets are available for example, adding or removing one feature at a time based on which feature increases the accuracy of the learning algorithm the most[57]. Another type of FSS-method, embedded methods, leverages the behaviour of learning algorithms such as LASSO logistic regression and random forests to select features. Indeed, these algorithms make the weights of irrelevant features zero (or very close to zero) and therefore ignores them during classification. By ranking the features based on their importance for classification, features can be selected.

FSS-methods only select a subset of features, thus removing possibly little yet useful information described by the dropped features. Interestingly, a recent study from Shi et al leverages the complete dataset to put the selected features in their biological context[58]. By using unsupervised learning algorithms on the transposed data matrix, features with similar patterns across samples can be grouped[58]. Indeed, proteins with similar expression profiles are clustered whereof the medoid, i.e., the most central protein in a cluster, can serve as the protein representative. The key assumption is that co-expressed proteins share similar biological functions[58]. Based on this assumption, the subset of features most important for classifying a certain sample can be functionally interpreted, which indirectly leverages the dropped data during feature selection. A more direct method to preserve information from all features are FE-methods, which reduces the

dimensionality by summarizing all features in a smaller set of components by leveraging patterns in the complete dataset. Popular approaches include principal component analysis (PCA) and t-stochastic Neighbour Embedding (t-SNE)[59]. One drawback of these approaches in contrast with FSS-methods that retain the original features, is that interpretation of the decision function of the model becomes more difficult, due to the abstraction of many features into one during dimensionality reduction.

Feature selection tackles the curse of dimensionality by removing redundant and unrelated features. Although useful, the causative problem remains: a lack of data. Additionally, dataset sizes are diverse and many cell lines are understudied, which upon meta-analysis often result in an imbalanced datasets: many more samples of one type are present than others. This imbalance can influence classifier performance, which will focus on accurately classifying the majority class at the cost of ignoring harder to predict minority samples. This is due to the fact that because there are few minority samples, these will not to minimising the cost function during training[60]. This issue can be resolved in multiple ways, where cost-sensitive and oversampling approaches are the most utilised methods[61]. In the cost-sensitive method, larger misclassification costs are assigned to minority samples, thus emphasising correct classification of minority samples during model training. In a similar fashion, duplicating minority samples with resampling approaches achieves the same goal of emphasising minority samples. Although simple and effective in mediating class imbalance, replicating minority samples tends to result in overfitting since the model needs to memorise only a limited number of samples to achieve maximum performance[60]. Additionally, if minority samples overlap with the majority class, defining a generalisable decision boundary remains hard. Therefore, oversampling methods generating synthetic data resembling the minority classes can alleviate these issues. The most used method of this type is Synthetic Minority Over-sampling TEchnique (SMOTE), which generates new samples by interpolation of two neighbours of k-nearest neighbours from the same class[62]. In terms of feature vectors, this translates to a generation of a sample randomly along the line segment connecting two minority samples . Doing this would ideally make the decision regions of the classifier larger and less specific towards one sample, which improves generalisability to new samples. Although often successful, SMOTE does not perform optimal in scenario's where dimensionality is high and minority samples group in small clusters[63]. Therefore, many extensions have been developed which include removing noisy generated samples, choosing the samples to interpolate on in different manners and reducing the dimensionality prior SMOTE.

Recently, generative models such as Variational Autoencoders (VAE) are proposed as promising alternatives outperforming the traditional oversampling methods while providing additional advantages[64]. VAE's are composed of an inference network, which is trained to learn an appropriate distribution of a small set of latent variables given the input data and a generative network, trained to reconstruct the input data from the latent variables. The model is trained by minimising the summation of the log-likelihood and the Kullback-Leibler (KL) divergence, called the evidence lower bound. These terms have contrasting effects: the log-likelihood, often represented as the mean-squared error ensures the reconstructed sample is similar to the input, while the KL-divergence acts as a regulariser limiting the latent variables to take values following a prior distribution, typically the Normal distribution. Indeed, these two terms should be weighted carefully, as too much regularisation would result in suboptimal reconstruction while the lack of regularisation would result in a non-normal latent space which becomes difficult to sample from[65]. Despite this, VAE's have shown to be successful in finding a biologically meaningful latent representation of gene expression profiles of different cancer types[66]. Consequently, by using the generative network to reconstruct a new sample from an instantiation of the latent manifold, a limitless number of synthetic yet biologically similar samples can be generated to improve the balance in the dataset.

Once all these issues are addressed, machine learning algorithms such as Support Vector Machines (SVM), Logistic Regression (LR) and Decision Trees (DT) can be used to classify cell lines. These algorithms are trained on a dataset and learn an optimal decision boundary by minimising a cost function in the case of SVM and LR, or use a measure of impurity such as in decision trees when splitting samples based on a logical expression of a feature. The cost functions and impurity measures utilise known labels associated with the training data, making these models supervised classification models. By minimising the cost function, the optimal model parameters are learned, which in a broad sense means that the model learns to identify an expression pattern of proteins to accurately classify a certain cell line. When provided with a new, unknown sample the model can predict the cell line identity by using the learned objective function.

However, although inferring cell line identity from unknown samples can be useful, it is often more important to know why the model classifies a sample as a certain class. When using LR, the prediction is made by a logistic transformation of the weighted sum of the feature variables, making the model directly interpretable by analysing the assigned weight. However, interpreting the model is not always directly inferable from the model parameters. When using non-linear kernels with SVM, the input features are transformed into a higher-dimensional space, making it challenging to directly interpret the contribution of the original features to the prediction. Therefore, model-independent methods have been developed to determine the contribution of features to each prediction[67]. The most notable one was developed by Lundberg et al, named SHapley Additve exPlanations (SHAP), which extends on the work of Lloyd Shapley. This framework determines the feature importance of a feature by calculating the overall impact the addition of this feature has on the prediction. Using such methods on a trained cell line classifier can uncover what makes each cell line unique and potentially find proteins that are of key importance to generate the cancer cell line specific phenotype.

In this project, supervised machine learning will be applied on a large set of PRIDE projects, which analysed the proteomes of cell lines with a label-free approach. After a laborious metadata annotation initiative, these shotgun DDA proteomic projects are reanalysed with the ionbot search engine to ensure accurate peptide identifications. Subsequently, the proteins are label-free quantified with the NSAF-method. After careful exploration of batch effects and correction thereof as well as the application of feature selection and oversampling methods, multiple machine learning algorithms are trained and the most performant one is optimised on the collected data. By using SHAP, the trained model is interpreted to acquire a set of characteristic protein features for each cell line used in this study. These are subsequently biologically contextualised through the integration of ontology data, protein-protein interaction data and the literature. With the results of this project, we hope to show the potential of large-scale data integration for cell line selection and characterisation as well as highlight significant limitations of this workflow.

6. Materials and Methods

6.1. Data collection

6.1.1. PRIDE data

Suitable PRIDE projects were selected based on several criteria. Projects should have focussed on the analysis of the cell contents of human cell lines with label-free methods. Additionally, no enrichment procedures can be employed to ensure a global proteomic representation of the cell line. The method of fragmentation of the MS/MS instrument in the study should be of type 'high-collision dissociation'. This is to ensure compatibility with the ionbot search engine.

To find PRIDE projects, a list of previously ionbot-reprocessed PRIDE projects was evaluated. Additionally, through a web scraping script, more PRIDE accessions were extracted that were linked to cell line accessions in Cellosaurus[68]. All collected PRIDE accessions were manually checked for suitability based on the above-described criteria.

6.1.2. Metadata annotation

Metadata annotation of the collected PRIDE data was performed on the raw file level. To acquire the correct annotation, metadata from PRIDE, the material and methods section of the corresponding research articles and the supplementary materials were used. Additional metadata to identify the original tissue of origin of the cell lines was acquired from Cellosaurus. The type of disease the cell line models was acquired from both Cellosaurus and the vendor product pages including, ATCC, Sigma-Aldrich and Thermo Fisher Scientific. Added metadata includes (i) cell line name, (ii) tissue of origin of the cell line, (iii) disease type (iv) sample preparation protocol, (v) MS-instrument type, (vi) ionbot version number and if applicable, (vii) experimental treatment, (viii) fraction number and (ix) sub cell line name. Based on the disease type of the cell line, the cell lines were grouped in 15 distinct groups. The annotation file and grouping file can be found in the Master Thesis GitHub repository (https://github.com/SamvPy/Master_thesis_AJ_22_23) in the 'Metadata' folder (annotation file: 'unify_metadata.csv'; grouping file: 'group_cells_annotation.csv').

6.1.3. Additional data files

UniProt sequences and descriptions were downloaded from UniProt (September 2020). Gene ontology data (GO version: 2022-07-01) of Homo Sapiens (taxon: 9606) was downloaded from the Gene Ontology page as a GAF-file[69]. In order to filter probable contaminants from the identified protein list, UniProt protein identifiers from 'common Repository of Adventitious Proteins' (cRAP) were extracted. Data on cancer-related proteins and proteins holding prognostic value were downloaded from The Human Protein Atlas (THPA)[70] (proteinatlas.org). Protein-protein interaction data was downloaded from the STRING-database[71] (version 11.5).

6.2. Ionbot reprocessing

Using ionbot, the selected public datasets were reprocessed. The spectra in these datasets were searched against the reviewed UniProt proteome of Homo sapiens, appended with 5 common contaminants. The cleavage pattern is fixed at K/R with a minimal peptide length of seven and maximum of 30 amino acids. Additional settings are: (i) fragmentation method 'HCD', (ii) quantification label 'No label', and (iii) precursor error tolerance of 20 ppm. Additionally, carbamidomethyl on cysteine was set as a fixed modification, and oxidation on methionine as a variable modification. An open modification search was performed. From the ionbot output file,

only the best scoring peptide to spectrum (PSM) matches (PSM=1) were kept, with q-value ≤ 0.01. Non-proteotypic peptides were removed, and spectral counts for each peptide were calculated.

Due to the timing wherein the projects were collected and the continuous development of the ionbot search engine, several ionbot versions were used. Version numbers include 6.2, 6.3, 7 and 8. The ionbot version used for each project was added to the previously mentioned annotation file ('unify_meta.csv'). For 379 raw files used in this project, result files of the same raw file, yet from multiple ionbot versions were present. This allowed to measure variability caused solely by differences in ionbot version. For each result file, the spectral counts were extracted and used to quantify proteins as described in section 6.4. The Pearson correlation coefficient was calculated on the NSAF values for each combination of ionbot versions of the same raw file. All statistical tests were performed with scipy (version 1.4.1).

6.3. MySQL database

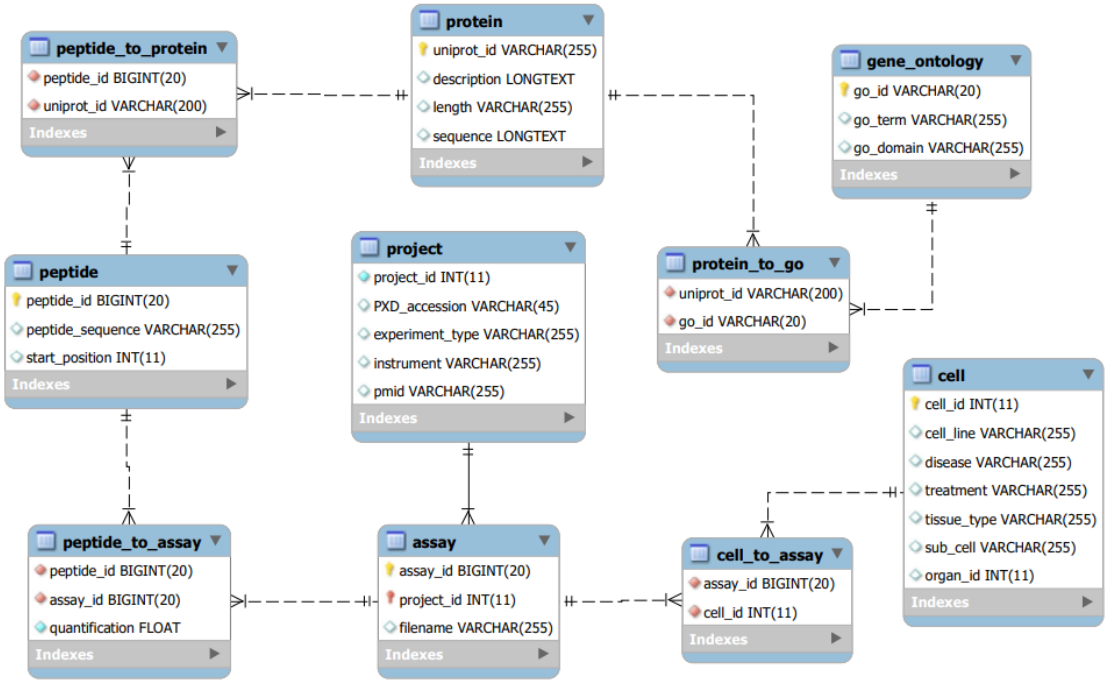


Figure M-1: MySQL Workbench schema of the custom-built cell line database

A custom MySQL database was built to make access of data linked to each raw-file easier in a relational manner (Figure M-1). Each project is linked to a raw-file in the database, which is denoted as an assay. Extra metadata on what the assay represents is added as described previously in the metadata section. Each peptide identified by ionbot is stored and linked with the corresponding assay. Ionbot uses the 'best-scoring-peptide-for-each-protein' method implemented in Percolator 3 for protein inference and only stores protein inferences passing the 1% protein false discovery rate (FDR) threshold (https://ionbot.cloud/help). Proteins which are similar are grouped to reduce non-proteotypic peptides which are not used for quantification. These protein inferences are stored in the database and linked appropriately. Additionally, protein information from UniProt including protein length and sequence, is used to complete the protein table and further linked with functional biological information on gene ontology.

6.4. Label-free quantification and clustering analysis

Before quantification was performed, the peptide and protein tables stored in the database were filtered on several criteria. (i) Only proteotypic peptides were retained to prevent ambiguous quantification by utilising shared peptides. (ii) all proteins in the list of protein contaminants from cRAP are removed. (iii) Only proteins with more than two (proteotypic) peptides were retained. To start quantification, all spectral counts for each assay were extracted from the database.

For protein quantification, the Normalised Spectral Abundance Factor (NSAF) was used and is calculated as follows[45]:

$$(NSAF)k = \frac{(SpC_k / L_k)}{\sum_{i=1}^{N}(SpC_i / L_i)}$$

With SpC and L the spectral count and length of protein k in assay i, and N the total amount of quantified proteins in assay i.

This metric normalises for protein length by dividing the spectral counts of a protein by its length. Additionally, this quantification method normalises protein abundances across each assay which reduces sample-to-sample variation.

For samples using an offline fractionation approach, the spectral counts of each of the fractions from one sample were aggregated. This combined assay was subsequently interpreted as one run whereon the NSAF could be calculated. To evaluate the soundness of this approach, a scatterplot of the first two principal components after fitting Principal Component Analysis (PCA) on a select subset of quantified samples before and after pooling was used. The resulting NSAF-matrix is referred to as the PEMatrix.

6.5. Sample preparation comparison

The PEMatrix was used to compare the hydrophobic character of the samples between three sample preparation protocols (in-solution, in-gel and on-filter). To calculate the hydrophobic character, first hydrophobicity scores for proteins and peptides were calculated as follows:

$$GRAVY_i = \frac{\sum_{k=1}^{n} AA_k}{L_i}$$

Where for each protein or peptide sequence i, the hydropathy values AA of each amino k in the sequence is summed and divided by the length of the sequence Li.

The used hydropathy values were determined by Kyte and Doolittle[72] and the resulting value is termed the Grand Average of Hydropathy (GRAVY) of a protein. Next, for each sample, the NSAF of a protein was multiplied by the corresponding GRAVY score and summed for all proteins. For a peptide-centric representation of hydrophobicity of a sample, the GRAVY score of a peptide was multiplied with the spectral counts, summed for all peptides identified and divided by the total spectral counts in the sample. Each sample represents all the spectral counts across raw files from all fractions of one sample.

To statistically determine whether there is a difference in hydrophobicity caused by sample preparation, the Kruskal-Wallis H-test test was performed where each value in the different groups is the calculated hydrophobicity score of a sample.

6.6. Data exploration

To explore the data, clustermaps were generated using seaborn (version 0.11.2) by calculating the overlap of protein identifications between two assays and normalizing it by the maximum number of protein identifications from the compared assays. For visualizing the dataset in two dimensions, PCA (Principal Component Analysis) and t-SNE (t-Distributed Stochastic Neighbor Embedding) techniques were employed with scikit-learn (version 1.0.2), using perplexity values ranging from 15 to 20.

6.7. Pre-processing

Assays with less than 1100 protein identifications were dropped to limit the effect of very large number of missing values in the dataset. For similar reasons, the proteins were filtered based on 50% occurrence of a protein in all the assays in the dataset, as was used in Jarnuczak et al[24]. Groups with less than 10 assays were dropped. The resulting dataset is referred to as fPEMatrix.

6.7.1. Normalisation

Four datasets were generated with different normalisation methods. The first dataset was generated by log2-normalising all NSAF values in the PEMatrix. This dataset is referred to as the NSAF-dataset. The other three datasets were generated independently starting from the NSAF-dataset.

The generation of the second dataset, referred to as the “median-normalised dataset”, involved several steps. This method assumes most proteins are not systematically over- or underexpressed, resulting in an equal median protein abundance across samples. Initially, a protein list was created, consisting of proteins commonly identified in 90% of all samples. Using this set of proteins, for each sample the median and standard deviation was calculated. Lastly, every protein abundance was transformed by subtracting the calculated median for that sample and dividing by the standard deviation.

To generate the quantile-dataset, quantile normalisation was applied to the NSAF-dataset. This normalisation approach involved assigning ranks to the NSAF values of each sample, with a rank 1 representing the highest NSAF value within that particular sample. Missing values were disregarded during this process. To obtain the reference values for normalisation, all protein abundances per sample were sorted, and the column-wise median was calculated. These reference values were ranked accordingly and used to substitute the NSAF values by mapping the reference rank with the ranks assigned per sample.

Finally, a well-established batch correction method called pyComBat[73] was used to generate the ComBat-dataset. The PRIDE project identifier was assigned as batch. Because pyCombat cannot handle missing values they were handled by mean imputation on protein columns prior fitting the pyComBat algorithm and reset after transforming the dataset.

6.7.2. Imputation and scaling

Several forms of missing value imputation, both MNAR- and MCAR-methods, were implemented and the best performing one was selected based on the methodology described in section 6.8.1.

One MNAR method was implemented: imputation by sampling from a rescaled normal distribution with the mean defined as the mean of the protein abundances of all samples subtracted by 2 times the standard deviation. The standard deviation of the distribution is the standard deviation of the protein abundances of all samples multiplied by 0.3. This method is from now on referred to as limit-of-detection (LOD) imputation. Imputation was performed on the protein level.

Two MCAR imputation methods were implemented. (i) KNN-imputation based on the k-Nearest Neighbours (KNN) algorithm implemented in the sklearn package as 'KNNImputer' and (ii) PCA-based imputation.

For KNN-imputation, 10 nearest neighbours and a weighted distance metric were used as hyperparameters.

For PCA-based imputation, a similar workflow as described in the paper by McCoy et al[55] was followed. In short, first every feature in the NSAF-dataset was rescaled with the MinMaxScaler and missing values were imputed in the first iteration by mean imputation. Next, PCA was used to reduce the dimensions of the data that retains 95% of the variance. The data is reconstructed based on these principal components and rescaled. Missing values of the original PEMatrix are imputed with the reconstructed values. This loop was iterated for 15 times while keeping the number of principal components as defined in the first iteration the same. The number of iterations and whether to update the number of principal components used to reconstruct the missing values for each iteration were optimized based on the mean squared error of the non-missing values between the original and PCA-reconstructed data.

Finally, an approach combining MNAR and MCAR imputation was implemented. For proteins identified in less than 20% of samples in a certain group, above described LOD-imputation was used. The idea is that if less than 20% of samples identified this protein, this is caused by non-random missingness, i.e., limitations in MS sensitivity. For the remainder of missing values, either KNN- or PCA-based imputation was fitted on the dataset after MNAR-imputation.

Differences in imputation value estimation were explored by calculating Pearson correlations between- PCA and KNN-methods, with or without using MNAR-imputation for each implemented normalisation method.

After normalisation and imputation, features were scaled with the MinMaxScaler class from the sklearn package.

6.7.3. Feature selection and correlation clustering

To get a rough estimate of the approximate features to select, a recursive feature elimination class with built-in cross validation that defines the optimal number of features was used with three machine learning algorithms: (i) Logistic LASSO regression (L1-LR), Random Forest (RF) and Support Vector Machines (SVC) implemented in scikit-learn with default settings. For each iteration, 50 features were deleted, and the performance was measured with the macro F1-score. The macro F1-score, along with other metrics used to evaluate model performance, are described more in detail in section 6.8.1.

Five feature selection methods were implemented: two univariate filter methods, two wrapper methods and one embedded method. As univariate methods, ANOVA and mutual information were used to establish a ranking of most informative features based on the calculated scores. Based on this ranking, features are selected. As wrapper-methods, SVC and RF were used with a recursive feature elimination strategy. L1-LR was used as embedded method. All above-described methods were implemented in the scikit-learn python package.

A final selection of features was achieved by imputing the quantile-dataset with LOD-imputation. Following this, all feature selection methods selected the top 300 features. Features were selected if 3 out of 5 methods choose the feature as important.

Correlation analysis was performed on the quantile dataset after LOD-imputation. First, all pairwise Pearson correlations were computed. These were compared with annotated protein-protein interactions from the STRING database. According to the thresholds defined by String,

combined interaction scores of 900, 700 and 400 were chosen as to indicate interactions of the highest, high or medium confidence respectively. The ratio of confirmed interactions was calculated as a percentage of the pairwise correlations which are annotated as a protein-protein interaction over all pairwise correlations that have a higher correlation than a defined cut-off value.

Hierarchical correlation clustering was performed on the same dataset. First a distance matrix was computed by subtracting the absolute values of pairwise correlations with 1. Subsequently, hierarchical clustering was performed with Ward linkage. For the validation of identified clusters, the proteins from the respective clusters were queried in the STRING database either directly or through the API.

Venn-diagrams were made with the matplotlib-venn library (version 0.11.7) and Upset plots with the UpSetPlot library (version 0.6.1). All other plots were created with seaborn and matplotlib (version 3.1.3).

6.7.4. Handling class imbalance

Three SMOTE-based methods, one data augmentation method based on Variational Autoencoders, and one weighting-based method for dealing with class imbalance were implemented. The synthetic minority over-sampling technique (SMOTE) was used either alone or in combination with the Tomek links or Edited Nearest Neighbour under-sampling techniques. All three classes were used from the imbalanced-learn package. The Variational Autoencoder architecture is described in section 6.9 and is used by passing the original data through the network and using the reconstructions as new samples. This was performed either until all classes have the same number of samples as the majority class or 20 times this number for validation purposes. Finally, a weighting-based method was used by assigning higher class weights to classes with less samples. The weight of a class is calculated by dividing the total sample count by the product of the number of classes and the number of samples in the class. These weights were passed to the learning algorithms when fitting the models.

The statistical tests used are the Levene test, Kolmogorov-Smirnoff test and t-test to evaluate significant differences for variance, distribution and a shift in means of the features respectively. All p-values were corrected by Bonferroni correction. All tests were performed using the SciPy package.

6.8. Machine learning modelling

Four types of machine learning classifiers were used: logistic regression (LR), support vector machines (SVM), random forest (RF), and LightGBM. LR, SVM, RF were used from the scikit-learn package.

6.8.1. Model selection and hyperparameter optimisation

Several types of hyperparameters were optimized including the imputation strategy, oversampling method and model-specific hyperparameters for the top three performing models. For each type of hyperparameter, two types of cross validation methods were utilised. The first is a stratified k-fold cross validation (SKFCV) method with 10 splits. In the second method, the dataset is split by selecting one complete project as test set, termed leave-one-project-out cross validation (LOPOCV). This was done for projects which ensured at least 4 samples remained in the training set. This condition was met for 35 projects only.

Two metrics were used to evaluate classification performance: (i) the macro-averaged F1 score and (ii) weighted-average F1 score for SKFCV and LOPOCV respectively. Macro F1-score is calculated as follows:

$$Macro\ F1-score = \frac{\sum_{i=1}^{n} \frac{TP_i}{TP_i + 0.5 * (FP_i + FN_i)}}{n}$$

Where n is the number of classes and TP_i, FP_i, FN_i are the number of true positives, false positives and false negatives for class i respectively. The weighted-average F1-score is calculated by weighting each class F1-score with the number of occurrences of samples in that class. The reason why the weighted-average F1-score is used for LOPOCV is that most of times a project has samples from a limited number of classes only.

To estimate the minimum features required for adequate classification performance, recursive feature elimination was performed with the RFECV class from the scikit-learn package, using the model-based feature selectors. For each step, 50 features were dropped and a fivefold cross validation was performed after each step using the macro-averaged F1-score as performance metric. The minimum number of features required was visually determined.

For imputation method selection, five imputation strategies were compared on all four normalised datasets. These include LOD imputation on all missing values, PCA-imputation, KNN-imputation, PCA-LOD imputation and KNN-LOD imputation. Each feature of the imputed datasets were subsequently scaled in a range between 0 and 1. Features were selected that are returned by at least 3 out of 5 feature selection methods described in section 6.7.3.

For finding the optimal feature selection method each method choose 100 features and the performance was evaluated similarly as described for imputation method selection.

6.9. Variational Autoencoder data augmentation

Two variational autoencoders were optimised and trained separately on the quantile dataset (VAE1) and the feature selected quantile dataset (VAE2). VAE1 consists of the following layers mentioned in chronological order: an input layer of 2615 neurons, 1 hidden layer with 500 neurons with ReLu activation, a layer containing the latent distribution parameters which consist of 10 pairs of nodes representing the variance and mean of the 10 latent distributions. These are reparametrized to 10 nodes by using the following equation:

$$z = \mu + \sigma * \varepsilon$$

where z is the reparametrized node, μ the mean, σ the standard deviation, and ε a random number drawn from a normal distribution. These 10 latent variables are followed by another hidden layer of 500 neurons with ReLu activation and the output layer containing 2615 neurons with a sigmoid activation function. VAE1 was trained over 1000 epochs with a learning rate of 0.001 and batch size 10 using the Adam optimiser. These hyperparameters were optimised by trying different latent space dimensions (5, 10, 15, 20, 30, 50), batch sizes (5, 10, 15, 20) and learning rates (0.0005, 0.001, 0.0025, 0.005, 0.01) over 100 epochs with 80% of the data. The rest was used for validation. The loss function consists of the Kullback-Leibler divergence and the mean-squared error which were averaged over batches during training. VAE2 has a similar architecture with different number of neurons in each layer. The input layer is equal to the dimensions of the dataset after feature selection with a hidden layer of 50 neurons and latent dimension of 6. In VAE2, batch size was set to 5, learning rate to 0.001 and kappa to 0.1. Kappa is multiplied with the KL-divergence when computing the loss and acts as a balancer between the KL-divergence and MSE. VAE2 was trained for 500 epochs.

6.10. Biological interpretation

SHapley Additive exPlanations (SHAP) was used as primary tool for interpreting the model. The tuned model was trained on 80% of the dataset, which was randomly split once, and then passed to the SHAP linear explainer object to compute the conditional expectations. This process involved an examination of how the inclusion or exclusion of each feature affected the predictions made by our model. Subsequently, the model was used to predict the classes for the remaining 20% of the dataset, and SHAP-values were computed specifically for the correctly predicted samples. Based on these SHAP-values, we calculated the average SHAP-value for each feature within each class. To biologically validate the most important features, we cross-checked them with available annotations in THPA and consulted relevant literature. To determine the most important features overall, we summed the absolute values of these averages across all classes.

To investigate the tissue-specificity of cell line group-specific features, we leveraged a tissue-classifier that was previously trained and validated [28]. This classifier assigned importance scores to features based on their relevance to classify specific tissues.

7. Results

7.1. Selected data

407 PRIDE projects were found by scraping the Cellosaurus site for cross-references to PRIDE of human cell lines. This number was reduced to 107 projects by excluding projects not satisfying the selection criteria described in section 6.1.1. Finally, 63 PRIDE projects containing a total of 5860 raw-files were found to be readily reprocessed with ionbot and were subsequently manually annotated on the raw-file level as described in section 6.1.2. In total, this included 371,091,640 spectra whereof 65,534,626 were selected as significantly identified (q-value < 0.01) best peptide-to-spectrum matches (PSM). Unfortunately, reprocessing of 1446 raw-files (25%) did not produce any protein identification, which is also reflected by the lack of PSMs of those raw-files (**Figure 1**). Indeed, some projects failed in the reprocessing pipeline due to incompatibility of the fragmentation type with the ionbot search engine, which were wrongly included during the annotation process (**Supplementary figure 1**). Ultimately, 4,414 raw-files were successfully reprocessed. From these 4,414 raw-files, only 389 (9%) contained whole lysate samples. The remaining raw files contained fractionated samples. These individual fractions were combined during protein quantification, as mentioned in section 6.4, resulting in 621 combined samples. A validation of this approach is described in section 7.2. The combined samples are referred to as samples throughout the text.

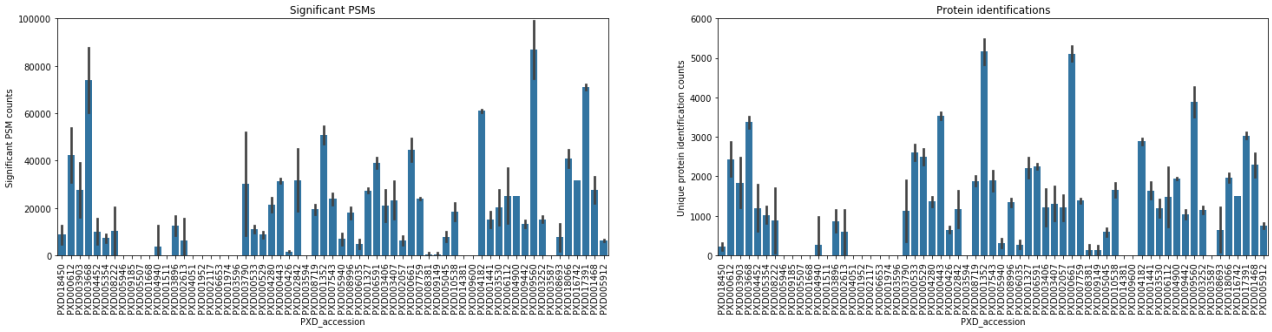


***Figure 1**: Barplots of significant PSMs (left) and unique protein identifications (right) counts per project. Error bars on the bars indicate the standard deviation of the PSM and unique protein identification counts of all the samples in one project.*

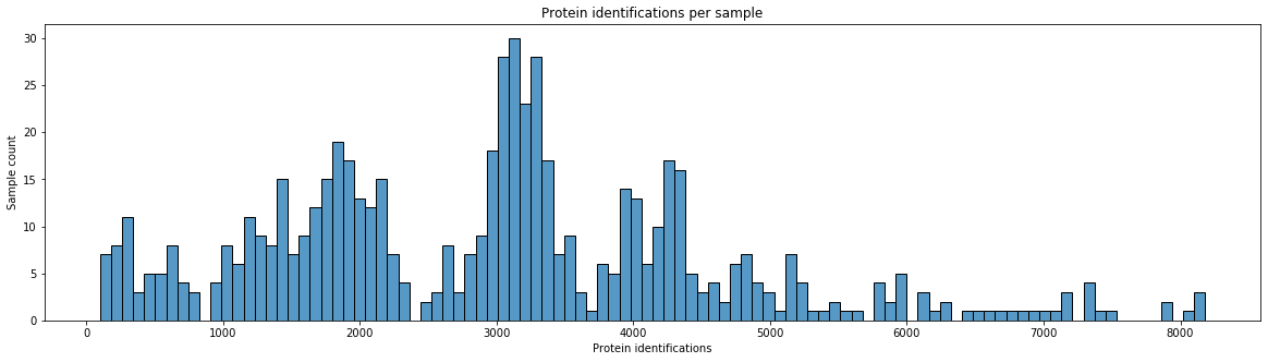


***Figure 2**: Histogram of protein identifications per sample. The proteome coverage ranges between 100 and 8180 protein identifications over all 621 samples.*

Proteome coverage across assays varied considerably, ranging between 100 and 8180 unique protein identifications per sample (**Figure 2**). Whenever an assay with less than 1100 protein identifications was observed, the corresponding research article was consulted to identify the

potential reason for this lack of proteome coverage. Generally, a large disparity between the reported and our observed proteome coverage was apparent. The most striking example is PXD009149 which reported in total 4519 protein identifications, while we only report 1512 unique protein identifications whereof not a single sample identified over 400 unique proteins. PXD002842 reported 3721 protein identifications from control and treated cells while we only identified 2265 in total and no more than 1612 proteins per sample. As a last example, also projects with a larger number of protein identifications were discordant to our findings: PXD003790 reported 7555 unique protein identifications while we only found 3966 with on average 2160 unique protein identifications per sample. In order to prevent low coverage samples to affect downstream analysis, a cut-off was set on 1100 protein identifications, reducing the dataset from 621 to 557 proteome samples originating from 44 projects.

Due to the low number of proteome samples per cell line, cell lines were grouped according to similar cancer subtypes or other cell line characteristics such as induced-pluripotent stem cells and HEK293 for non-cancerous cells guided by the available annotation in Cellosaurus ('group_cells_annotation.csv' on GitHub). If the annotation was ambiguous, the cell line was chosen as group in itself. This resulted in 28 classes whereof 15 classes included at least 10 samples. This 10 samples per class cut-off is deemed necessary to enable a 10-fold cross validation strategy during model performance analysis. The samples from the other groups are left out from training and used to test the cell line group classifier. Ultimately, 518 samples subdivided in 15 categories were retained and used for downstream analysis. An overview of the sample count and project counts per class is shown in **Figure 3** and illustrates the imbalance of class distribution on both sample and project level.

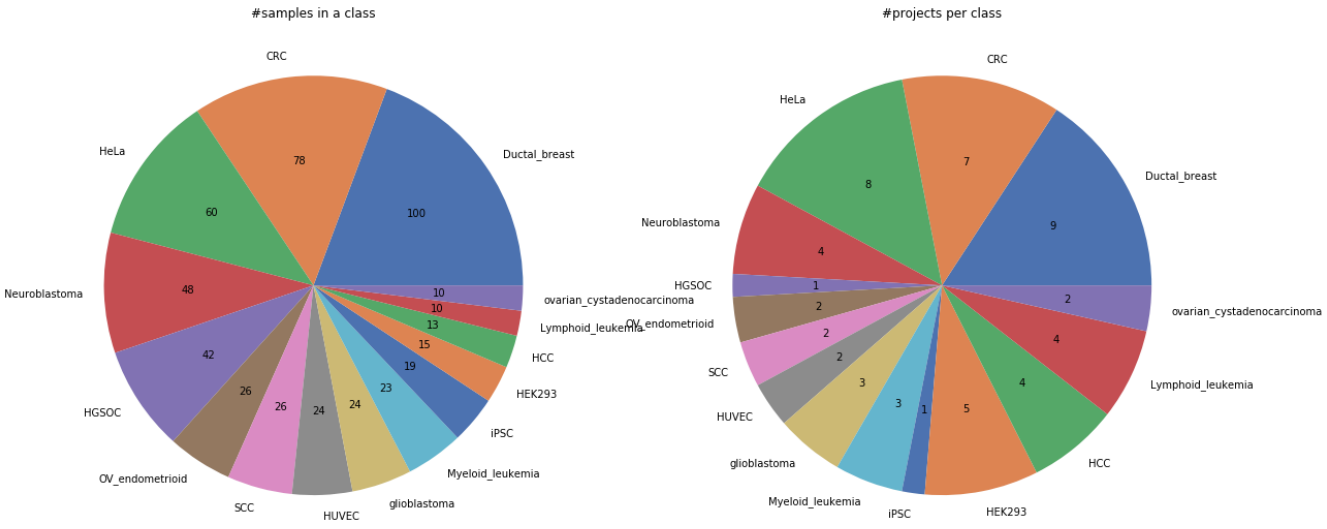


***Figure 3**: Sample (left) and project (right) counts per category. A large disparity can be observed of the number of samples and projects represented in each category. This shows class imbalance is present.*

The proteins from the 518 samples were quantified with the NSAF-method as described in section 6.4, which resulted in a protein expression matrix (PEMatrix) of 14506 unique proteins. Due to the large proportion of missing values (**Table 1, Supplementary figure 2**), proteins present in less than 50% of all samples were dropped. The filtering step was based on a similar cut-off as applied in Jarnuczak et al [24]. Applying this filtering on our dataset effectively reduced the missing values from 77.8% to 24.1% and reduced the unique proteins from 14506 to only 2615 proteins. Indeed, most proteins (56.4%) are only present in less than 10% of all samples.

***Table 1**: Missing value statistics before and after filtering proteins based on presence in 50% of all samples*

	PEMatrix	fPEMatrix
Number of samples	518	518
Number of unique proteins	14506	2615
Average missing values in a sample	77.8%	24.1%
Most missing values in a sample	93.0%	70.9%
Least missing values in a sample	43.6%	0.4%

In summary, careful selection of protein expression data reduced the collected data to a matrix of 518 samples with 2615 unique proteins. All downstream analyses as will be discussed in the remainder of the results section will be performed on the resulting fPEMatrix, unless stated otherwise.

7.2. Pooling fractions improves compatibility

The effectiveness of pooling fractions into one sample to improve compatibility with other non-fractionated samples was evaluated on PXD003406 and PXD003407. Both PRIDE projects originate from a study of Gerner et al, in which the HUVEC cell line proteome was analysed as in-solution digests without prefractionation and in-gel digests with four fractions, generating one raw-file for each fraction[74]. This experimental set-up allowed us to validate the effectiveness of our simple pooling workflow, by using the unfractionated samples as controls.

Raw-files of fractions of samples are obtained by first applying a protein separation method based on physicochemical properties of proteins and subsequently loading each fraction separately on the LC-MS/MS. In contrast, unfractionated in-solution samples present all proteins in one raw-file, making the results of these two types of raw-files inherently incomparable. Indeed, most proteins identified in the unfractionated sample are also independently identified across the fractions (**Figure 4**). Interestingly, both the unfractionated sample and the fractions identified many different proteins which can be attributed to differences in protein isolation when using in-solution versus in-gel digestion. This effect is further explored in section 7.3.1.

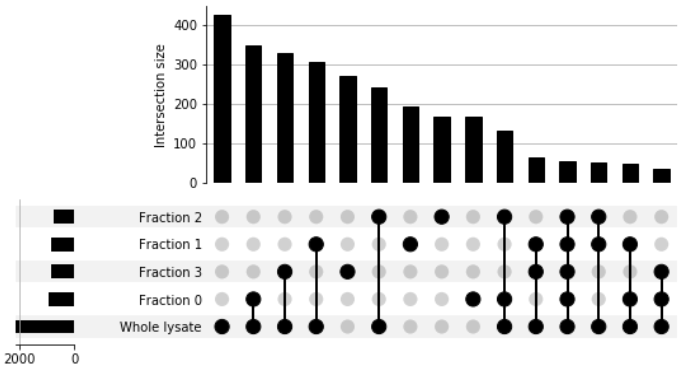


***Figure 4**: Upsetplot of commonly identified proteins for the fractions of 1 sample and an unfractionated sample (whole lysate).*

Two biological replicates, whereof two technical replicates each, were analysed with off-line fractionation and in-gel digestion. Surprisingly, between biological replicates a discrepancy in fractionation efficiency was observed. In biological replicate 1, on average 19.8-23.6% of proteins identified in each of the four fractions are commonly identified in all fractions, while for biological

replicate 2 only 7.3%-8.2% overlap in all fractions was observed. To observe the effect of this incomplete protein separation on protein quantification, the protein abundances from proteins commonly identified in all fractions in biological replicate 1 but not commonly identified in biological replicate 2 were plotted (**Figure 5**). This figure suggest, albeit limited, that lower abundant proteins that are identified in every fraction can be overestimated when aggregating the spectral counts during protein quantification.

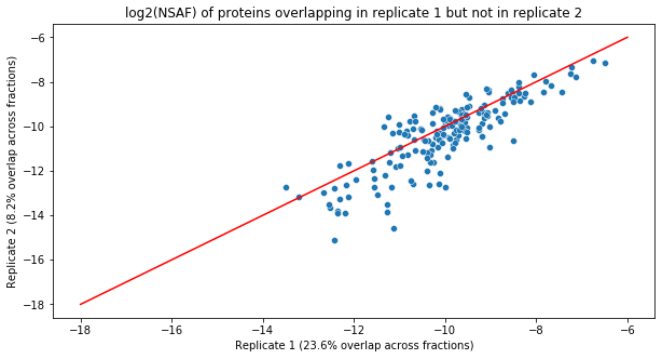


***Figure 5**: Scatterplot of protein abundances of proteins identified in every fraction of replicate 1 but not in every fraction of replicate 2. The red line indicates a one-to-one relationship. A small fraction of medium abundant proteins seem to be overestimated in replicate 1*

Fractions were aggregated on the raw-file level before quantifying the proteins with NSAF as outlined in section 6.4. Three types of samples are now available: (i) fractions, (ii) whole lysates and (iii) pooled samples. To evaluate improved comparability after pooling, principal component analysis (PCA) was performed on all three types of samples with the filtered proteins in the fPEMatrix. As expected, the same fraction index for each sample clustered together (**Figure 6**). Interestingly, the pooled and whole-lysate samples clustered together indicating improved comparability after reconstitution of the fractionated samples.

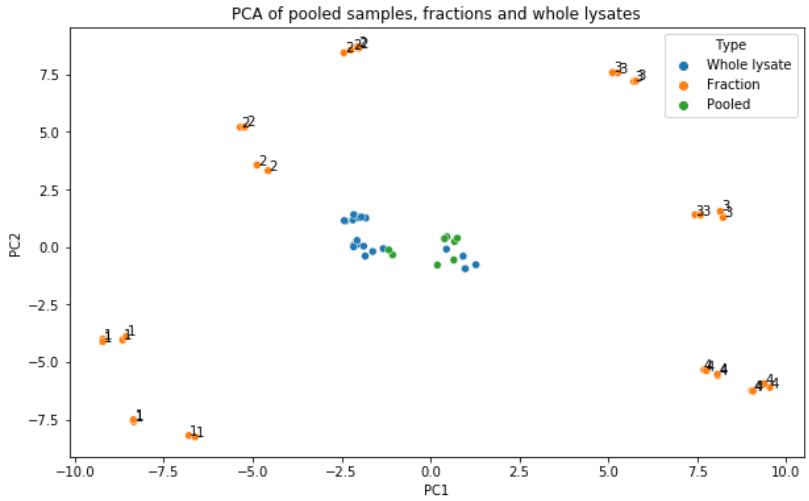


***Figure 6**: Scatterplots of the first 2 principal components of fractions, whole-lysates and pooled samples. Numbers indicates the fraction index of the samples. As 8 samples each separated the sample in 4 fractions, each number is represented 8 times. The cells were either untreated or treated with interleukin 1-beta (IL-1b).*

7.3. Sources of variability

The dataset variability was explored from three different perspectives: (i) sample preparation; (ii) ionbot version and; (iii) project. To ensure thorough analysis, the PEMatrix was used for the first two analyses. For peptide-level sample preparation comparison, all peptides used during NSAF-quantification are used. The analysis on project level was examined solely on the fPEMatrix.

7.3.1. Sample preparation

We compared three sample preparation workflows in terms of sample (i) hydrophobicity, (ii) distribution of molecular weight of peptides, and (iii) peptide length. The three sample preparation workflows include in-solution, on-filter (FASP) and in-gel digestion. The sample counts for each method are shown in **Figure 7 (left)**.

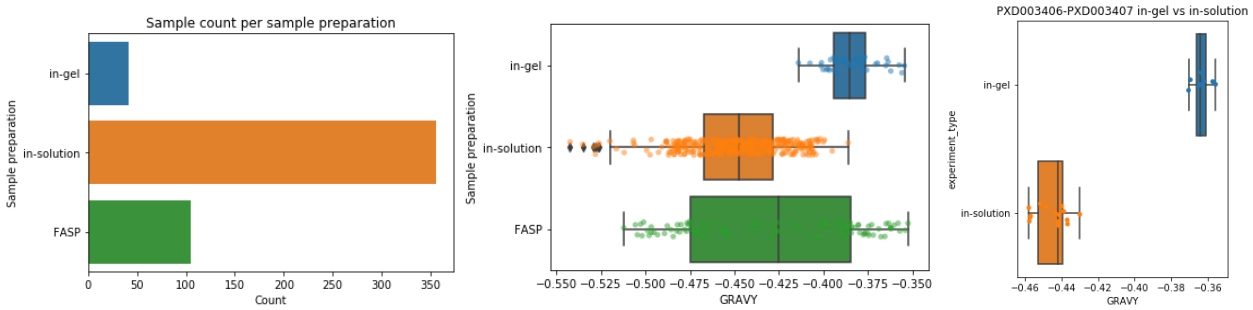


***Figure 7**: (left) Sample counts for each method. (middle) Sample hydrophobicity based on protein GRAVY-scores for all samples for each sample preparation. (right) Same as middle but for projects PXD003406 and PXD003407 only.*

The first comparison on sample hydrophobicity was calculated on protein and peptide level in terms of GRAVY-score following the methodology outlined in section 6.5. with a high GRAVY-score indicating increased hydrophobicity. Statistical analysis using the Kruskal-Wallis test revealed significant differences in hydrophobicity of samples among the sample preparation methods. Specifically, in-gel digested samples exhibited significantly higher protein hydrophobicity compared to both in-solution (p-value: $3.9*10^{-25}$) and on-filter (p-value: $6.2*10^{-6}$) digested samples (**figure 7 middle**). These results indicate a loss of hydrophilic proteins during in-gel digestion. It is worth noting that despite the presence of a larger number of in-solution digested samples, none of them reached higher hydrophobicity scores than on-filter and in-gel digested samples. This suggests that there might be a loss of more hydrophobic proteins during the in-solution sample preparation process.

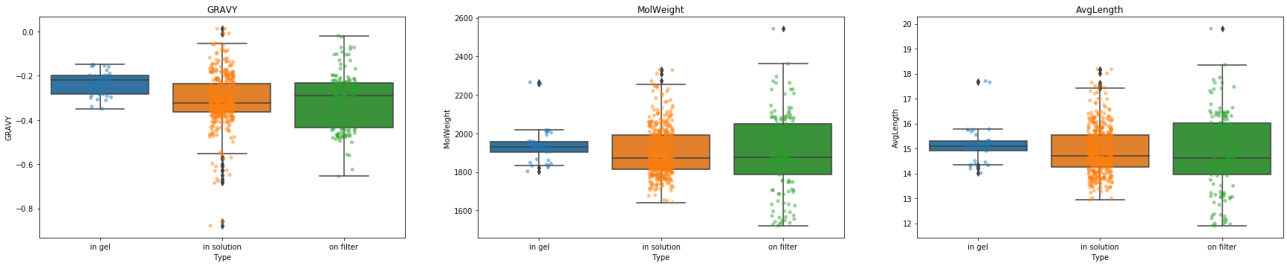


***Figure 8**: Comparison of (left) hydrophobicity, (middle) molecular weight and (right) length of peptides between sample preparation workflows.*

A similar trend, although less significant (p-value: 6.3*10-7), was observed when comparing hydrophobicity on peptide level between in-gel and in-solution sample preparation methods (**figure 8 left**). Marginal differences were observed in the average molecular weight and length of

peptides identified in the samples across the different sample preparation workflows (**figure 8 middle and right**). However, it is worth noting that the range of average molecular weight and length of peptides in on-filter digested samples was considerably larger compared to the other two methods.

7.3.2. Ionbot version

Due to the use of multiple ionbot versions for peptide identifications, an assessment of the potential differences between these versions was deemed necessary. For 379 raw-files included in the study, ionbot output from ionbot version 6.2 (V6), version 7 (V7) and version 8 (V8) was found and the difference was measured in terms of Pearson correlation after NSAF-quantification.

V7 and V8 were very comparable in terms of Pearson correlation (**Figure 9**). In contrast, V6 seemed less comparable to V7 and V8. The mean correlation between the same samples was 0.896 (+- 0.7) and the lowest correlation was 0.5. Indeed, some samples showed a lack of comparability when analysed with V6 and V7/V8.

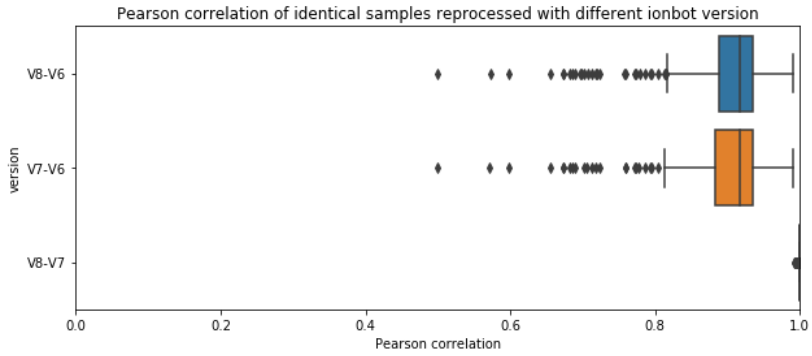


***Figure 9**: Boxplots containing Pearson correlation coefficients of the same raw file reprocessed with different ionbot versions.*

7.3.3. Project

Interlaboratory bias was explored in three different ways: (i) visual inspection of hierarchical cluster maps of protein identification overlap between samples, (ii) boxplots of protein abundance (NSAF) values in samples, and (iii) dimensionality reduction-based clusters.

Protein identification overlap between two samples was calculated by dividing the number of commonly identified proteins by the maximum protein identifications of the sample pairs. Hierarchical cluster heatmaps were created for each group. For the sake of brevity, only heatmaps for the largest groups (HeLa and ductal breast cancer cell lines [ductal_breast]) are shown in **Figure 10**. A large protein identification overlap is mainly observed between samples from the same project. This is not a surprising finding considering the total number of protein identifications is more constant within then between projects, which can be inferred from the smaller error bars in comparison to the height difference between the bars in **Figure 1 (right)**.

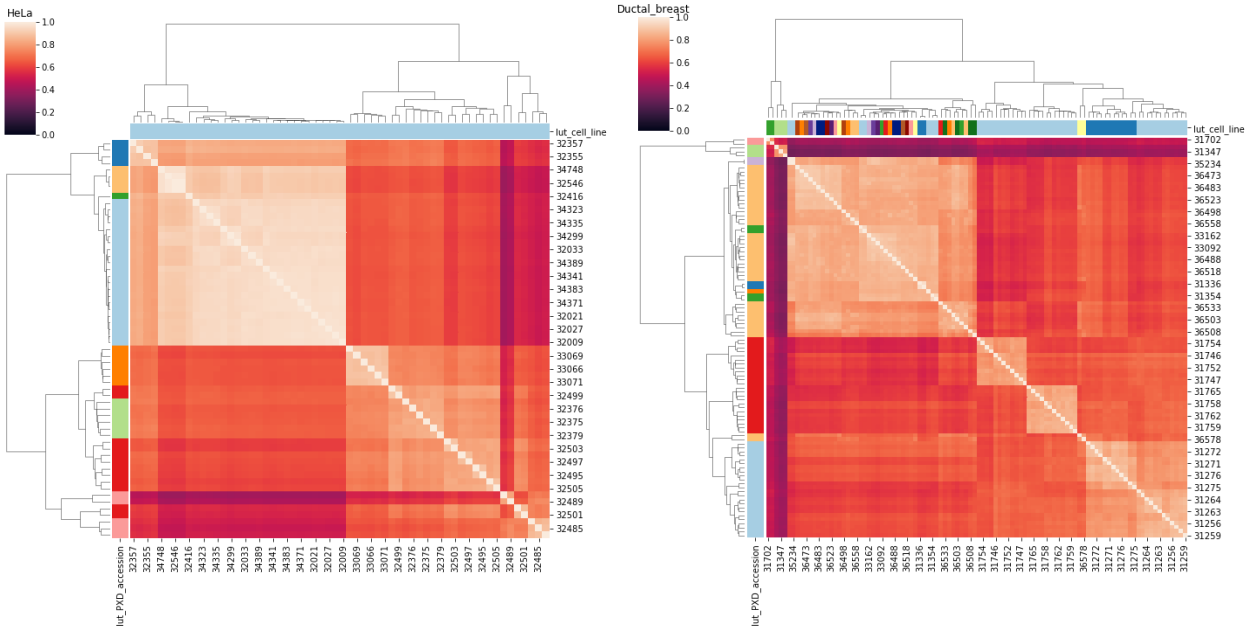


***Figure 10**: Clustermap of protein identification overlap for (left) HeLa and (right) ductal breast cancer cell lines. The y-axis colors indicate projects while the x-axis colors indicate cell line identity.*

The size of the set of identified proteins in part and indirectly reflects the dynamic range of the proteome analysis. Analyses with smaller proteome coverage tend to detect only the most abundant proteins due to the DDA-based peak picking. As described in the introduction, several sample preparation and instrumentation characteristics can enhance the detection of a larger part of the dynamic range of the proteome, more specifically the low abundant proteins. In our dataset, samples have a large diversity in the number of identified proteins, and thus have large differences in the range of abundance values between samples. Surprisingly, also the quantification value of the most abundant protein is negatively correlated with the depth of the proteome analysis (summarised in **Table 2**). To test whether the NSAF-values systematically underestimate the abundance of proteins in samples with many protein identifications, the correlation was calculated on two subsets of the dataset: (**i**) proteins identified in 90% all samples and (**ii**) the 100 most abundant proteins. For the first subset, the assumption is that most of the commonly identified proteins are not expected to systematically be higher or lower between cell line groups. For the second subset, the assumption is that regardless of depth of analysis, the most abundant proteins will lie in the same abundance range. For both subsets, a significantly negative spearman correlation was measured (**i**: -0.55; p-value: $5.6*10^{-43}$ and **ii**: -0.56; p-value: $4.1*10^{-45)}$. This indicates the NSAF tends to underestimate the abundance of proteins in high proteome coverage samples.

***Table 2**: This table shows the spearman correlation (p-value in brackets) between either the maximum, minimum or median abundance value (in log2-NSAF) of each sample and the number of protein identifications in the sample.*

	PEMatrix	**fPEMatrix**
Proteins	14506	2615
Maximum abundance (spearman correlation, p-value)	-0.31 ($3.0*10^{-13}$)	-0.25 ($8.9*10^{-9}$)
Minimum abundance (spearman correlation, p-value)	-0.87 ($3.8*10^{-157}$)	-0.82 ($8.5*10^{-129}$)
Median abundance (spearman correlation, p-value)	-0.93 ($2.6*10^{-229}$)	-0.75 ($2.3*10^{-94}$)

Finally, interlaboratory bias was explored by dimensionality reduction with PCA **(Figure 11**). Although clusters for each group are for the most part well separated, multiple clusters for the same group are apparent which appears to coincide with project membership. When performing PCA on samples belonging to the groups with the most projects (HeLa and DBCCL), indeed strong project clusters are apparent (**Supplementary figure 3**). This observation suggests systematic bias might be present on the project level. In the next section, methods to mediate these forms of bias through normalisation are explored.

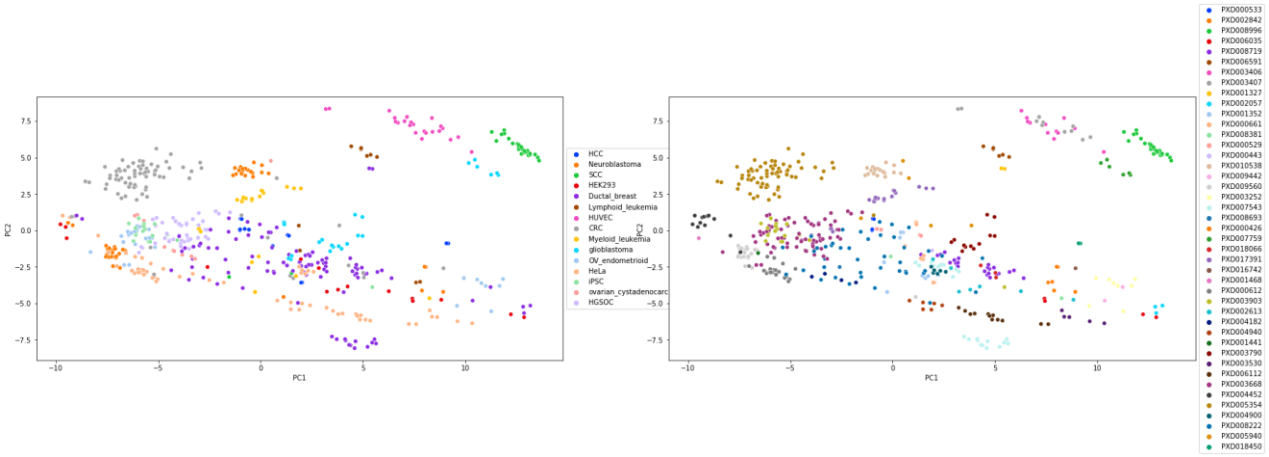


***Figure 11**: Scatterplot of the first two principal components after PCA for the fPEMatrix coloured by (left) group membership and (right) project of origin.*

7.4. Pre-processing

7.4.1. Normalisation

Extra normalisation can reduce systematic bias when batch effects are present. Three normalisation methods were applied on the dataset with each their inherent assumptions. To evaluate the effect of the normalisation method, (i) the distribution of abundance values per sample (**Figure 12**), (ii) correlations between samples from the same and different projects per group (**Supplementary figure 4**) and (iii) the correlation between median protein abundance for a subset of the proteins and number of protein identifications (**Table 3**) are used. To maintain the flow of the text, the table and figure are shown in this paragraph.

***Table 3**: Spearman correlations between median log2-NSAF of the indicated subset of proteins and number of protein identifications*

Normalisation method	**Proteins in 90% of samples Spearman correlation (p-value)**	**Top 100 abundant proteins** Spearman correlation (p-value)
NSAF	-0.55 ($5.6*10^{-43}$)	-0.57 ($4.1*10^{-45}$)
Equalise medians	**-0.03 (0.5)**	0.52 ($7.1*10^{-37}$)
Quantile normalisation	-0.54 ($1.4*10^{-40}$)	**-0.04 (0.3)**
ComBat	-0.48 ($1.2*10^{-30}$)	-0.33 ($2.9*10^{-94}$)

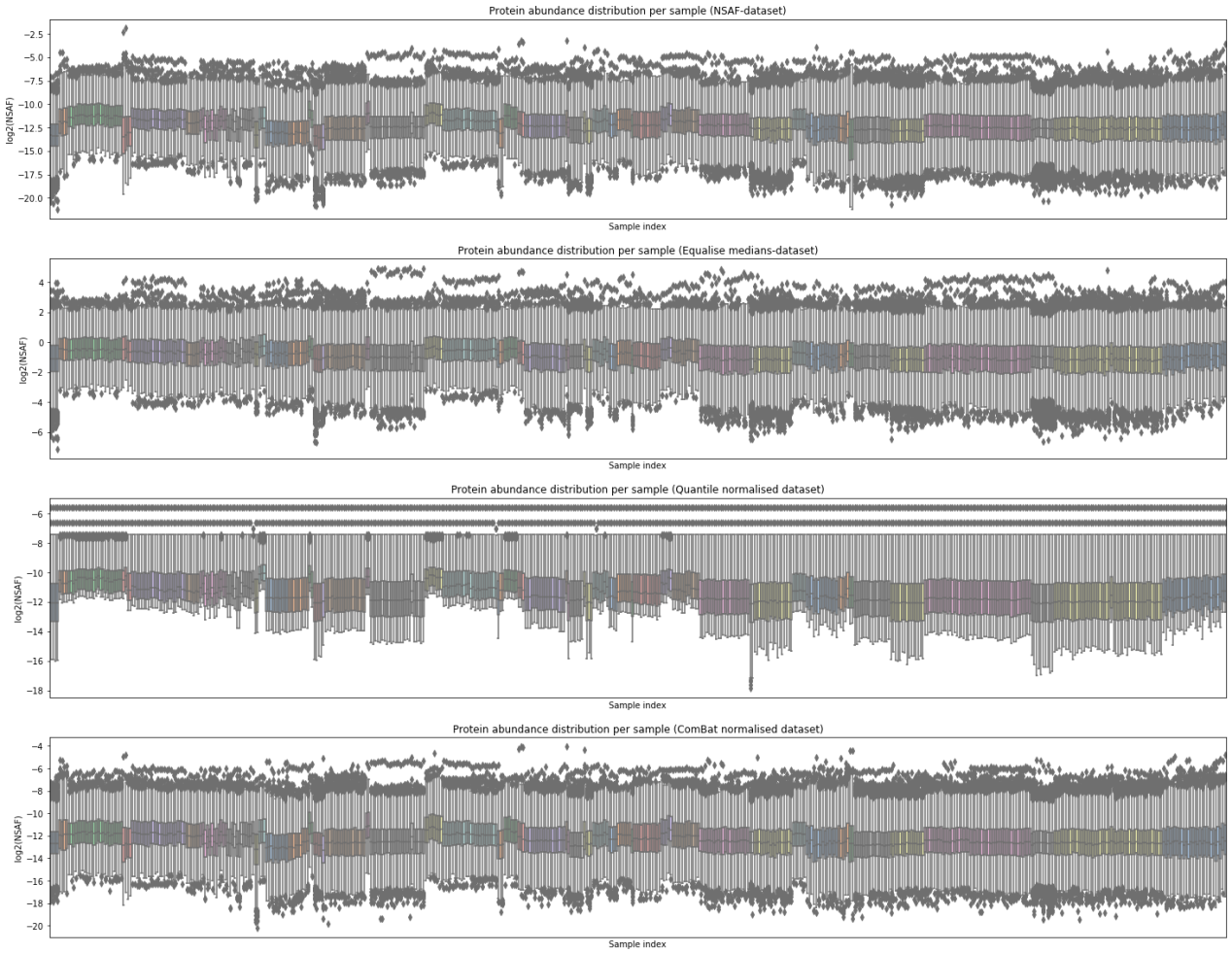


***Figure 12**: Boxplots showing protein abundance distributions per sample coloured by project for different normalisation methods. From top to bottom, the normalised datasets are NSAF-dataset, Median-normalised dataset, quantile-dataset, and the ComBat-dataset.*

7.4.1.1. Equalise medians

The underlying premise of the equalise medians method is that all commonly identified proteins lie roughly in the same range and most proteins are not systematically either all over- or underregulated. This means the median abundance and the standard deviation of the proteins in a sample should be the same over all samples. Since this premise was not met in the NSAF-dataset, equalise medians could reduce some bias in this manner.

To calculate the median and standard deviation for normalisation, only the proteins that are present in at least 90% of all samples are considered. This threshold was selected to minimise the possibility of a systematic under- or overregulation of the proteins that are used for normalisation, which is in violation with the assumption of equalise medians. Specifically, if only the proteins identified in all samples were considered, the normalisation would be performed on only 71 proteins. By including 90% reoccurring proteins, the set is expanded to 636 proteins, reducing the risk of violating the assumption

After the medians and standard deviation of the 90% reoccurring proteins were equalised over all samples, the distribution of the protein abundances over all samples were visualised (**Figure 12**). The boxplots indicate two aspects: (i) protein abundance distributions over all proteins are less

extreme than seen in the NSAF-dataset and (ii) the low quantification values of lower abundant proteins in deeper proteome analyses are retained. However, the NSAF-value of high abundant proteins are still negatively correlated with number of protein identifications. Additionally, no direct increase in correlation between projects of the same group were observed after equalise median normalisation (**Supplementary figure 4**).

7.4.1.2. Quantile normalisation

Quantile normalisation is an extreme form of normalisation which assumes that the distribution of abundance values is equal. Through a rank-based substitution of median or mean values, this assumption is enforced on the dataset. Although the mean and median substitution values were roughly the same, median was taken because the medium-to-high ranked median values increased more smoothly (**Supplementary figure 5**).

The distributions of protein abundance between samples are more heavily affected by quantile normalisation in comparison to the other normalisation methods (**Figure 12**). Of note is that the lower range increases with more protein identifications, which is directly caused by the rank-based substitution of abundance values. Indeed, this causes samples with less protein identifications unable to quantify any protein as less abundant compared to low abundant proteins quantified in high proteome coverage samples. The abundance range can be made more equal across all samples by imputing the missing values with low values as will be described in the next section. Although the correlation between number of protein identifications and the median abundance of most abundant proteins logically disappeared, this was not the case for the proteins identified in 90% of all samples (**Table 3**). Finally, Pearson correlation within a group for different projects did not increase after quantile normalisation (**Supplementary figure 4**).

7.4.1.3. ComBat normalisation

The goal of the ComBat normalisation, or batch effect removal method, is to equalise the expression value distribution of a protein across batches. In ComBat, the parameters to scale the protein values affected by batch effects are estimated on the dataset with an empirical Bayes approach by assuming batch effects affect proteins in a highly similar way within batches. In our case, batches are represented by PRIDE accession numbers and ComBat was applied on the NSAF-dataset imputed with mean-imputation.

After ComBat adjustment, the distribution of abundance values across samples was less uniform than for the other methods (**Figure 12**). However, the correlation of the 100 most abundant proteins and 90% commonly identified proteins in relation to the number of identifications decreased slightly (**Table 3**). Additionally, Pearson correlation between projects of the same group increased slightly for some groups (**Supplementary figure 4**).

7.4.1.4. Normalisation method comparison by dimensionality reduction

All datasets were imputed with the feature mean and projected in 2D-space with TSNE using perplexity of 20 (**Figure 13**). A similar figure to 13, but samples coloured by project is shown in **Supplementary figure 6**. Regardless of the described differences between the resulting datasets, no normalisation method appeared to produce noticeable improved clusters when observed through the dimensionality reduction techniques. ComBat seemed to influence the clustering the most. Based on our metrics, we gained insights into how normalisation is affecting our data, however we could not pick an optimally performing normalisation method. Therefore, a final decision of the choice of normalisation was postponed until after imputation.

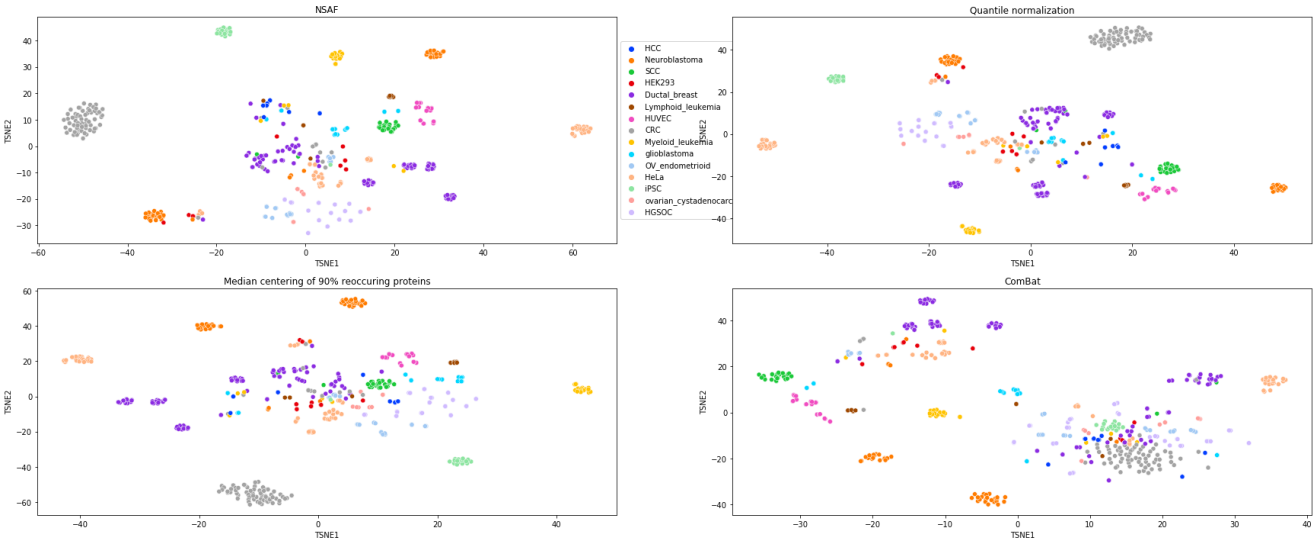


***Figure 13**: 2D-representation after t-SNE transformation with perplexity 20 for differently normalised datasets. Colours indicate cell line groups.*

7.4.2. Imputation

Each of the normalised datasets were prefiltered to include only proteins present in 50% of all samples, resulting in 325.829 missing values, accounting for 24% of the fPEMatrix. This substantial number of missing values highlights the importance of selecting an appropriate imputation strategy. The implemented imputation strategies can be categorised into three categories: (i) MNAR-imputation, (ii) MCAR-imputation, and (iii) a hybrid approach combining both. In this section, the imputation methods are compared based on their influence on protein abundance distributions and the correlation amongst each other when appropriate. Additionally, the effect of normalisation on imputation will be explored. Lastly, the most optimal combination of normalisation and imputation will be determined through a cross-validated evaluation of model performance.

For the MNAR-imputation method, missing values were imputed using a left-shifted Gaussian distribution (referred to as LOD-imputation). This approach assumes that the missing values represent protein abundances that are below the detection threshold of the MS instrument. **Supplementary figure 6** illustrates the imputation distribution in relation to the protein abundance distributions for nine proteins before and after imputation, specifically for the quantile normalised dataset.

To address the possibility that not all missing values are attributed to limit-of-detection, MCAR-imputation was also implemented. In the case of MCAR-imputation, the required number of PCA reconstruction iterations was determined by calculating the mean squared error (MSE) of the non-missing values after each iteration. When the PCA-components were kept constant after the first iteration (i.e., the number of components that explain 95% of the variance) of missing value reconstruction, the MSE declined until reaching 15 iterations (**Supplementary figure 7 left**). Similarly, when keeping the explained variance (95%) stable across iterations, the MSE increased strongly after reaching 15 iterations (**supplementary figure 7 right**). As a result, 15 iterations was considered the optimal hyperparameter for PCA-imputation. The second MCAR-imputer is based on the k-nearest neighbour algorithm using 10 nearest neighbours with a weighted distance metric. When both imputers were applied on each normalised dataset separately, correlations between imputed values defined by each method ranged from 0.63 to 0.81. The quantile-normalised dataset exhibited the lowest correlation, while the ComBat-dataset

showed the highest. These correlations indicate a substantial discrepancy between these two imputation methods that is also dependant on the type of normalisation used.

In the combinatorial method, the missing values were subdivided into MNAR-values and MCAR-values, accounting for 18.4% and 81.6% of the missing values, respectively, as described in the materials section. The impact of including LOD-imputation on the resulting protein abundance distributions before PCA-imputation is depicted in **Figure 14**. PCA-imputation tends to distribute imputed values evenly across the abundance range, emphasising the added value of combining PCA with LOD-imputation.

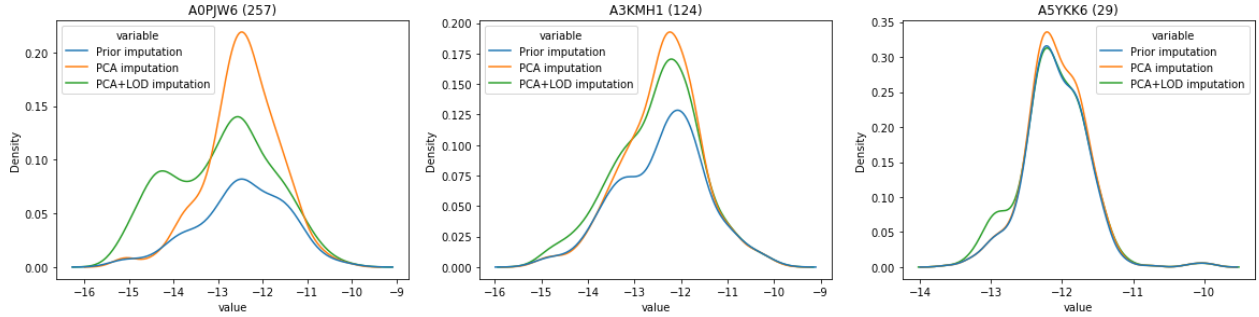


***Figure 14**: Distribution of abundance values for three proteins differing in number of missing values as depicted by the number next to the title of each subplot.*

The stability of the PCA-imputed missing values after LOD-imputation was assessed using Pearson correlation and was on average 0.92 after 20 repetitions. This suggest that the randomness introduced with LOD-imputation has a limited effect on PCA-imputation. However, not including LOD-imputation at all does appear to influence the missing value estimations made by PCA somewhat as evidenced by a lower average Pearson correlation (0.89).

The performance of each combination of normalisation and imputation method on several machine learning models was assessed as detailed in section 6.8.1. Through 10-fold stratified cross validation, ComBat underperformed significantly for all imputation methods except LOD-imputation (**Figure 15**). This suggests that ComBat normalisation might not be the best option for this dataset. In general, PCA, PCA+LOD and KNN+LOD were among the least effective imputation methods across all normalisation techniques and classifiers. On the other hand, LOD-imputation demonstrated the best performance overall, with minimal exceptions attributable by normalisation method or classifier.

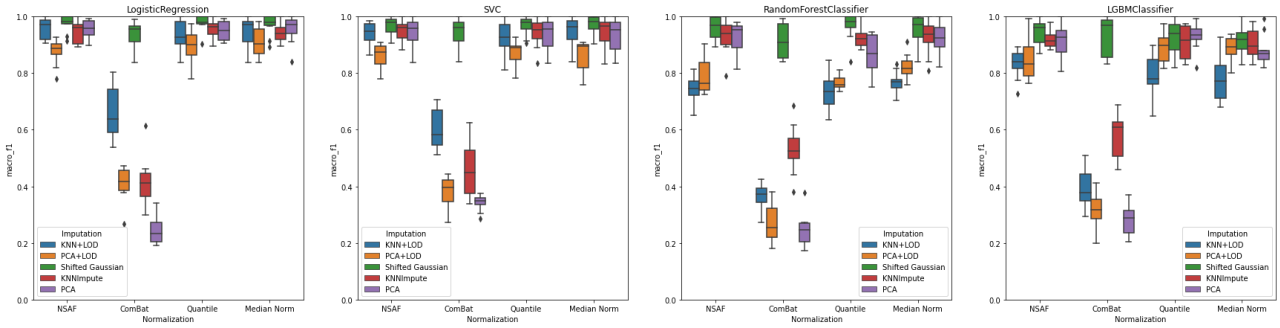


***Figure 15**: Macro F1-scores on the test sets over 10 folds of Stratified K-Fold cross validation for all possible normalisation-imputation combinations with four machine learning classifiers.*

To assess the generalisation performance over projects, leave-one-project-out cross validation was performed. The weighted-average F1-scores corroborated with the results from the stratified k-fold cross validation, indicating that LOD-imputation performs optimally (**Figure 16**).

Surprisingly, despite the observed bias caused by differences in proteome coverage within the NSAF-dataset as discussed in section 7.3.3, this bias did not adversely affect cross-project generalisation. Thus, the identified bias appeared to be less significant than initially anticipated.

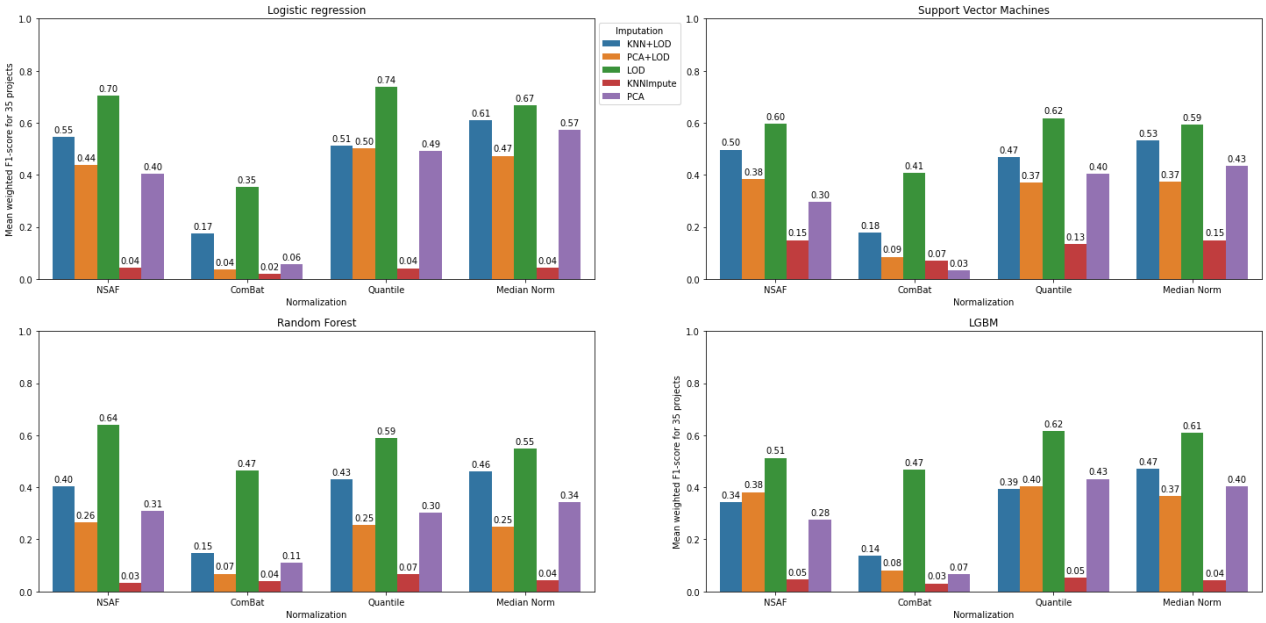


***Figure 16**: Barplots of the weighted-average F1-score over all 35 cross validation folds for every combination of normalisation method, imputer and classifier.*

Based on the maximum weighted-average F1-score across all methods, the combination of quantile normalisation and LOD-imputation was selected as the optimal choice for further analyses. Only in the next section, all methods will be utilised to evaluate the impact of selecting different pre-processing steps on feature selection. This analysis aims to demonstrate how the choice of specific pre-processing steps affects downstream analyses.

7.4.3. Feature selection

All feature selection methods were compared in terms of which features are deemed important for the classification algorithms. Additionally, the influence of normalisation and imputation methods on feature selection is analysed. Finally, a hierarchical clustering on the protein correlation matrix was performed on the reduced fPEMatrix after feature selection to validate whether functional associations can be picked up through correlation clustering.

For the ANOVA-based method, all features had a p-value below 0.01. However, a select number of features (129) demonstrated significantly larger F-values (**Supplementary figure 8 left**). This was determined by setting the cut-off at 1.96 times the standard deviation above the average F-value. Mutual information scores (MI) followed a similar trend, with 111 proteins identified as being more important (**Supplementary figure 8 right**). Interestingly, both methods identified few albeit different proteins being by far the most important. These were Keratin, type II cytoskeletal 8 (P05785) and 14-3-3 protein sigma (P31947) for ANOVA and Spectrin alpha chain, non-erythrocytic 1 (Q13813) for MI. Furthermore, both methods seemed to differ in assigning importance to the same proteins, showing a spearman correlation of only 0.56 (p-value <<< .05). Notably, there was an overlap of 44% among the top 100 most important proteins identified by both methods. Additionally, ANOVA tended to select slightly more proteins with a higher number of missing values (**Supplementary figure 9).**

To estimate the minimum number of features required for satisfactory classification performance, a recursive feature elimination (RFE) strategy was employed using three distinct machine learning models, further detailed in section 6.7.3. The random forest, linear support vector machines, and Logistic LASSO Regression models reached approximately peak performance on the test set using between 100-200 features (**Figure 17**). Performance decreased slightly after utilising more than 500 features.

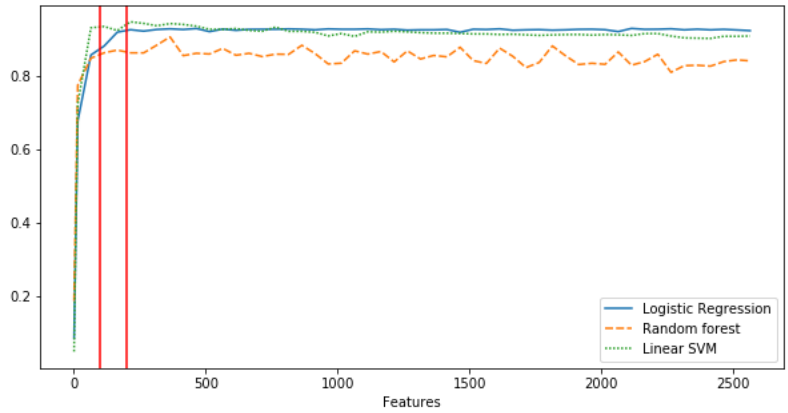


***Figure 17**: Average macro F1-score for the three models over 5 folds for each step during RFE. Red lines indicate 100-200 features.*

Only 12% of the top 100 features were chosen by all three models (**Supplementary figure 10 left**). The UpSet plot in **Figure 18** clearly demonstrates that each feature selection method selects proteins in distinct ways, with at least 26% of the selected features being unique to each method. Interestingly, only three proteins were consistently selected by all methods, including cell surface glycoprotein MUC18 (P43121), BTB/POZ domain-containing protein KCTD12 (Q96CX2), and keratin type II cytoskeletal 8 (P05785). If the number of selected features are increased to 300, 42 proteins are commonly selected and include important cancer-related proteins such as DNA-topoisomerase 2 (P11388), protein-glutamine gamma-glutamyl transferase 2 (P21980) and many proteins involved in cell adhesion. In agreement with the univariate feature selectors, feature selectors do not select the same proteins in terms of missing value percentage (**Supplementary figure 10 right**). This observation prompted us to investigate the influence of normalisation and imputation methods on the selection of the top 100 features.

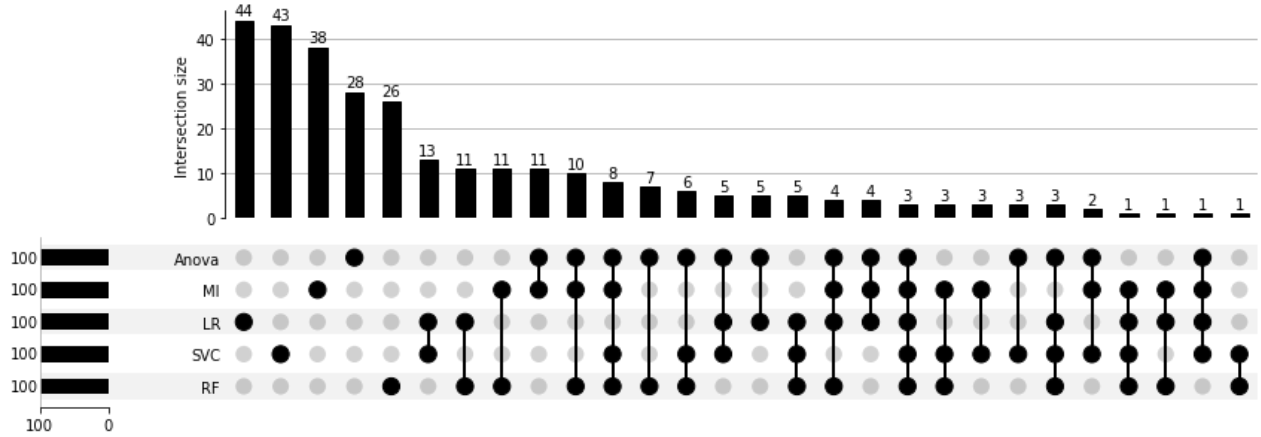


***Figure 18**: UpSet plot indicating the overlap of selected features among all used feature selection methods.*

To compare the impact of normalisation and imputation on feature selection, only features selected by at least three out of five selectors were considered. Among the different normalisation methods, the feature selectors exhibited the greatest disagreement when KNN-imputation was applied. On the other hand, the highest level of agreement among selectors was observed for the ComBat-PCA+LOD-imputation combination. This can be observed from **Supplementary figure 11**.

Imputation had an influence on feature selection as demonstrated in **Supplementary figure 12**. PCA+LOD and LOD-imputation resulted in the most similar selection of features compared to the other methods. Also the choice of normalisation method had a notable impact on the choice of features. Particularly when using ComBat, a distinct selection of features in comparison with other normalisation methods was consistently observed.

These findings highlight the substantial impact of upstream pre-processing steps, such as normalisation and imputation on feature selection. This impact is evident in both individual feature selectors and the level of agreement between feature selection techniques.

As it is clear each feature selector selects features differently, each subset of 100 most important features were evaluated in terms of their classification performance on four distinct machine learning classifiers. Overall, it was observed that the model-based feature selectors achieved the highest performance, albeit with marginal differences (**Figure 19**). Notably, the SVC and LR-based selectors in combination with LR yielded the best scores. To proceed with further analyses, a subset of features was chosen based on a consensus of the methods. Specifically, each method selected the top 300 features, and only those features identified by at least three out of the five methods were retained. This resulted in a final list of 161 proteins, which can be found in the 'selected_features.txt' file on GitHub in the 'preprocessing' folder.

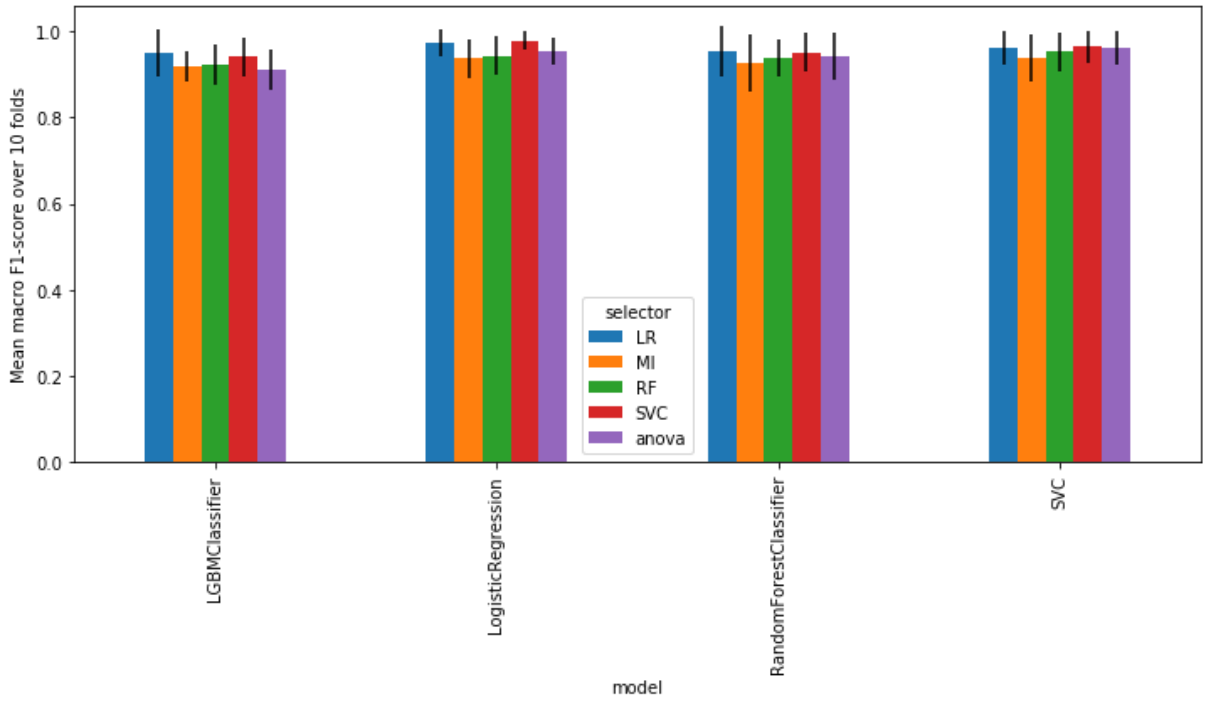


***Figure 19**: Stratified K-Fold CV-based model performance evaluation after selecting top 100 features with each selector.*

7.4.3.1. Identifying functional connections with correlation clustering

To evaluate the effectiveness of Pearson correlation in detecting known protein-protein interactions in the dataset, pairwise protein correlations were calculated, resulting in a total of 3,417,805 correlation coefficients. The majority of correlations fell within the range of -0.25 to 0.25 (**Figure 20 left**). Upon comparison of the correlations with the interaction scores from STRING, it became apparent that higher correlations were indicative of protein interactions. This was tested by iteratively taking a subset of the correlations above a defined threshold and determining the percentage of the correlations with an annotated protein-protein interaction from STRING. In STRING, the confidence of the interaction is defined by the STRING-score. The cut-offs for a qualitative measure of confidence are set at a STRING-score of 400, 700 and 900 for medium, high and the highest confidence respectively and are used in **Table 4** and **Figure 20 right** to illustrate the fraction of correlations above a certain threshold with annotated functional associations.

***Table 4**: In the table, the fraction of all pairwise protein correlations above a set correlation threshold that have STRING-annotated associations are shown in absolute numbers. The bracket numbers in the left column are the number of protein pairs above the set correlation threshold.*

	Medium confidence (STRING score 400)	High confidence (STRING score 700)	Highest confidence (STRING score 900)
Total annotated correlations	67105	26417	13618
Correlations > 0.8 (9)	3	4	6
Correlations > 0.65 (659)	157	208	303
Correlations > 0.5 (7184)	856	1284	1889

The subset of protein pairs with a Pearson correlation higher than 0.8 was seen to have the highest percentage (30%) of annotated protein-protein interactions with the highest confidence (STRING-score > 900). This percentage rapidly declined when including correlations below 0.6. Furthermore, when examining the functional connection for the highest correlated proteins that did not exhibit a protein-protein interaction as annotated by STRING, both proteins were seen to be functionally associated as illustrated in the STRING network in **Figure 20 middle**. Moreover, by lowering the STRING-score to 400, the percentage of annotated protein-protein interactions increased to 70% (**Figure 20 right**) These findings suggest that high Pearson correlations in our dataset can serve as a highly predictive measure for indicating functional associations between proteins in an unsupervised manner.

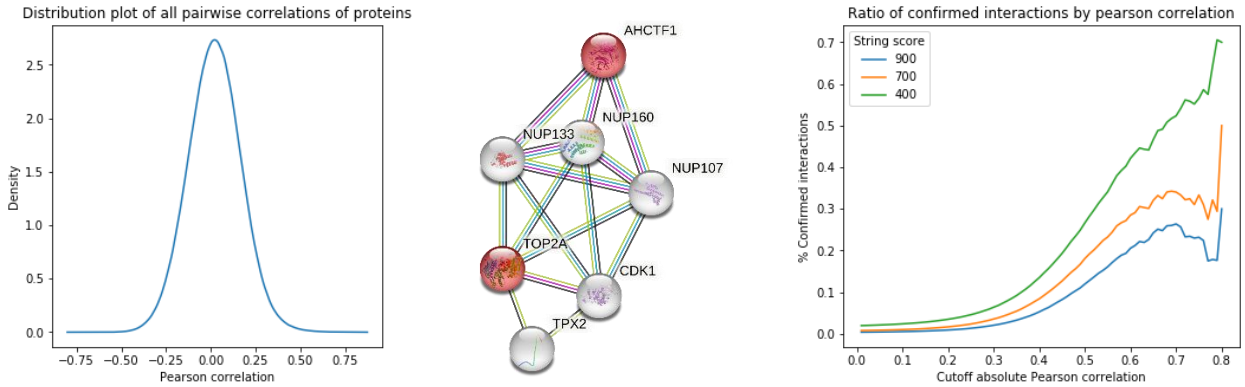


***Figure 20**: (left) Distribution of all pairwise Pearson correlations. (middle) String network for the pair of proteins showing the highest correlations with a STRING-score of 0. (right) Confirmed interaction ratio's for all proteins showing an absolute Pearson correlation higher than indicated on the x-axis.*

To further demonstrate the ability of our dataset to identify hub proteins, we conducted a search for proteins that consistently exhibited high correlations with a large number of other proteins. Our analysis revealed that Nucleolar RNA helicase 2 and Nucleolar protein 58 showed strong correlations with 25% or more proteins, with Pearson correlation values exceeding 0.7. Both of these proteins are associated with Gene Ontology (GO) term “small nucleolar RNA-binding”, which aligns with their known roles in gene expression regulation. These findings highlight that Pearson correlation analysis not only enables the identification of protein-protein interactions but also allows the detection of proteins that have a significant impact on the expression of numerous other proteins. These hub proteins likely play a crucial role in defining the cellular phenotype.

In the light of these observations, a correlation-based hierarchical clustering approach was taken to identify functional clusters of co-regulated proteins based on their abundance patterns across samples. These clusters provide insights into the potential functional associations among groups of proteins, and additionally could allow to reduce the feature space. The cluster heatmap, showing all pairwise protein correlations in the dataset, revealed several large, distinct and visually discernible clusters (**Supplementary figure 13**). To limit our focus towards the proteins deemed predictive by the feature selectors, only the clusters found in the feature-selected dataset, containing 161 proteins, were explored.

By using hierarchical correlation clustering with Ward linkage, 15 clusters could be discerned by cutting the dendrogram at threshold 3 (**Figure 21 left**). Each of these clusters were subjected to biological evaluation using protein interaction and gene ontology enrichment analysis. Notably, the most distinct clusters, cluster 1 and 2, exhibited subclusters when subjecting these proteins to protein-protein enrichment analysis with the STRING database (**Figure 21 right**).

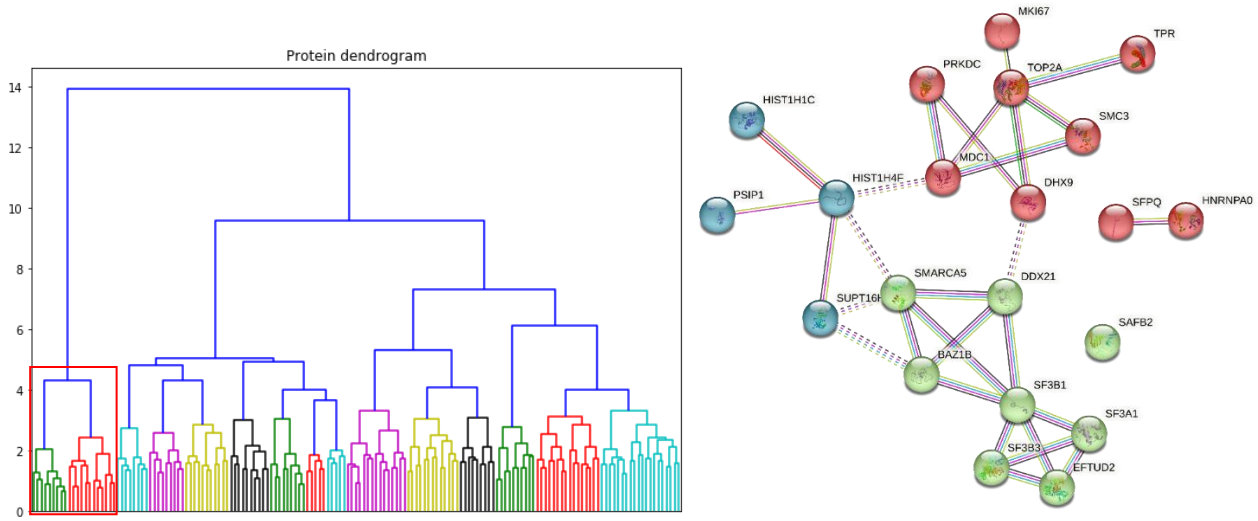


Figure 21*: (left) Protein dendrogram showing clusters of proteins after performing hierarchical correlation clustering analysis with Ward linkage. (right) The two clusters, indicated with a red box in the left figure 21, were subjected to a STRING query and clustered with KMeans (3 clusters). Only interactions with high confidence (STRING-score > 700) are shown.*

The three identified subclusters (**Figure 21 right**) were enriched for nucleosome organisation (blue), spliceosomal proteins (green) and proteins regulating RNA export from nucleus, alternative mRNA splicing and DNA integrity (red). A similar protein-protein interaction enrichment was performed for the other clusters and is summarised in **Supplementary figure 14**. Notably, cluster 6 was enriched for cell-cell junction molecules associated with desmosomes and included the important epithelium-specific marker, 14-3-3 sigma protein (P31947). Furthermore, cluster 13

included two 26S proteasome subunits and two alpha-actinin subunits. Other clusters grouped proteins whereof a small number interacted with high confidence.

In summary, these findings suggest that correlation cluster analysis of the feature selected dataset does pick up proteins that are biologically related. However, also proteins with no previously reported associations were clustered together. Due to this ambiguity and the uncertainty associated with choosing one protein representative of a cluster, further feature reduction was not pursued.

7.5. Fighting imbalance with weights and oversampling

7.5.1. VAE hyperparameter selection

The hyperparameters for two separate Variational Autoencoders (VAEs), VAE1 and VAE2, were tuned as explained in the materials section. VAE1 was trained on the quantile dataset before feature selection, while VAE2 was trained on the dataset containing 161 proteins. The network architectures for both VAEs can be found in the materials section. In this section, we present an evaluation of the hyperparameter tuning process and provide pairwise plots showcasing the learned latent distributions for the dataset.

For VAE1, it was observed that larger latent dimensions (50 latent variables) were less effective. Optimal results were achieved with latent dimensions between 5 and 10 and a batch size of 5, based on the running average loss over 100 epochs on the validation data (**Supplementary figure 15**). The tested learning rates did not seem to impact the performance, so the standard learning rate of 0.001 was deemed appropriate. Since the model did not converge within 100 epochs, it was refitted with the chosen hyperparameters and trained for 1000 epochs. Although the hyperparameter search was not performed for 1000 epochs due to computational limitations, 1000 epochs seemed to provide satisfactory results, as the loss only slightly decreased beyond this point (**Figure 22 left**). To visualise the learned latent distributions per group, each sample in the dataset was resampled 20 times by the VAE, and pairwise plots were generated for the latent variables of each sample. This analysis supported the notion that 10 latent dimensions were enough, as only 8 out of 10 latent variables captured biologically meaningful encodings (**Supplementary figure 16**).

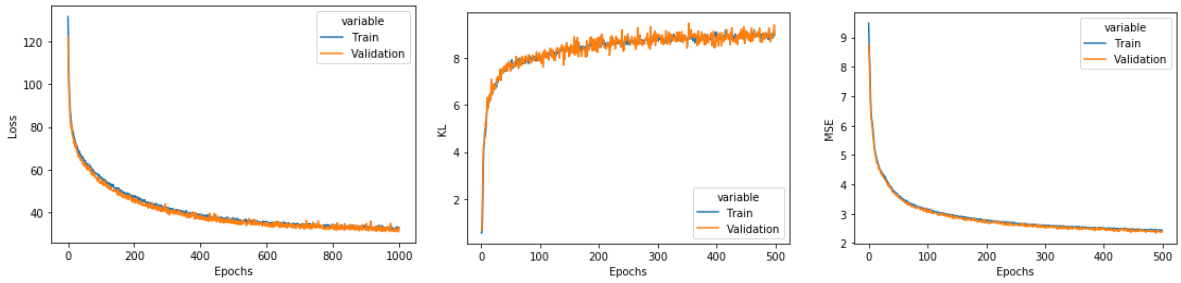


***Figure 22**: (left) Line plot showing validation and test loss over all training epochs for VAE1. (middle and right) Line plots showing the (middle) KL and (right) MSE for validation and test set for VAE2.*

During the hyperparameter tuning process for VAE2, the most important parameters to optimize were kappa and the size of the latent dimensions. Similar to VAE1, a parameter sweep was conducted, and it was found that 6 latent dimensions, a batch size of 5, and a kappa value of 0.001 were optimal for VAE2 (**Supplementary figure 17**). However, the latent space appeared to be highly uncontrolled, so the kappa value was increased to 0.1 to introduce more generalisation, which resulted in improved distributions of the latent variables (**Supplementary figure 18**).

Furthermore, PCA plots of resampled samples indicated that the reconstruction was somewhat noisy but still accurate enough to be utilised as a data augmentation tool (**Figure 23**).

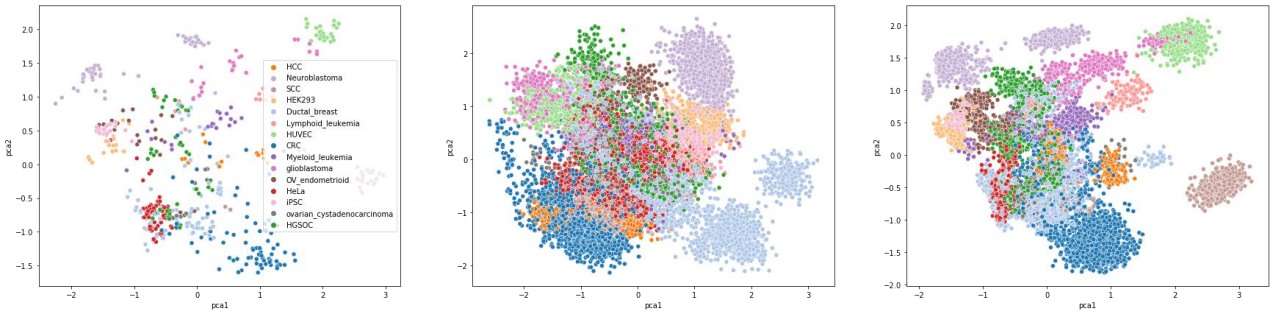


***Figure 23**: 2D-representation of the first two principal components after PCA on (left) the feature selected dataset (middle) the encodings containing six variables and (right) reconstruction. For the encodings and reconstruction, the dataset was resampled 20 times.*

Of note is that certain encodings seemed to represent class specific concepts, which clustered samples from the same class but different labs close together (**Supplementary figure 19**). However, further interpretation of these encodings was not pursued in this study. A description of previously successful biological interpretations of latent variables is provided in the discussion.

7.5.2. Evaluation of balancing methods

Four oversampling methods and one weighting-based method were assessed to address class imbalance. Several metrics were used to evaluate the soundness of the generated samples. These metrics include statistical tests on distribution, variance, and the mean of the features. Furthermore, the ability of the oversamplers to produce similar yet distinct samples that preserve or enhance the original group clustering in lower-dimensional space using PCA was examined. Finally, the performance of each method was evaluated using k-fold cross validation with four machine learning models, using the weighting-based balancing method as a baseline.

To measure the quality of the generated samples, new datasets for each oversampler were generated with a class distribution that is equal to the original PEMatrix. A summary of the number of features being statistically different from the PEMatrix for each oversampler after Bonferroni correction are shown in **Table 5**.

***Table 5**: Number of features that are significantly different from the original dataset in terms of variance, distribution and mean after Bonferroni correction.*

	SMOTE	SMOTE-Tomek	SMOTE-ENN	VAE
Levene test	126	57	83	68
Kolmogorov-Smirnoff test	0	0	0	154
t-test	0	0	0	8

In the VAE-dataset, 2.6% and 5.9% of features showed significant differences in terms of variation and distribution compared to the original dataset. Interestingly, these features consistently exhibited less variation in the VAE-resampled dataset (**Supplementary figure 20**). For the other methods, a similar number of features differ in terms of variation. However, no features differed significantly in terms of both distribution and mean, suggesting that the VAE was

less effective in generating a statistically similar dataset through resampling compared to the other oversamplers. To provide a visual illustration, the distributions of features with the most significant distribution differences for the VAE are presented in **Figure 24**, alongside the distributions of the generated datasets from the other oversamplers. It can be observed that the VAE tends to overestimate the lower values in the distributions.

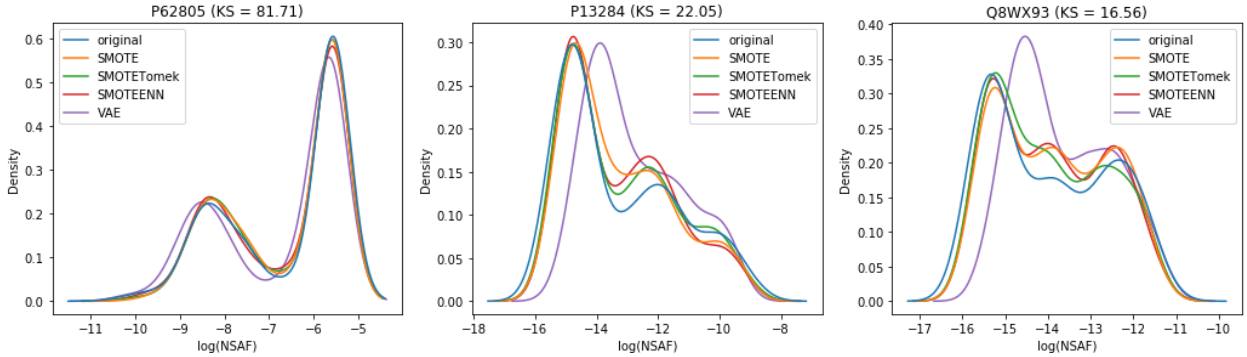


***Figure 24**: Kde-plots of the five proteins differing the most significantly between VAE and the original dataset.*

To assess the improvement in group-clustering, each oversampling method was applied separately to the minority classes until reaching the sample size of the majority class (100 samples). PCA was then conducted on the original and balanced datasets (**Figure 25**). It can be observed that the SMOTE-based methods generated data points that were not close to any other samples, which could be attributed to interpolating between samples with high-intra class variability. In contrast, the VAE produced broader clusters, indicating the generation of class-specific replicates with some level of noise added. Moreover, the VAE was not affected by high-intra class variability as it employed sample-specific data augmentation rather than interpolation between class samples.

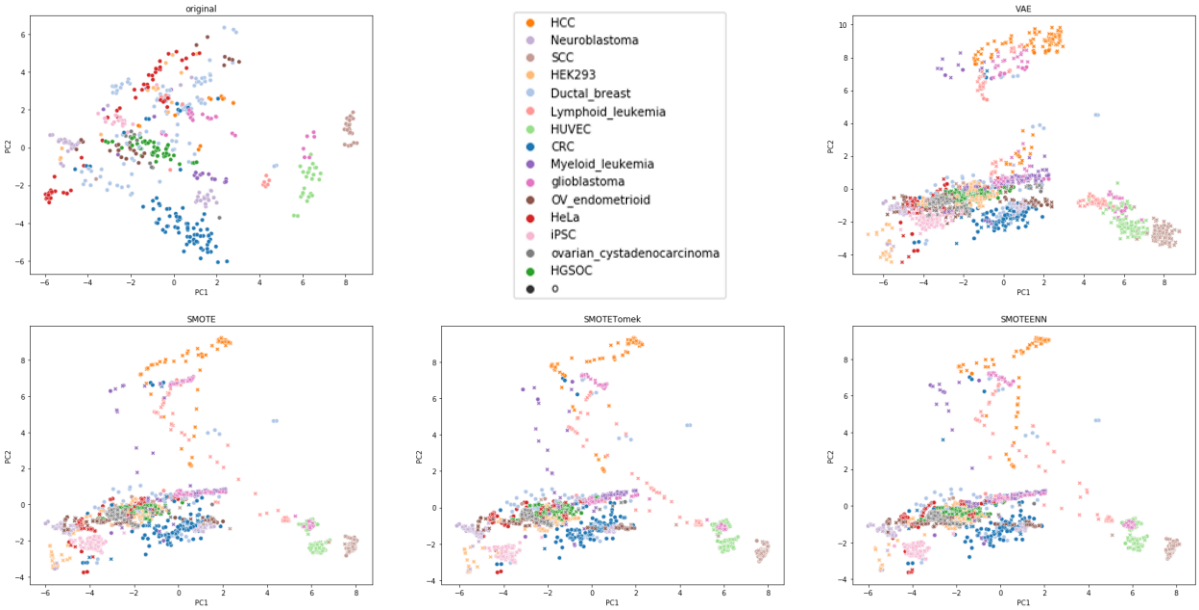


***Figure 25**: PCA-plots of the original and the balanced datasets after oversampling so each class has 100 samples. Crosses indicate synthetic datapoints while circles indicate the original samples*

The effectiveness of using oversamplers compared to weighting minority samples was assessed using four classifiers in a 10-fold cross-validation setup. In each fold, the training set was processed by imputing missing values, scaling features, and applying oversampling techniques based on the training set only. Feature selection was then performed by retaining the top 161 features identified previously. The results, depicted in **Figure 26**, indicate that, overall, the cost-sensitive method yielded equal or superior performance over all oversamplers for each classifier, except for the VAE, which exhibited significantly poorer performance.

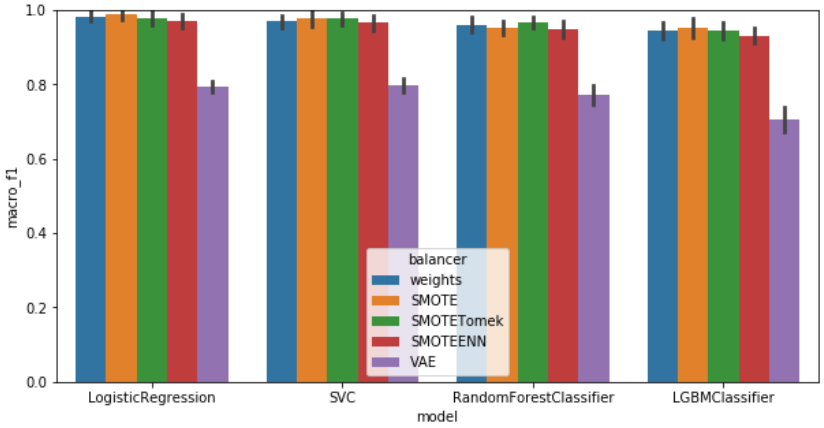


***Figure 26**: Bar plots showing macro F1-score after 10-fold stratified cross validation for each balancer and classifier.*

The effect of using feature selection before or after oversampling was evaluated by measuring the overlap of the features selected by the feature selection pipeline described in section 7.4.3 before or after oversampling (**Figure 27**). An overlap of 41.2-46.4% for all methods was observed. Nonetheless, the original dataset selected the most unique features, suggesting a significant effect when changing the order of these pre-processing steps.

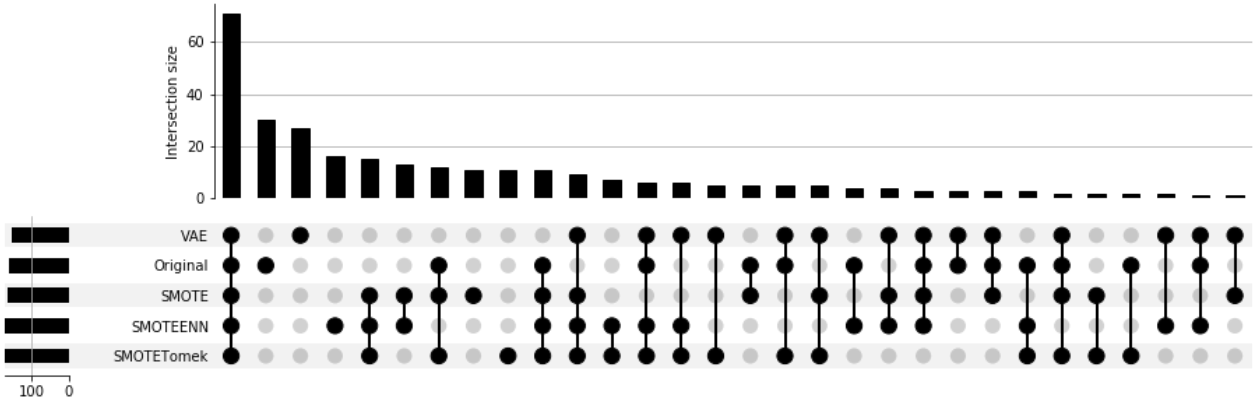


***Figure 27**: UpSet plot showing the overlap of features selected after using each oversampler. The 'Original' label represents the features that were selected as described in section 7.4.3.*

Therefore, a second evaluation was conducted to determine the best balancing method. This time, oversampling is performed after feature selection. The results were consistent with the evaluation of oversampling before feature selection. Additionally, VAE also did not perform as well as the other methods. As a result, SMOTE was chosen as optimal balancer after feature selection, although minor differences were observed between weighting, SMOTE, SMOTE-Tomek and SMOTE-ENN.

7.6. Hyperparameters and model selection

To optimise the model hyperparameters, the dataset was used with the following pre-processing steps in order: (i) quantile normalisation, (ii) reduction of features to the selected 161 features (iii) LOD-imputation, (iv) minmax-scaling of the features, and (v) oversampling with SMOTE. Steps 3-5 were fitted on the training set only to prevent data leakage.

A randomised grid search with 50 iterations per fold (5 in total) on the training set (90% of total data) was performed for the most important hyperparameters of Logistic Regression, Random Forest and Support Vector Machines. In **Table 6**, the selected hyperparameters per model are summarised. For Logistic Regression, optimal regularisation factor (C) over 5 folds on the test set ranged between 94 and 3.6. The mean was approximately 50 and was taken as the optimal parameter. The mean F1-score across the folds was higher for Ridge regularisation and was thus taken as loss function with the liblinear solver. For Random Forest, the criterion was entropy uniformly over all folds and the lowest maximum depth (10) was chosen with 150 estimators. For Support Vector Machines, the radial basis function kernel was optimal with a mean regularisation factor of 15.

***Table 6**: Summarisation of the selected hyperparameters with randomised grid search*

	Hyperparameters
Logistic Regression	C = 50; Penalty: L2; solver = “liblinear”
Support Vector Machines	C = 15; kernel = “rbf”
Random Forest	N_estimators = 150, criterion = “entropy”, max_depth = 10

To evaluate the generalisation capability of the models, cross validation by project was performed for 35 projects. In contrast to **Figure 16**, performance seemed to generalise well over projects (**Figure 28**). The best mean score was obtained for the Logistic Regression-SMOTE combination and was 93.7%. Interestingly, only the Logistic Regression model generalised better by using SMOTE as oversampler (p-value = 0.024). For the Support Vector Machines and Random Forest models, VAE had a slightly higher mean value than SMOTE (SVC: 84.6% – 83.8%,RF: 85.2% – 81.7%) although not significant. These results suggest that the models with the selected hyperparameters and pre-processing steps are able to generalise over projects, at least for the 35 projects that were tested. The other projects were not tested due to not meeting the criteria described in 6.8.1. Furthermore, the performance when using VAE-resampling with stratified k-fold cross validation was the only one (apart from the LR-SMOTE combination) to reflect the generalisability score when using the project split methodology, suggesting more robustness against overfitting than SMOTE or weighting. Because the LR-SMOTE combination performed best, this combination is used to biologically interpret the model in relation to the cell line groups.

During cross validation by project with LR-SMOTE, only 5 projects were not predicted with 100% accuracy. An overview of the true and wrong predictions are supplemented in the addendum as **Supplementary table 1** (after supplementary figure 22).

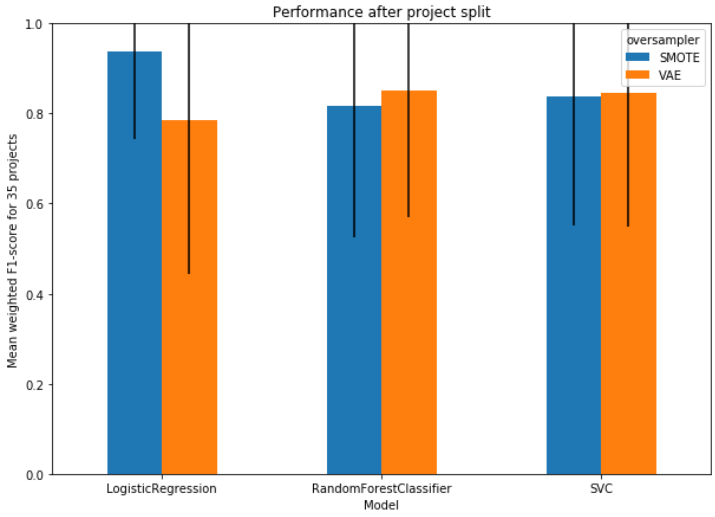


***Figure 28**: Weighted-average F1-score after leave-one-project-out cross validation on 35 projects with the optimised workflow and tuned hyperparameters for three machine learning models.*

7.7. Biological interpretation

To identify the key proteins responsible for distinguishing cell line groups, the dataset was divided into a set used to fit the model (80% of samples) and a test set used to measure feature importance. The tuned Logistic Ridge Regression model was employed and a SHAP-explainer object was fitted. This SHAP-explainer is capable of measuring the individual impact of a protein in the prediction. SHAP-values were then computed exclusively on the test set and averaged within each group for accurately predicted samples. By summing the averages of absolute SHAP-values per group, it was determined that microtubule-associated protein 1B (P46821) and Filamin-C (Q14315), which play crucial roles in cytoskeleton reorganization, were the primary proteins contributing to the discrimination of cell line groups.

Average feature importance per group showed that only a few proteins show a strong impact on the prediction of group identity (**Figure 29**). This means only few proteins are needed to define a group of cell lines. Features with an average SHAP-value above 0.4 were considered group specific and are listed with their corresponding relative abundance in 'most_important_features.csv' on GitHub. Group-specific proteins were compared with annotations from the Human Protein Atlas (THPA) concerning (i) the enrichment of a protein in a particular cancer type, the prognostically favourable or unfavourable value of a protein and (iii) the annotation of the RNA-based gene expression cluster for the protein. From the total of 101 unique proteins, 28 were cancer-related according to THPA. STRING enrichment analysis of this subset demonstrated a significant enrichment for cell-substrate junction assembly (FDR = $8.62*10^{-5}$) and mitotic cell cycle checkpoint (FDR = 0.03) (**Supplementary figure 22** for STRING interaction network). Additionally, 24 proteins were prognostic to some kind of cancer according to the Human Protein Atlas (THPA). Interestingly, only 2 prognostics were assigned to the correct cell line group in terms of type of cancer it is believed to model. This includes the favourable prognostic value of isocitrate dehydrogenase (P48735) for cervical cancer which in HeLa cells was the second most important feature for classifying HeLa cells when in low abundance. These observations indicate a disparity between the prognostic indications defined by THPA and the cell line group specific proteins defined by our model.

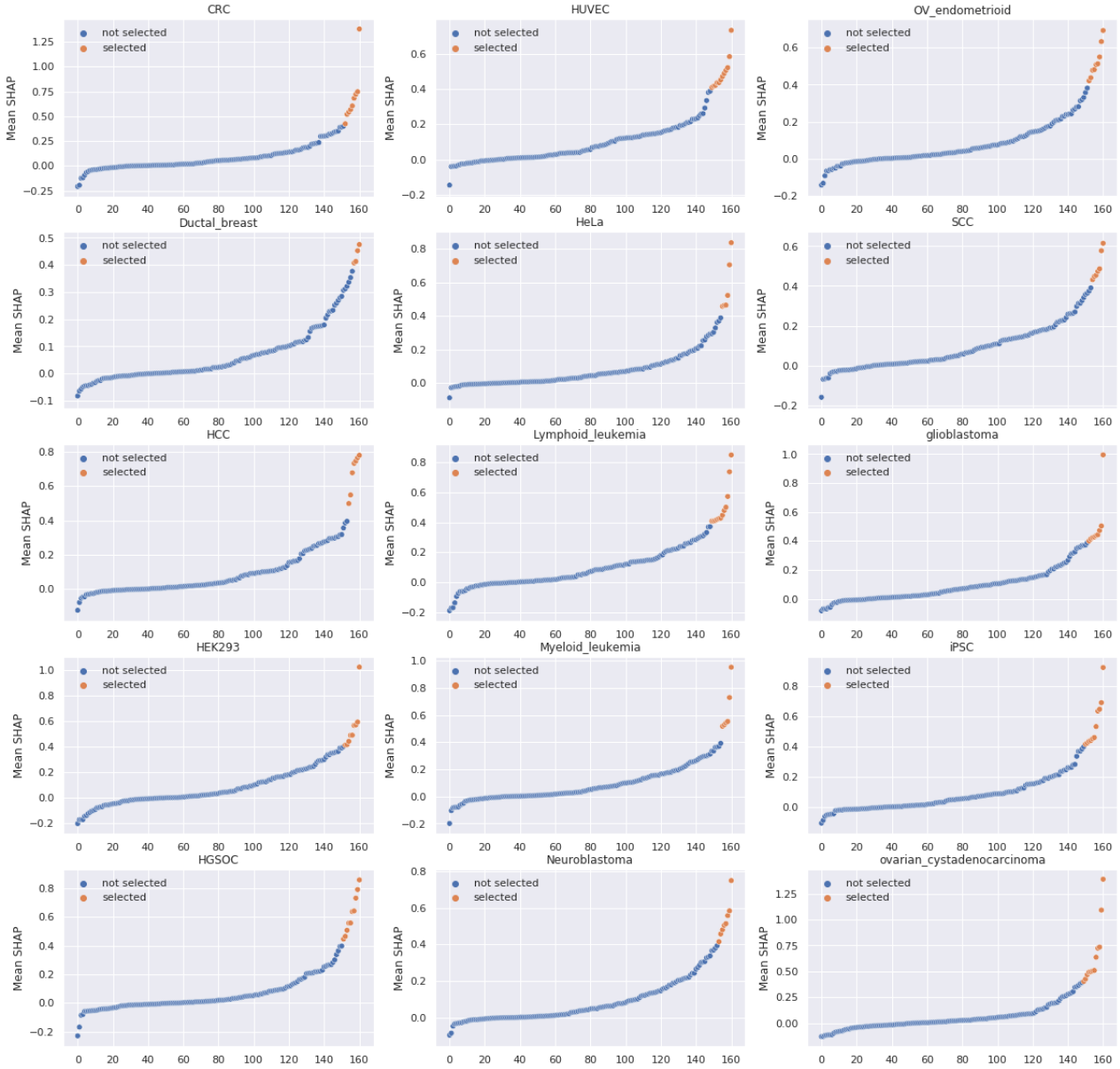


***Figure 29**: Rankplots of feature importance defined by the average SHAP-value of correctly predicted test samples for each class.*

Upon comparison of the cell line specific feature importance with the RNA-expression data provided by THPA, a certain agreement can be observed. For example, Coronin-1A (P31146), which is assigned to the lymphoid cancer expression cluster in THPA, emerges as the most important feature for classifying lymphoid leukemic cancer cell lines. Similarly, Tyrosine-protein kinase Lyn (P07948), a key protein for the myeloid cell line group, is part of the myeloid cell line expression cluster. The significance of G protein-regulated inducer of neurite outgrowth 1 (Q7Z2K8) as the most important protein for neuroblastoma cell lines was supported by THPA annotations. However, there are also a few discrepancies. Integrin alpha 3 (P26006), despite being the most important feature for glioblastoma cell lines, does not correspond to the RNA cell line cluster in THPA and instead is assigned to the ovarian cancer cluster. Nevertheless, the literature has indicated that an overexpression of integrin alpha 3 in glioma cells is associated with increased invasive and migrative capabilities of these cells[75]. Moreover, integrin alpha 3 was

proposed as a therapeutic target for glioblastoma[75]. The most intriguing discrepancy found so far involves shootin-1 (A0MZ66), a protein crucial for neurite outgrowth and neuronal polarisation. Although THPA indicates its enrichment in brain tissue, particularly in oligodendrocytes, the absence of shootin-1 ranks it as the second and 16th most predictive feature for neuroblastoma and glioblastoma respectively. This specific example shows that it is not only important to look at the importance of a protein in prediction as such, but also the relative abundance must be taken into account. Indeed, the reason why shootin-1 is an important feature for neural-like cell lines is the reverse of what was expected.

To compare the protein profile of a cell line with their corresponding tissue of origin, the top 20 features specific to each group were compared to the feature importance and tissue specificity of the in-house developed tissue classifier[28]. **Supplementary figure 23** shows a global overview of the tissue specificity of the cell line group discriminating proteins according to the tissue classifier.

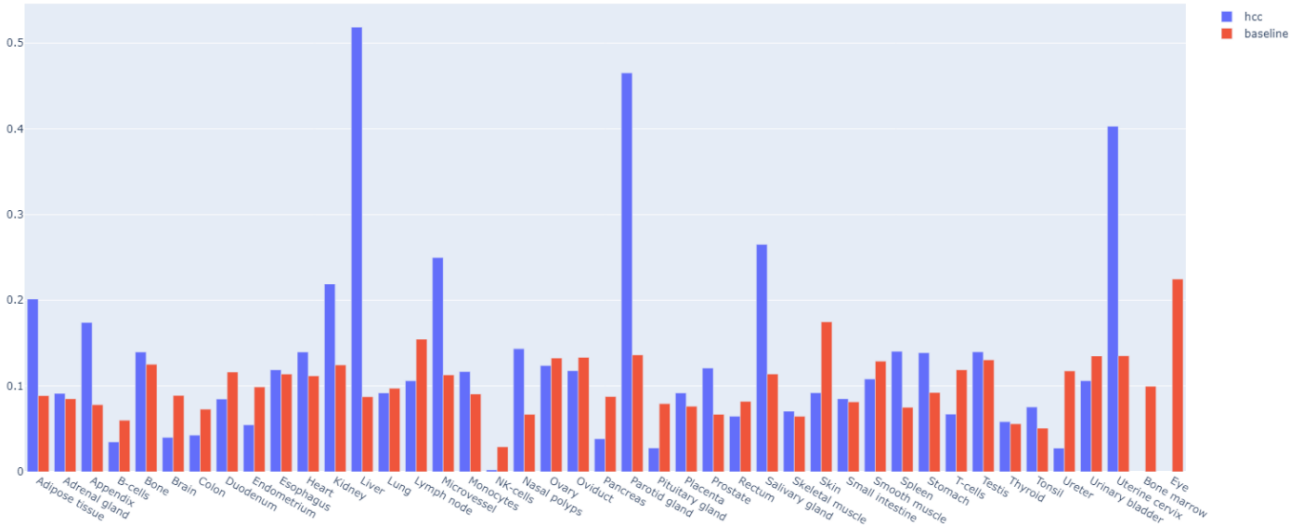


Figure 30: *Bar plots showing the tissue specificity of the HCC-specific proteins. Liver, parotid gland and uterine cervix show the largest enrichment.*

Notably, the hepatocellular carcinoma (HCC) group displayed distinct protein profiles associated with liver-specific functions, indicating that HCC cell lines retain characteristics of liver tissue (**Figure 30)**. The liver-specific proteins, such as aldehyde dehydrogenase (P05091), pyruvate carboxylase (P11498), and UDP-glucose 6-dehydrogenase (O60701), are involved in metabolic processes, exhibited at high expression levels in the HCC group. However, this kind of similarity cannot be determined for every cell line and their acclaimed tissue of origin.

For iPSCs, the most important discriminative feature was high expression of podocalyxin (O00592). According to the tissue classifier, podocalyxin is associated with microvessels and the kidney, with greater importance assigned to the former, which is in part corroborated by the information provided by THPA, which states that podocalyxin is a protein specific to podocytes in the kidney. However, the literature have proposed podocalyxin as a stem-cell marker and has pointed towards the important role of podocalyxin during development of adult stem cells, which could explain the high importance associated with this protein for classifying iPSC[76].

In the light of these findings, the role of podocalyxin in classifying HEK293 was explored. HEK293 is a human embryonic kidney cell line, which according to the involvement of podocalyxin in the developing kidney should have a high expression of podocalyxin. Although we observed podocalyxin to be the 16th most predictive feature for HEK293, HEK293 had the lowest relative expression of podocalyxin over all groups.

Interestingly, nestin (P48681) emerged as an important feature for classifying breast cancer cell lines, neuroblastoma, and iPSCs. However, according to the tissue predictor, nestin is primarily associated with heart tissue, which is in corroboration with the tissue profile annotation from THPA. However, nestin is also a neural stem cell marker which plays a role in proliferation, differentiation and migration in the context of the cytoskeleton[77]. Consistent with its neural stem cell function, nestin exhibited the highest average expression in iPSC and neuroblastoma, while its expression was among the lowest in breast cancer cell lines. Furthermore, when examining the correlations within the complete unfiltered dataset, nestin showed a positive correlation with MAP1B (P46821) and vimentin (P08670), both of which are involved in cytoskeletal interactions and migration. In contrast, nestin exhibited negative correlations with plakophilin-3 (Q9Y446) and keratin 8 (P05787), which are more specific to epithelial tissues. In line with this finding, it is not surprising that low abundance of nestin ranks among the top 20 most important features for classifying squamous cell carcinoma of the skin, a cancer type with a highly epithelial character.

This preliminary biological interpretation of the output of the cell line group classifier highlights the biological validity of its predictions. It demonstrates the ability to detect proteins, representing cell line specific characteristics. The integration with the previously trained tissue predictor model reveals the association (or lack thereof) between cell lines and their tissue of origin. Delving deeper into the patterns uncovered by the model could provide valuable insights into the unique aspects of cell line models.

8. Discussion

In this research, the goal was to find characteristic protein patterns of cell lines by leveraging public label-free proteomics data. In order to achieve this goal, 43 previously published experiments containing 518 full proteome samples were used from the PRIDE-database. After careful selection of the optimal pre-processing method, a Logistic Ridge Regression model was trained to classify the dataset in 15 cell line groups and achieved on average 93.7% classification performance. Indeed, we did not achieve our goal to identify cell line-specific proteins as the lack of data forced us to aggregate multiple cell lines in groups. Nonetheless, by interpreting the model, i.e. analysing which proteins are important for classifying cell lines in their respective groups, this work does provide a resource worth exploring further. Furthermore, the workflow employed in this study can serve as both a source of inspiration and a cautionary tale for future repurposing experiments. In the upcoming sections, the findings will be contextualized within the scientific landscape and a critical reflection will be presented on the work undertaken while highlighting areas that warrant improvement.

As mentioned, instead of finding cell line-specific protein patterns, cell line group protein patterns were identified. This was mainly caused by the lack of data. Indeed, much less projects than initially expected fulfilled the selection criteria that were set out in this study. It seemed more projects, including comprehensive cell line datasets, analysed samples with labelled approaches such as SILAC, TMT and iTRAQ, which we unfortunately could not include because protein quantifications obtained with labelled approaches are generally not comparable across multiple runs, whereas label-free methods are[78]. Comparing labelled experiments across runs requires the same reference sample in each run to normalise upon. Because here, analyses with diverse experimental designs are aggregated in one dataset, this requirement is not met. However, recent developments have indicated that by using an MS1-based quantification method such as iBAQ, individual channels from isobarically labelled samples can be transformed to individual samples like those obtained from label-free experiments by splitting the MS1 peak area proportional to the reporter ion intensities obtained in the MS2 spectrum[79]. Although this could enlarge the aggregated dataset when label-free and labelled methods can be directly compared, these two forms of protein quantification still have their own intrinsic forms of bias further complicating the data analysis.

While the acquisition of more data is generally desirable, it introduces a new challenge that must be addressed: metadata annotation. Apart from a few projects, obtaining metadata proved to be a challenging task, consuming considerable time. Particularly for a novice researcher in the field, this undertaking at the start of the research was inevitably prone to errors. As more data is collected, the burden of metadata annotation falls on the bioinformatician, limiting the time available for what matters most: the actual analysis. Luckily, initiatives have been undertaken to combat this overlooked yet significant limitation in current proteomic research[80]. Nevertheless, the reader should be aware of the potential presence of wrong annotations in this research which may have influenced the results, despite the efforts taken to minimise this possibility.

Of the collected data, we observed a systematic lower number of protein identifications in comparison with the reported number in the publications. This can in part be caused by the use of different search engines with differing search setting. However, the most plausible explanation for this is the following: We only included peptides that are proteotypic, thus peptides that uniquely map to a canonical protein sequence. While we also considered isoforms to be a distinct protein, this reduced our set of useable peptides, and in extrapolation the number of identified proteins, tremendously. Although grouping proteins can limit the number of discarded peptides, this reduces the sensitivity to detect cell line specific differences which may present itself as a difference in expression of a specific isoform. Furthermore, as most of the cell lines used in this

research are cancerous, single amino acid variants (SAAV) are expected to be present due to the high mutational burden of these cells. These can be included in the search space of the search engine but this will increase the search space tremendously. This could in turn negatively impact the number of selected peptides in the following two ways: Firstly, due to the enlarged search space, the false discovery rate must be reduced to prevent the identification of more false positives which could ultimately lower the number of peptides that are deemed useable. Secondly, during the proteotypic filtering procedure, less peptides will be kept if also a distinction is made between highly similar isoforms with only few differing amino acids. Therefore, it could be desirable to not be too stringent on the definition of a protein group to limit the number of peptides that get filtered, yet are confidently identified.

Three sources of technical variation among the collected datasets were explored and include (i) sample preparation workflows, (ii) ionbot versions and (iii) interlaboratory bias.

The first source of variation arises from a discrepancy in the type of proteins that are extracted when performing different sample preparation workflows. These analyses showed that in-gel digests extracted a more hydrophobic part of the proteome than in-solution digests. One explanation for this is that plasma membrane proteins, which are very hydrophobic, are less represented in in-solution and on-filter methods. This is believed to be caused by incomplete solubilisation of hydrophobic proteins, which makes them unable to be proteolyzed effectively[81]. Previous results using the same measure of hydrophobicity comparing the recovery of membrane proteins between sample digestion methods corroborate with our findings[81]. However, the reverse could also be true: in-gel digestion is less able to digest hydrophilic proteins than in-solution digestion. This can be caused by the insufficient binding of SDS, which results in a lack of denaturation and thus inefficient digestion of these proteins[82]. Although both are plausible, we note that our method to measure hydrophobicity is not optimal. Advancements in protein folding prediction can be utilised to enhance the precise determination of a protein's hydrophobicity, as well as its propensity to aggregate or resist denaturation during various sample preparation workflows. This will provide better insights into why and which proteins tend to be identified better in a certain workflow.

A second form of variation was caused by the use of different ionbot versions. Although unclear what caused this, there are two possibilities: (i) either the ionbot versions indeed do differ in their capabilities to identify spectra or (ii) the ionbot results files do not use the same format to report protein groups. As we filtered the peptide identifications on proteotypicality, this could cause peptides filtered out in one version to be included in the other.

The last form of variation that was explored is project-specific bias. Indeed, this form of bias is an amalgamation of known and unknown sources of variability related to instrumentation, sample handling, experimental design amongst others. However, identifying this source of variability can be dubious as it is also possible the same cell lines are in fact biologically different, or that the grouping employed, which is based on histopathology, is not accurate. Therefore, in each step of the workflow multiple methods were tried with a double focus in mind: (i) optimise model performance and (ii) prevent the model to overfit on the project-level. To measure this, we used two forms of cross validation strategies, namely, a stratified k-fold split (SKFCV), which is a standard in the field, and a leave-one-project-out splitting (LOPOCV) strategy which measures how much the model can see past project-related bias. Although cross validation is the standard method used to measure model performance, with the limited data we had at our disposal, these metrics cannot be fully trusted. Therefore, the behaviour of each method in the pre-processing pipeline was documented and used to measure the consistency and reliability of the data processing throughout the analysis.

Two types of project-related bias were identified. The first type is associated with quantification, where deeper proteome analyses tend to underestimated protein abundances compared to shallow proteome analyses. The second type is linked to protein identifications, where samples within the same project tend to share more identified proteins than samples from different projects. Subsequent paragraphs will discuss each of these biases in detail.

The first type of systematic project-related bias is a tendency of NSAF-quantification to underestimate abundances from samples with a larger set of protein identifications in comparison to shallower proteome analyses. Most likely this is caused by the normalisation factor used during the calculation of NSAF. After the spectral counts are normalised for protein length, each value is divided by the sum of all abundance values in the sample. The goal of this normalisation is to improve comparability between samples. However, as more proteins are identified, the size of the normalisation factor becomes larger while the spectral counts of an individual protein does not necessarily increase in the same manner. To account for this, we tried three additional normalisation methods, whereof the quantile normalisation was seen to perform optimally. Interestingly, ComBat performed significantly worse than any of our other normalisation methods. Although this method is widely used to reduce batch effects when aggregating disparate datasets, in our case, the biological variation was also reduced which led to a decrease in classification performance. As we defined a batch as a project, and the fact that often only one or a few groups are represented in a project, this method could not distinguish accurately between project-related variation and biological variation. However, if groups were represented by more projects, this method could perform better.

The second type of project-related bias is the project-specific identification of a set of proteins. There were two ways we handled this issue: (i) dropping proteins that are not identified in more than 50% of all samples and (ii) imputation. Although we lose a lot of information by dropping more than 11,000 protein features, most samples contained less than 3000 proteins, which if completed by imputation, questions the validity of the resulting sample. Therefore, we first reduced the missing values by filtering before applying imputation. We must note that each imputation method showed a strong yet different effect on feature selection. This suggests that imputation guides the selection of features. Therefore, it is important to determine that the imputation does not affect the true biological patterns present in the data in a negative way, and instead, makes it easier for machine learning models to recognise them. To evaluate this, LOPOCV was used. LOD-imputation consistently performed optimally over all differently normalised datasets and machine learning models and was therefore selected as the preferred imputation method. However, we do note that LOD-imputation simulates missingness and thus cannot entirely eliminate bias stemming from project-specific identifications. However, as LOD-imputation could more correctly classify samples from unseen projects than the other imputation methods, we believe that the missingness of proteins is more specific to the group of cell lines rather than project-related biases.

Although we saw good performance after imputation and normalisation based on SKFCV, the classification performance reached a maximum of only 74% when using LOPOCV. This indicates project specific overfitting. To reduce this possibility, stringent feature selection was performed.

Through feature selection, the fPEMatrix was further reduced to only contain 161 proteins (6.2% of the filtered dataset) without affecting the performance measured with SKFCV. Although only a limited amount of data remained for classification, this task was not deemed impossible. A similar, although more large-scale study performed by Goncalves et al, quantified 8498 proteins across 949 cell lines and was able to accurately predict drug responses using only 1500 randomly selected proteins [22]. This could be explained by the highly connected and co-regulated nature of proteins. This finding corroborates in part with our observation that 6.2% of our prefiltered dataset

was enough to retain satisfactory classification performance. Similarly to the study of Goncalves et al, we were able to detect protein-protein interactions by protein correlation in our data. Furthermore, by performing hierarchical clustering analysis on the 161-protein dataset, we determined several functionally associated clusters. Although we did not further reduce our dataset based on these clusters, other studies have described a methodology to do this[58]. Indeed, picking one protein representative for each cluster brings additional advantages: (i) multicollinearity, which is detrimental for the stability of the model parameters and thus the feature importance estimation, is reduced and (ii) interpreting the feature importance of a key protein can be directly interpreted by extrapolating to the biological context, i.e. the clusters. Taken together, our clustering analysis showed that the proposed approach could be feasible to apply on our dataset.

One limitation of correlation clustering, especially with the goal of reducing the feature space, is that only linear relationships can be captured. We showed, although in a preliminary and visually-based fashion, that variational autoencoders are able to describe the 161-protein dataset in six variables. Some of these variables were seen to group samples from different projects but the same group together. A similar approach was taken by a recent study from Way et al, which based on transcriptomics data showed that Variational Autoencoders are able to describe 5000 genes in 100 latent variables[66]. By using one hidden layer only, i.e. the layer containing the latent distribution parameters, interpretation was possible by capturing the weights of an input gene to a latent variable. Based on both the ability of an encoding to separate either the sex or the metastatic character of a melanoma and the weights of the input genes to those encodings, a biological interpretation could be made of what the encodings represent. As we used a hidden layer for both the encoder and decoder network, such an analysis was not possible in our case, thus we cannot safely say the encodings capture a biological meaningful representation of the data. However, by using extra hidden layers, more complex patterns can be learned, which was more preferable than interpretability since we initially used the VAE to meet the challenge of class imbalance.

As we showed in figure 3, our dataset is imbalanced both in terms of samples and of projects. As machine learning algorithms tend to be biased towards the majority classes, this imbalance needs to be addressed. Otherwise, the model will be unable to accurately learn patterns from the minority class, leading to a decrease in predictive performance. Imbalance can be mediated by removing samples from the majority class, which is not feasible due to our already small dataset. Another solution involves assigning higher weights to samples from minority classes, however, this does not directly deal with the problem of the lack of data. As we do not have access to more data of the minority classes, new synthetic samples that are similar to the other samples must be generated. Therefore, three SMOTE-based methods and one VAE-based method were tried.

With SMOTE, samples are generated as interpolations in feature space of neighbouring minority samples of a class. By making interpolations utilising multiple projects, we additionally hope to decrease project specific bias. However, as can be seen in figure 25, if the intraclass variability is too high, this could create samples that are unlike any other sample, questioning the biological validity of the generation. This observation raises an important question: how to effectively evaluate the biological validity of the data generation process? In an attempt to address this question, statistical tests were employed to analyse the variance, distribution and mean of the features. Although no significant deviations were observed, it is important to acknowledge that a cell's phenotype encompasses more than just the sum of its proteins. Simply generating a sample by interpolating protein-by-protein fails to consider the intricate biological interactions that occur among these proteins. This inherent limitation of SMOTE highlights the advantage of utilising Variational Autoencoders (VAE). VAEs tries to describe the underlying patterns of the complete dataset in a latent space, accounting for complex biological interactions. By leveraging the

generative capabilities of VAEs, synthetic samples can be generated that capture the biological realism of the original data. Although promising, we observed that VAEs created statistically less similar datapoints than SMOTE. Additionally, in line with SMOTE, VAE also generated datapoints with lower variance. This is believed to be in part caused by the mean squared error term in the loss function, which encourages to reconstruct the mean value of a feature if the network is unable to learn to reconstruct it effectively[83]. On the other hand, the noise sampling process during reparameterization itself could also be responsible[83]. Improvements can be made by renormalising the generated data to match the variance of the original data or through a better balancing between the Kullback-Leibler term and reconstruction error to maximise the potential of the network. Since in our case, we can observe some latent distributions that perfectly fit the prior distribution, we do think these additions will proof beneficial and if the model is properly optimised can be more useful than SMOTE in creating biologically realistic datapoints to combat the class imbalance.

By using feature selection and oversampling the performance of the model improved based on the LOPOCV to an average of 93.7%, whereof only five projects were not able to be perfectly predicted. After a preliminary biological interpretation of the model, we observed that the model classifies cell line groups based on characteristics that are in corroboration with the literature. However, further validation is necessary to strengthen this claim as due to the lack of data, it is highly likely that the model, although demonstrating a high classification performance, is still overfitted. Nonetheless, more data can be added and the methods used further optimised and extended to improve the robustness of the model.

Limited similarities were observed between the proteins identified as important for cell line and tissue classification, indicating that differences in cell lines are measured differently compared to tissues. An intriguing avenue to explore would be comparing tumour samples in a similar manner as described for tissues. This comparative analysis would enable the identification of overlapping and non-overlapping aberrations between in vivo tumours and cell lines in a highly interpretable manner. Building on these findings, potential treatment options could be proposed by targeting the highlighted pathways that are affected by compounds proven to be effective on these cell lines. While this claim is speculative, cell line models have been extensively researched and continue to be a focus of study today. Profiling data from large cell line panels at the genomic, transcriptomic and proteomic levels, combined with extensive compound screening experiments, have generated a wealth of data. Integrating these diverse data types holds the potential to reveal novel connections and patterns, serving as a platform for discovery for biologists studying cancer and other diseases.

9. General conclusion

In this study, we have successfully demonstrated the accurate classification of cell line groups with a 93.7% performance. Our utilisation of SHAP enables subsequent biological interpretation of the model. We showed that in our dataset, correlated proteins are indicative of functional association, which can be utilised to provide a biological context for the most significant features. However, it is important to acknowledge the limitations associated with repurposing and amalgamating public data, which primarily include the absence of metadata and the presence of batch effects. Our investigation of the batch effects highlights that sample preparation workflows differ in terms of the hydrophobic composition of the proteins that are extracted. To mitigate such biases, we implemented multiple methods at each pre-processing step. By comparing the downstream effects within the pre-processing pipeline, we observed that the choice of method does influence the subsequent steps. Despite these challenges, our chosen approach yields a highly accurate model for classifying cell line groups, which holds value for discovering the differences between cell lines.

10. Reference List

1 Falzone, L., Salomone, S. & Libra, M. Evolution of Cancer Pharmacological Treatments at the Turn of the Third Millennium. *Front Pharmacol* **9**, 1300, doi:10.3389/fphar.2018.01300 (2018).

2 Liu, J., Dang, H. & Wang, X. W. The significance of intertumor and intratumor heterogeneity in liver cancer. *Experimental & Molecular Medicine* **50**, e416-e416, doi:10.1038/emm.2017.165 (2018).

3 Turashvili, G. & Brogi, E. Tumor Heterogeneity in Breast Cancer. *Front Med (Lausanne)* **4**, 227, doi:10.3389/fmed.2017.00227 (2017).

4 Mirabelli, P., Coppola, L. & Salvatore, M. Cancer Cell Lines Are Useful Model Systems for Medical Research. *Cancers* **11** (2019).

5 Rijal, G. & Li, W. Native-mimicking in vitro microenvironment: an elusive and seductive future for tumor modeling and tissue engineering. *Journal of Biological Engineering* **12**, 20, doi:10.1186/s13036-018-0114-7 (2018).

6 Hanahan, D. & Weinberg, Robert A. Hallmarks of Cancer: The Next Generation. *Cell* **144**, 646-674, doi:https://doi.org/10.1016/j.cell.2011.02.013 (2011).

7 Arora, M. Cell culture media: a review. *Mater methods* **3**, 24 (2013).

8 Ghandi, M. *et al.* Next-generation characterization of the Cancer Cell Line Encyclopedia. *Nature* **569**, 503-508, doi:10.1038/s41586-019-1186-3 (2019).

9 Guo, T. *et al.* Quantitative Proteome Landscape of the NCI-60 Cancer Cell Lines. *iScience* **21**, 664-680, doi:10.1016/j.isci.2019.10.059 (2019).

10 Barretina, J. *et al.* The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature* **483**, 603-607, doi:10.1038/nature11003 (2012).

11 Domcke, S., Sinha, R., Levine, D. A., Sander, C. & Schultz, N. Evaluating cell lines as tumour models by comparison of genomic profiles. *Nature Communications* **4**, 2126, doi:10.1038/ncomms3126 (2013).

12 Ben-David, U. *et al.* Genetic and transcriptional evolution alters cancer cell line drug response. *Nature* **560**, 325-330, doi:10.1038/s41586-018-0409-3 (2018).

13 Gisselsson, D., Lindgren, D., Mengelbier, L. H., Øra, I. & Yeger, H. Genetic bottlenecks and the hazardous game of population reduction in cell line based research. *Experimental Cell Research* **316**, 3379-3386, doi:https://doi.org/10.1016/j.yexcr.2010.07.010 (2010).

14 Yang, X., Wen, Y., Song, X., He, S. & Bo, X. Exploring the classification of cancer cell lines from multiple omic views. *PeerJ* **8**, e9440, doi:10.7717/peerj.9440 (2020).

15 Sud, A., Kinnersley, B. & Houlston, R. S. Genome-wide association studies of cancer: current insights and future perspectives. *Nat Rev Cancer* **17**, 692-704, doi:10.1038/nrc.2017.82 (2017).

16 Tam, V. *et al.* Benefits and limitations of genome-wide association studies. *Nature Reviews Genetics* **20**, 467-484, doi:10.1038/s41576-019-0127-1 (2019).

17 Dugger, S. A., Platt, A. & Goldstein, D. B. Drug development in the era of precision medicine. *Nat Rev Drug Discov* **17**, 183-196, doi:10.1038/nrd.2017.226 (2018).

18 Boyle, E. A., Li, Y. I. & Pritchard, J. K. An Expanded View of Complex Traits: From Polygenic to Omnigenic. *Cell* **169**, 1177-1186, doi:10.1016/j.cell.2017.05.038 (2017).

19 Correa Rojo, A. *et al.* Towards Building a Quantitative Proteomics Toolbox in Precision Medicine: A Mini-Review. *Frontiers in Physiology* **12** (2021).

20 Liu, Y., Beyer, A. & Aebersold, R. On the Dependency of Cellular Protein Levels on mRNA Abundance. *Cell* **165**, 535-550, doi:https://doi.org/10.1016/j.cell.2016.03.014 (2016).

21 Nusinow, D. P. *et al.* Quantitative Proteomics of the Cancer Cell Line Encyclopedia. *Cell* **180**, 387-402.e316, doi:https://doi.org/10.1016/j.cell.2019.12.023 (2020).

22 Gonçalves, E. *et al.* Pan-cancer proteomic map of 949 human cell lines. *Cancer Cell* **40**, 835-849.e838, doi:10.1016/j.ccell.2022.06.010 (2022).

23 Gholami, Amin M. *et al.* Global Proteome Analysis of the NCI-60 Cell Line Panel. *Cell Reports* **4**, 609-620, doi:https://doi.org/10.1016/j.celrep.2013.07.018 (2013).

24 Jarnuczak, A. F. *et al.* An integrated landscape of protein expression in human cancer. *Scientific Data* **8**, 115, doi:10.1038/s41597-021-00890-2 (2021).

25 Kustatscher, G. *et al.* Co-regulation map of the human proteome enables identification of protein functions. *Nat Biotechnol* **37**, 1361-1371, doi:10.1038/s41587-019-0298-5 (2019).

26 Schwarz, A. & Beck, M. The Benefits of Cotranslational Assembly: A Structural Perspective. *Trends Cell Biol* **29**, 791-803, doi:10.1016/j.tcb.2019.07.006 (2019).

27 Perez-Riverol, Y. *et al.* The PRIDE database resources in 2022: a hub for mass spectrometry-based proteomics evidences. *Nucleic Acids Research* **50**, D543-D552, doi:10.1093/nar/gkab1038 (2022).

28 Claeys, T., Menu, M., Bouwmeester, R., Gevaert, K. & Martens, L. Machine Learning on Large-Scale Proteomics Data Identifies Tissue and Cell-Type Specific Proteins. *Journal of Proteome Research*, doi:10.1021/acs.jproteome.2c00644 (2023).

29 Duong, V.-A. & Lee, H. Bottom-Up Proteomics: Advancements in Sample Preparation. *International Journal of Molecular Sciences* **24** (2023).

30 Glatter, T., Ahrné, E. & Schmidt, A. Comparison of Different Sample Preparation Protocols Reveals Lysis Buffer-Specific Extraction Biases in Gram-Negative Bacteria and Human Cells. *Journal of Proteome Research* **14**, 4472-4485, doi:10.1021/acs.jproteome.5b00654 (2015).

31 Varnavides, G. *et al.* In Search of a Universal Method: A Comparative Survey of Bottom-Up Proteomics Sample Preparation Methods. *J Proteome Res* **21**, 2397-2411, doi:10.1021/acs.jproteome.2c00265 (2022).

32 Feist, P. & Hummon, A. B. Proteomic challenges: sample preparation techniques for microgram-quantity protein analysis from biological samples. *Int J Mol Sci* **16**, 3537-3563, doi:10.3390/ijms16023537 (2015).

33 Rozanova, S. *et al.* in *Quantitative Methods in Proteomics* (eds Katrin Marcus, Martin Eisenacher, & Barbara Sitek) 85-116 (Springer US, 2021).

34 Davies, V. *et al.* Rapid Development of Improved Data-Dependent Acquisition Strategies. *Analytical Chemistry* **93**, 5676-5683, doi:10.1021/acs.analchem.0c03895 (2021).

35 Kreimer, S. *et al.* Advanced Precursor Ion Selection Algorithms for Increased Depth of Bottom-Up Proteomic Profiling. *J Proteome Res* **15**, 3563-3573, doi:10.1021/acs.jproteome.6b00312 (2016).

36 Lin, T.-T. *et al.* Mass spectrometry-based targeted proteomics for analysis of protein mutations. *Mass Spectrometry Reviews* **42**, e21741, doi:https://doi.org/10.1002/mas.21741 (2023).

37 Verheggen, K. *et al.* Anatomy and evolution of database search engines—a central component of mass spectrometry based proteomic workflows. *Mass Spectrometry Reviews* **39**, 292-306, doi:https://doi.org/10.1002/mas.21543 (2020).

38 The UniProt, C. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research* **51**, D523-D531, doi:10.1093/nar/gkac1052 (2023).

39 Degroeve, S. *et al.* ionbot: a novel, innovative and sensitive machine learning approach to LC-MS/MS peptide identification. *bioRxiv*, 2021.2007.2002.450686, doi:10.1101/2021.07.02.450686 (2022).

40 Dupree, E. J. *et al.* A Critical Review of Bottom-Up Proteomics: The Good, the Bad, and the Future of this Field. *Proteomes* **8**, doi:10.3390/proteomes8030014 (2020).

41 Bouwmeester, R., Gabriels, R., Hulstaert, N., Martens, L. & Degroeve, S. DeepLC can predict retention times for peptides that carry as-yet unseen modifications. *Nat Methods* **18**, 1363-1369, doi:10.1038/s41592-021-01301-5 (2021).

42 Declercq, A. *et al.* Updated MS²PIP web server supports cutting-edge proteomics applications. *Nucleic Acids Research*, gkad335, doi:10.1093/nar/gkad335 (2023).

43 Ahrné, E., Molzahn, L., Glatter, T. & Schmidt, A. Critical assessment of proteome-wide label-free absolute abundance estimation strategies. *PROTEOMICS* **13**, 2567-2578, doi:https://doi.org/10.1002/pmic.201300135 (2013).

44 Brenes, A., Hukelmann, J., Bensaddek, D. & Lamond, A. I. Multibatch TMT Reveals False Positives, Batch Effects and Missing Values. *Mol Cell Proteomics* **18**, 1967-1980, doi:10.1074/mcp.RA119.001472 (2019).

45 Florens, L. *et al.* Analyzing chromatin remodeling complexes using shotgun proteomics and normalized spectral abundance factors. *Methods* **40**, 303-311, doi:10.1016/j.ymeth.2006.07.028 (2006).

46 Wiśniewski, J. R., Hein, M. Y., Cox, J. & Mann, M. A "proteomic ruler" for protein copy number and concentration estimation without spike-in standards. *Mol Cell Proteomics* **13**, 3497-3506, doi:10.1074/mcp.M113.037309 (2014).

47 Bubis, J. A., Levitsky, L. I., Ivanov, M. V., Tarasova, I. A. & Gorshkov, M. V. Comparative evaluation of label-free quantification methods for shotgun proteomics. *Rapid Communications in Mass Spectrometry* **31**, 606-612, doi:https://doi.org/10.1002/rcm.7829 (2017).

48 Chawade, A., Alexandersson, E. & Levander, F. Normalyzer: A Tool for Rapid Evaluation of Normalization Methods for Omics Data Sets. *Journal of Proteome Research* **13**, 3114-3120, doi:10.1021/pr401264n (2014).

49 Callister, S. J. *et al.* Normalization Approaches for Removing Systematic Biases Associated with Mass Spectrometry and Label-Free Proteomics. *Journal of Proteome Research* **5**, 277-286, doi:10.1021/pr050300l (2006).

50 Välikangas, T., Suomi, T. & Elo, L. L. A systematic evaluation of normalization methods in quantitative label-free proteomics. *Briefings in Bioinformatics* **19**, 1-11, doi:10.1093/bib/bbw095 (2018).

51 Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118-127, doi:10.1093/biostatistics/kxj037 (2007).

52 Lazar, C., Gatto, L., Ferro, M., Bruley, C. & Burger, T. Accounting for the Multiple Natures of Missing Values in Label-Free Quantitative Proteomics Data Sets to Compare Imputation Strategies. *Journal of Proteome Research* **15**, 1116-1125, doi:10.1021/acs.jproteome.5b00981 (2016).

53 Desaire, H., Go, E. P. & Hua, D. Advances, obstacles, and opportunities for machine learning in proteomics. *Cell Reports Physical Science* **3**, 101069, doi:https://doi.org/10.1016/j.xcrp.2022.101069 (2022).

54 Webb-Robertson, B.-J. M. *et al.* Review, Evaluation, and Discussion of the Challenges of Missing Value Imputation for Mass Spectrometry-Based Label-Free Global Proteomics. *Journal of Proteome Research* **14**, 1993-2001, doi:10.1021/pr501138h (2015).

55 McCoy, J. T., Kroon, S. & Auret, L. Variational Autoencoders for Missing Data Imputation with Application to a Simulated Milling Circuit. *IFAC-PapersOnLine* **51**, 141-146, doi:https://doi.org/10.1016/j.ifacol.2018.09.406 (2018).

56 Lualdi, M. & Fasano, M. in *Proteomics Data Analysis* (ed Daniela Cecconi) 143-159 (Springer US, 2021).

57 Aboudi, N. E. & Benhlima, L. in *2016 International Conference on Engineering & MIS (ICEMIS).* 1-5.

58 Shi, Z., Wen, B., Gao, Q. & Zhang, B. Feature Selection Methods for Protein Biomarker Discovery from Proteomics or Multiomics Data. *Molecular & Cellular Proteomics* **20**, 100083, doi:https://doi.org/10.1016/j.mcpro.2021.100083 (2021).

59 Nanga, S. *et al.* Review of dimension reduction methods. *J. Data Anal. Inf. Process.* **09**, 189-231, doi:10.4236/jdaip.2021.93013 (2021).

60 Sharma, S., Gosain, A. & Jain, S. in *International Conference on Innovative Computing and Communications.* (eds Ashish Khanna *et al.*) 459-472 (Springer Singapore).

61 Kovács, G. An empirical comparison and evaluation of minority oversampling techniques on a large number of imbalanced datasets. *Applied Soft Computing* **83**, 105662, doi:https://doi.org/10.1016/j.asoc.2019.105662 (2019).

62 Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. SMOTE: Synthetic Minority over-Sampling Technique. *J. Artif. Int. Res.* **16**, 321–357 (2002).

63 Fernández, A., Garcia, S., Herrera, F. & Chawla, N. SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *Journal of Artificial Intelligence Research* **61**, 863-905, doi:10.1613/jair.1.11192 (2018).

64 Islam, Z., Abdel-Aty, M., Cai, Q. & Yuan, J. Crash data augmentation using variational autoencoder. *Accident Analysis & Prevention* **151**, 105950, doi:https://doi.org/10.1016/j.aap.2020.105950 (2021).

65 Asperti, A. & Trentin, M. Balancing Reconstruction Error and Kullback-Leibler Divergence in Variational Autoencoders. *IEEE Access* **8**, 199440-199448, doi:10.1109/ACCESS.2020.3034828 (2020).

66 Way, G. P. & Greene, C. S. Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders. *Pac Symp Biocomput* **23**, 80-91 (2018).

67 Lundberg, S. M. & Lee, S.-I. in *Advances in Neural Information Processing Systems* Vol. 30 (eds I. Guyon *et al.*) (Curran Associates, Inc., 2017).
68 Bairoch, A. The Cellosaurus, a Cell-Line Knowledge Resource. *J Biomol Tech* **29**, 25-38, doi:10.7171/jbt.18-2902-002 (2018).
69 The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Res* **49**, D325-d334, doi:10.1093/nar/gkaa1113 (2021).
70 Uhlén, M. *et al.* Proteomics. Tissue-based map of the human proteome. *Science* **347**, 1260419, doi:10.1126/science.1260419 (2015).
71 Szklarczyk, D. *et al.* The STRING database in 2023: protein-protein association networks and functional enrichment analyses for any sequenced genome of interest. *Nucleic Acids Res* **51**, D638-d646, doi:10.1093/nar/gkac1000 (2023).
72 Kyte, J. & Doolittle, R. F. A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology* **157**, 105-132, doi:https://doi.org/10.1016/0022-2836(82)90515-0 (1982).
73 Abdelkader, B., Julien, H., Chloé-Agathe, A. & Akpéli, N. pyComBat, a Python tool for batch effects correction in high-throughput molecular data using empirical Bayes methods. *bioRxiv*, 2020.2003.2017.995431, doi:10.1101/2020.03.17.995431 (2021).
74 Slany, A. *et al.* Contribution of Human Fibroblasts and Endothelial Cells to the Hallmarks of Inflammation as Determined by Proteome Profiling. *Molecular & cellular proteomics : MCP* **15**, 1982-1997, doi:10.1074/mcp.m116.058099 (2016).
75 Nakada, M. *et al.* Integrin α3 is overexpressed in glioma stem-like cells and promotes invasion. *Br J Cancer* **108**, 2516-2524, doi:10.1038/bjc.2013.218 (2013).
76 Toyoda, H., Nagai, Y., Kojima, A. & Kinoshita-Toyoda, A. Podocalyxin as a major pluripotent marker and novel keratan sulfate proteoglycan in human embryonic and induced pluripotent stem cells. *Glycoconjugate Journal* **34**, 817-823, doi:10.1007/s10719-017-9801-8 (2017).
77 Bernal, A. & Arranz, L. Nestin-expressing progenitor cells: function, identity and therapeutic implications. *Cell Mol Life Sci* **75**, 2177-2195, doi:10.1007/s00018-018-2794-z (2018).
78 Patel, V. J. *et al.* A Comparison of Labeling and Label-Free Mass Spectrometry-Based Proteomics Approaches. *Journal of Proteome Research* **8**, 3752-3759, doi:10.1021/pr900080y (2009).
79 Saltzman, A. B. *et al.* gpGrouper: A Peptide Grouping Algorithm for Gene-Centric Inference and Quantitation of Bottom-Up Proteomics Data. *Mol Cell Proteomics* **17**, 2270-2283, doi:10.1074/mcp.TIR118.000850 (2018).
80 Claeys, T. *et al.* (Research Square Platform LLC, 2023).
81 Choksawangkarn, W., Edwards, N., Wang, Y., Gutierrez, P. & Fenselau, C. Comparative study of workflows optimized for in-gel, in-solution, and on-filter proteolysis in the analysis of plasma membrane proteins. *J Proteome Res* **11**, 3030-3034, doi:10.1021/pr300188b (2012).
82 Tiwari, P., Kaila, P. & Guptasarma, P. Understanding anomalous mobility of proteins on SDS-PAGE with special reference to the highly acidic extracellular domains of human E- and N-cadherins. *Electrophoresis* **40**, 1273-1281, doi:10.1002/elps.201800219 (2019).
83 Asperti, A. in *Machine Learning, Optimization, and Data Science.* (eds Giuseppe Nicosia *et al.*) 297-308 (Springer International Publishing).

11. Poster

A machine learning approach to identify cell line specific protein patterns based on public proteomic data

Sam van Puyenbroeck, Tine Claeys[1,2], Lennart Martens[1,2]

1. VIB-UGent Center for Medical Biotechnology, VIB, Zwijnaarde, Belgium
2. Department of Biomolecular Medicine, Ghent University, Ghent, Belgium

Metadata

- +25 cell lines
- 10 tissue types
- Treated or untreated
- 3 types of sample preparation

ionbot

PRIDE projects

- 46 projects
- 621 samples
- 63,413,473 significant spectra
- 423,650 unique peptide sequences
- 14,507 unique proteins identified

Introduction

Cell lines are widely used in medical research. They provide a **standardized model** to test the efficacy of drugs in a high-throughput manner and allow to investigate human biology and diseases. Since the inception of the first cell line in the 1950s, many cell lines have been created. Besides their differences with real human cells, they differ from each genetically and, more importantly, in terms of their **protein composition** (1). Due to the widespread use of these highly standardized models and the emergence of more large-scale studies, a lot of proteomics data of cell lines is now publicly and freely available in repositories such as PRIDE (2). This allows to investigate their **biological differences and similarities**. In this work, **machine learning algorithms** are trained to classify cell lines based on protein expression. Analyzing which patterns of proteins are more important for classification will highlight **cell line specific characteristics** that are biologically meaningful and could in turn guide model selection for future experiments.

Materials and Methods

Data on protein expression of cell lines were retrieved from the PRIDE database. Only projects quantifying proteins in a label-free manner and not using enrichment procedures are used. Following a laborious manual metadata annotation initiative of the experimental data, the MS-spectra are reprocessed by ionbot to accurately identify and quantify peptides (3). The proteins are quantified with the Normalized Spectral Abundance Factor and stored in a protein expression matrix. The matrix and metadata are then used to evaluate multiple machine learning algorithms to classify cell lines, which will be evaluated by metrics such as accuracy and F1-score. The most important features that guide classification will be biologically interpreted by integrating knowledge from databases such as KEGG, STRING and the Human Protein Atlas and will be used to evaluate the differences between cell lines.

Results 1: Large variance between projects

The phenomenon of batch effects are a well-known problem in the biomedical field. Methods exist to limit these biologically irrelevant biases. However, they are mainly applied on large scale experiments with large sample sizes (4). In our dataset, many small datasets are combined making existing methods less ideal. Clustering all samples from the HeLa cell line based on overlapping protein identifications indicates the presence of batch effects (figure 1).

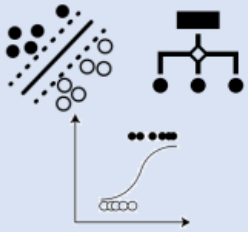
Results 2: Cell line classification is feasible

To reduce the feature space for machine learning modelling, proteins can be grouped based on the pathways they are involved in by using previous knowledge stored in public databases. Supervised machine learning algorithms can be used to learn which features are most distinct to a given cell and can be used to classify the cell lines. By utilizing different data splitting procedures during cross-validation, the performance of the model can be evaluated. When using all samples from a project only in the testing or training set, the performance drops significantly. This can be explained by the possibility that different projects identify a different subset of the cell line proteome or that cell lines are very heterogeneous across labs (5).

Protein expression matrix

	Protein A	Protein B	Protein C
Cell 1	0,9	0,1	0,8
Cell 2	0,4	0,15	0,5
Cell 3	0,4	0,7	0,6

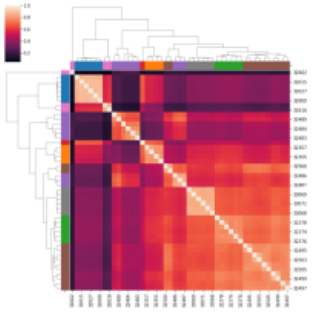
Machine learning classification of cell lines



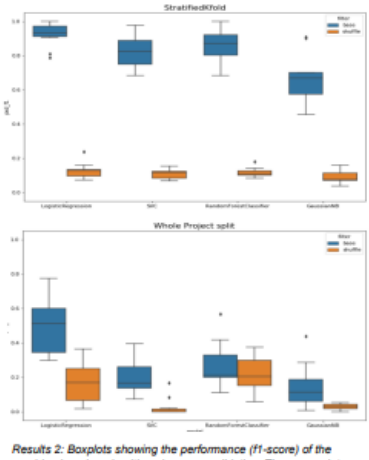
Biological interpretation

STRING

HPA



Results 1: Clustermap representing the overlap of identified proteins between samples of the HeLa cell line. The samples mainly cluster based on the project (colors) it originates from.



Results 2: Boxplots showing the performance (f1-score) of the machine learning algorithms by cross-validation. The upper plot utilized a StratifiedKFold split, the lower plot a complete project split methodology. The orange boxplots show the performance of classification when the labels are shuffled.

Future analyses

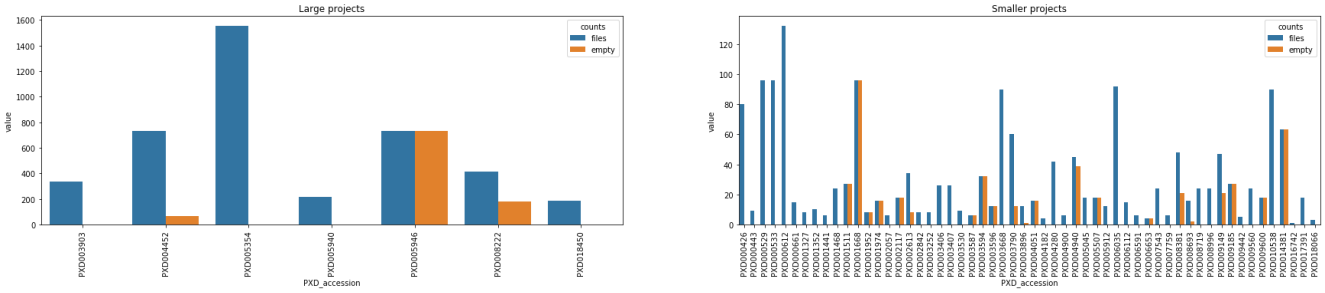
- Uncovering systematic peptide/protein identification differences based on sample preparation methods used
- Explore data generation methods to artificially largen the protein expression matrix. This could make the model more robust
- Biologically interpret the predictions made by the model

References:

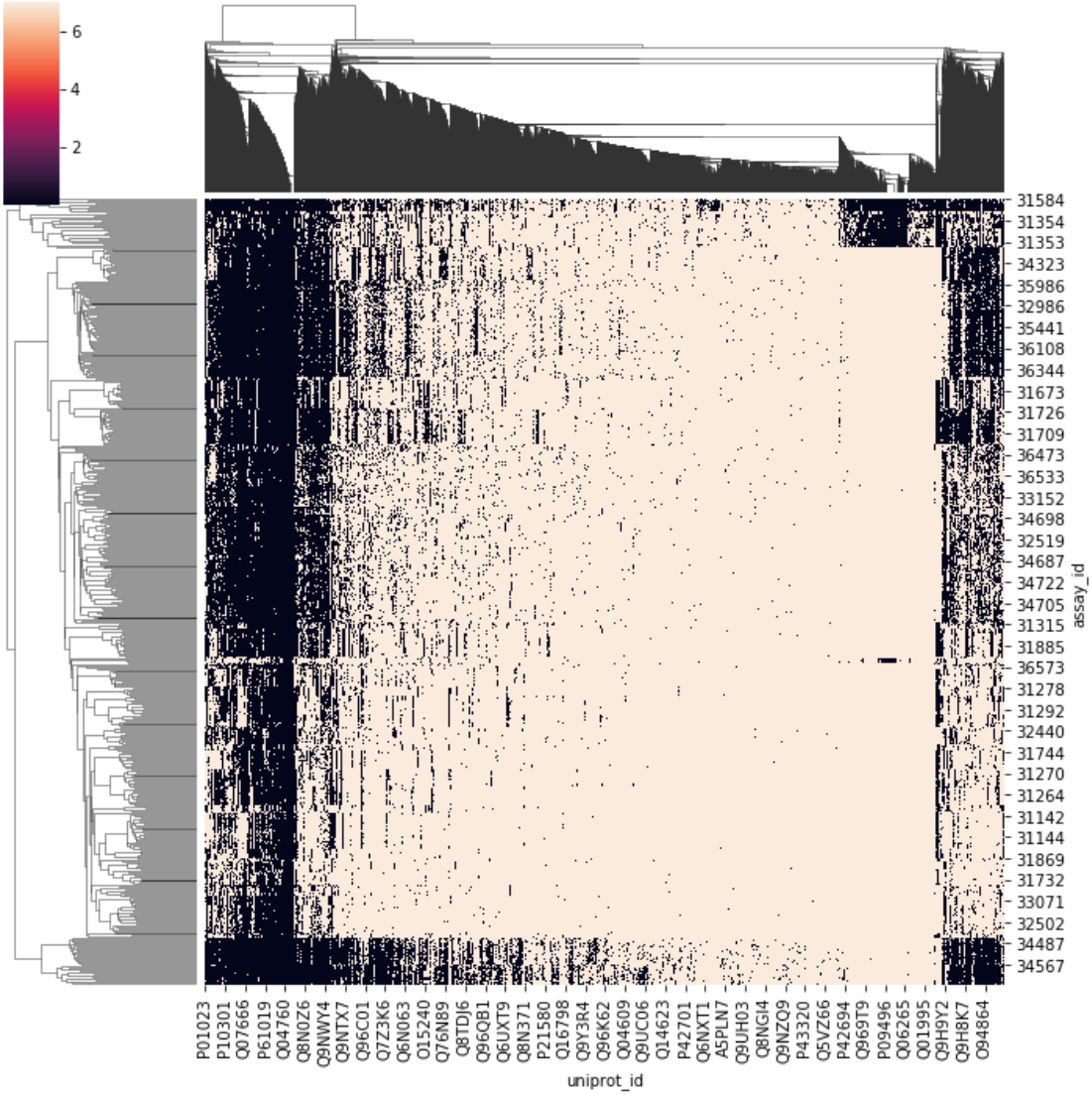
1. D. P. Nusinow et al., Quantitative Proteomics of the Cancer Cell Line Encyclopedia. Cell 180, 387-402.e316 (2020).
2. Y. Perez-Riverol et al., The PRIDE database resources in 2022: a hub for mass spectrometry-based proteomics evidences. Nucleic Acids Research 50, D543-D552 (2022).
3. S. Degroeve et al., ionbot: a novel, innovative and sensitive machine learning approach to LC-MS/MS peptide identification. bioRxiv, 2021.2007.2002.450686 (2022).
4. A. F. Jarnuczak et al., An integrated landscape of protein expression in human cancer. Scientific Data 8, 115 (2021).
5. Y. Liu *et al.*, Multi-omic measurements of heterogeneity in HeLa cells across laboratories. *Nature Biotechnology* **37**, 314-322 (2019).

UNIVERSITEIT GENT · VIB · FACULTY OF MEDICINE AND HEALTH SCIENCES · Compomics

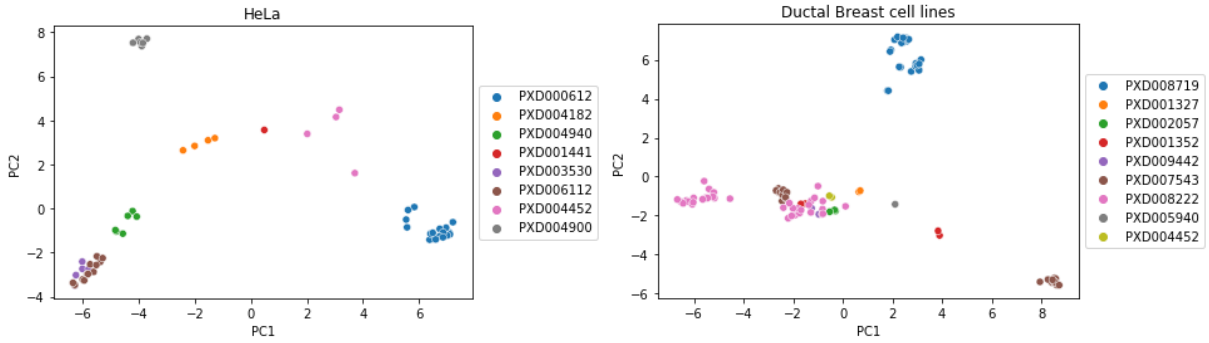
12. Addendum



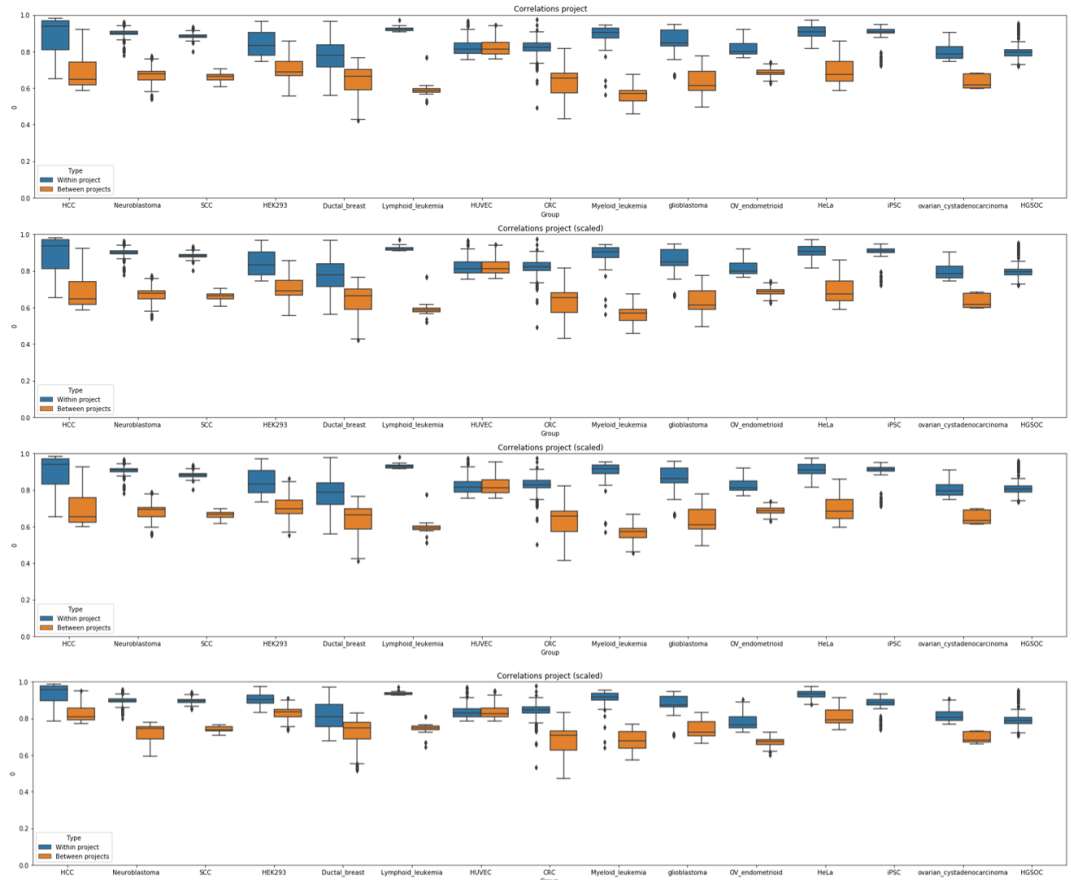
Supplementary figure 1: Summary of the empty ionbot result files per project for projects with (left) more than 200 samples and (right) less than 200 samples.



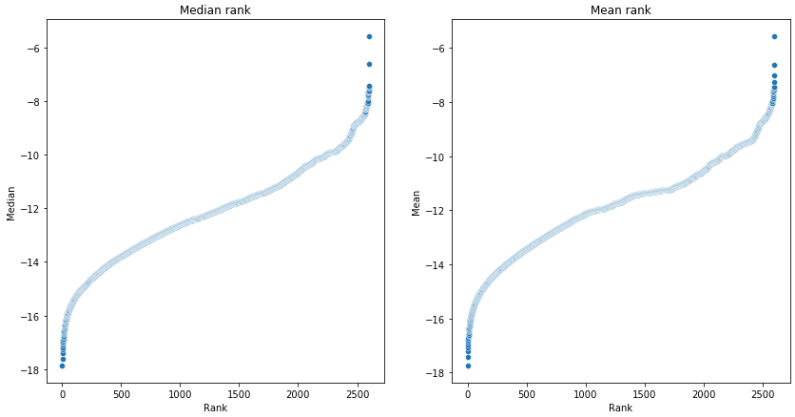
Supplementary figure 2: Visual illustration of missing values (in yellow) across samples.



Supplementary figure 3: Scatterplot of the first two principal components after PCA for samples belonging to the (left) HeLa and (right) ductal breast cancer cell line groups.



Supplementary figure 4: Boxplots showing pairwise correlations between samples within the same or different projects per group for different normalisation methods. From top to bottom these are log2-NSAF-normalised, Median normalised, Quantile normalised, and ComBat normalised.



Supplementary figure 5: The calculated reference values used during quantile normalisation using either (left) median or (right) mean.

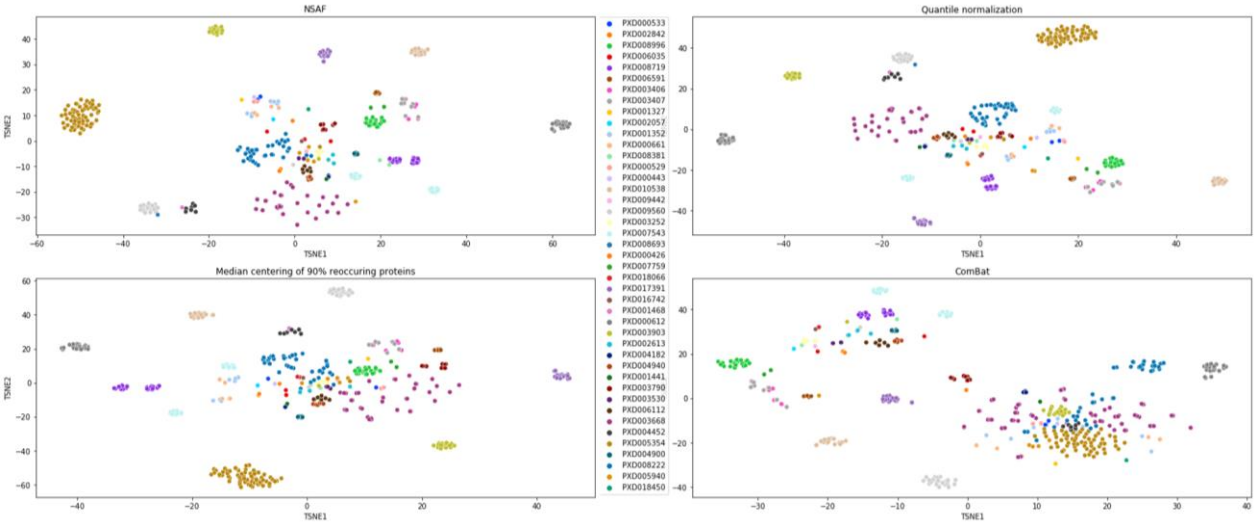
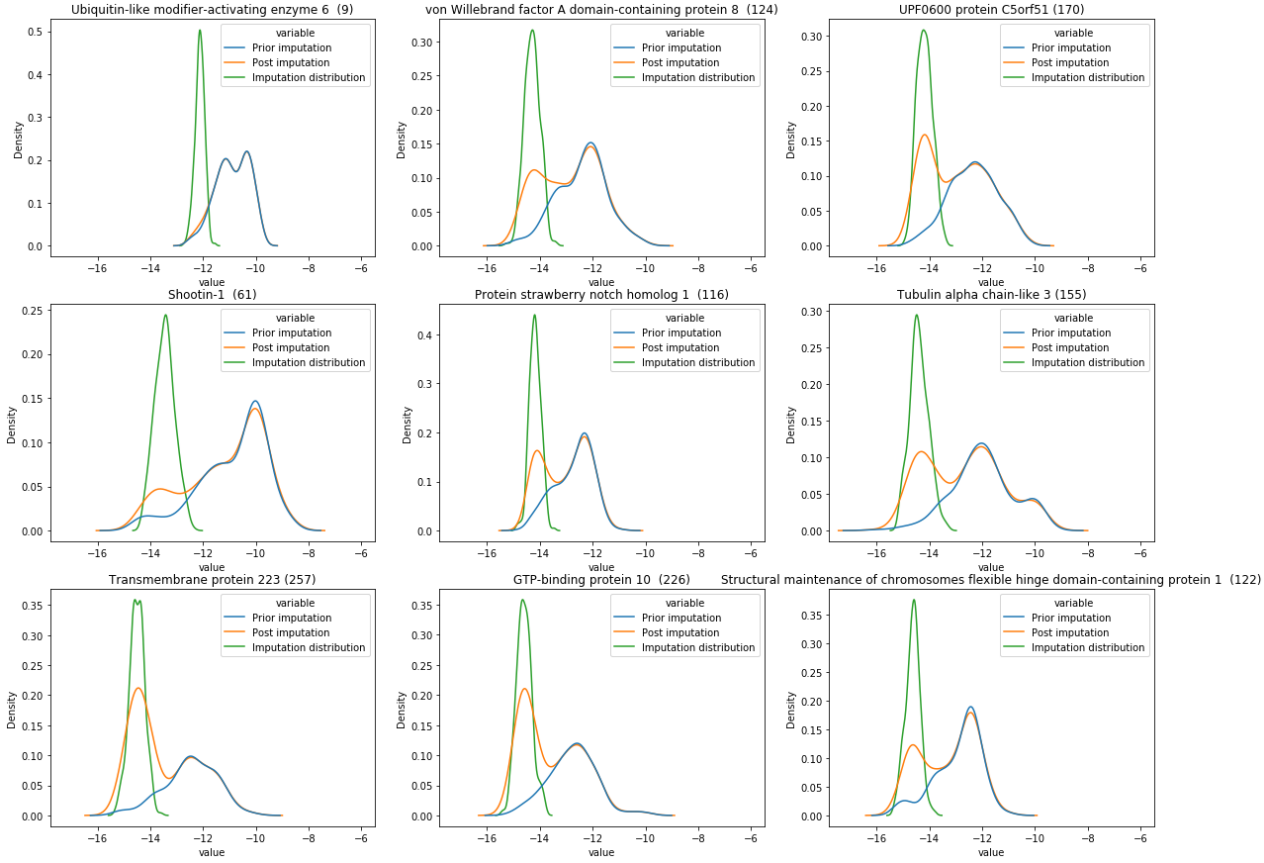
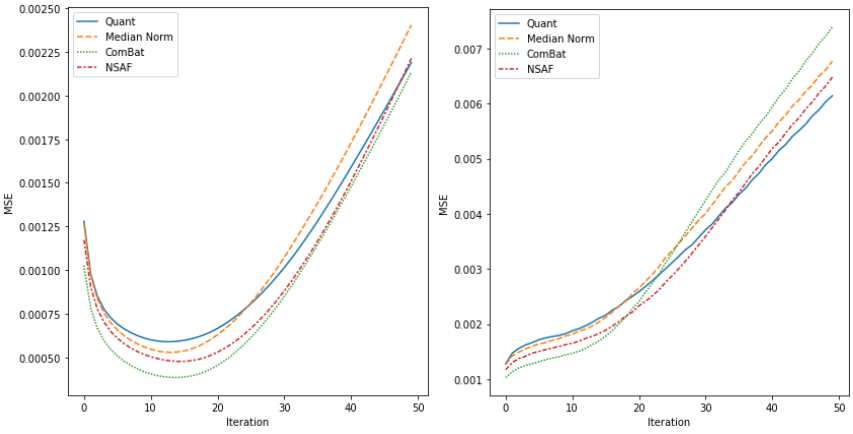


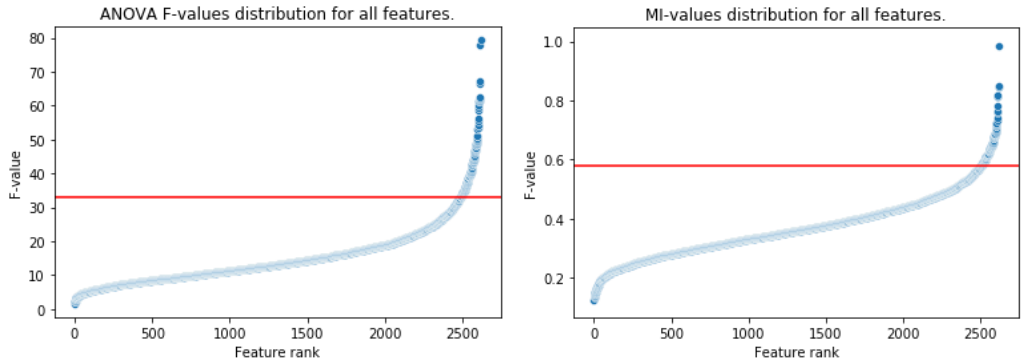
Figure 6: Similar to figure 13, 2D-representation after t-SNE transformation with perplexity 20 for differently normalised datasets. Colours indicate the project samples belong to.



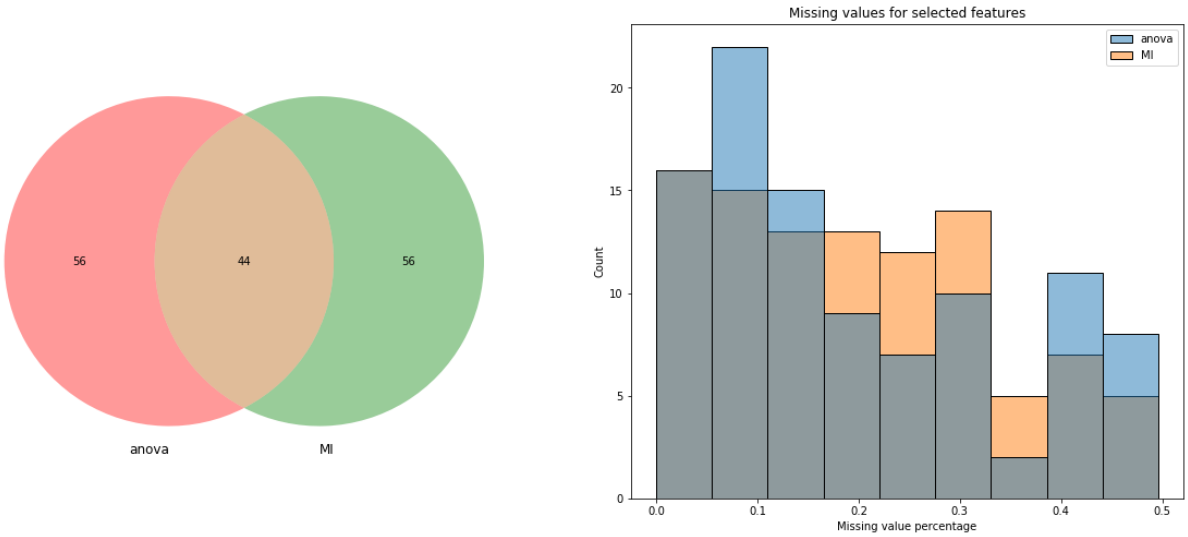
Supplementary figure 7: The distributions of protein abundances for the first nine proteins in alphabetical order by UniProt accession number before and after imputation. The green curve represents the gaussian distribution wherefrom the imputed values are drawn. The number next to the title of each subplot indicates the number of missing values for that protein out of 518 samples.



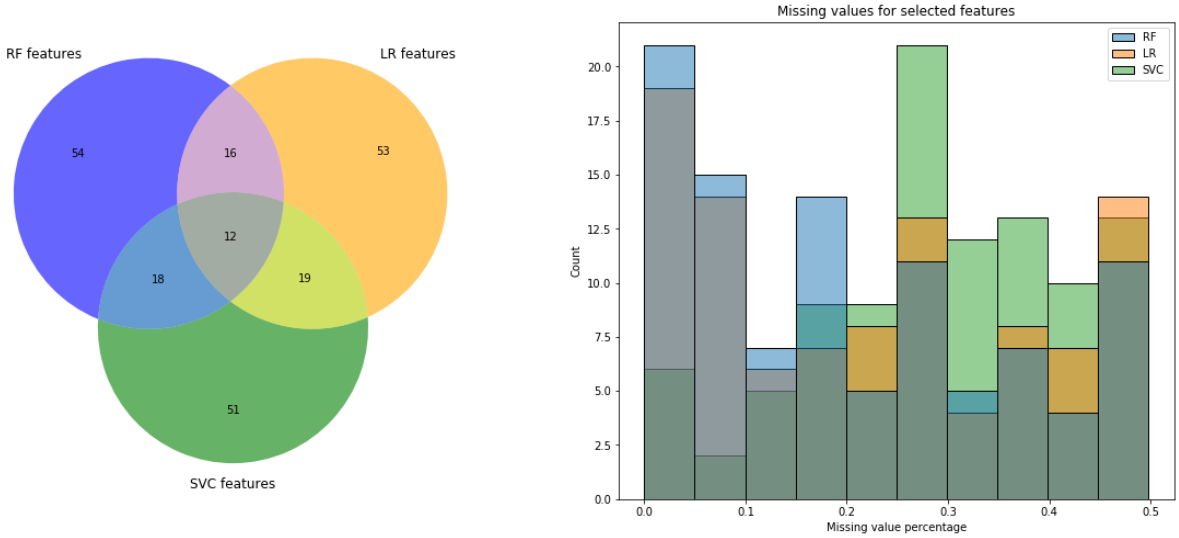
Supplementary figure 8: MSE of non-missing values for 50 PCA-reconstruction cycles while either keeping (left) the number of principal components or (right) the explained variance (95%) stable.



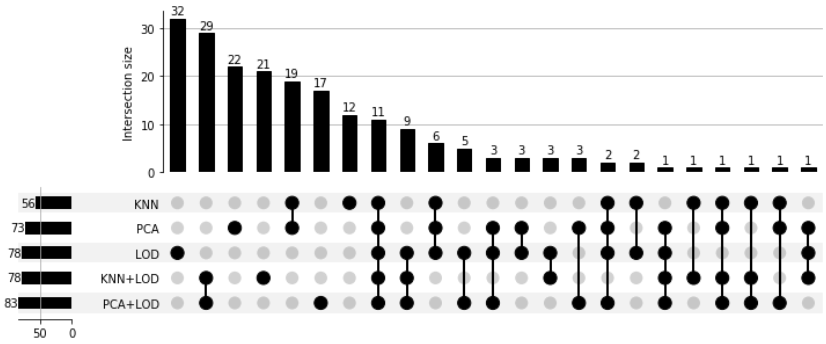
Supplementary figure 9: Sizes of assigned feature importance by two univariate feature selectors: (left) ANOVA and (right) Mutual Information. The red line indicates the threshold defined as significantly more important than the bulk features.



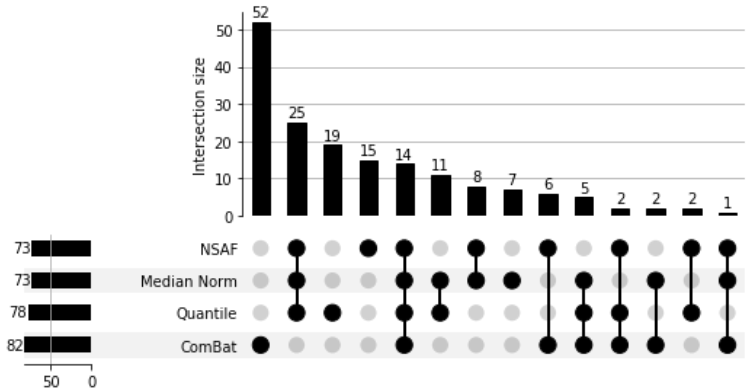
Supplementary figure 10: (left) Overlap of top 100 most important proteins selected by the univariate feature selectors. (right) Histogram showing the percentage of missing values the selected features originally contained.



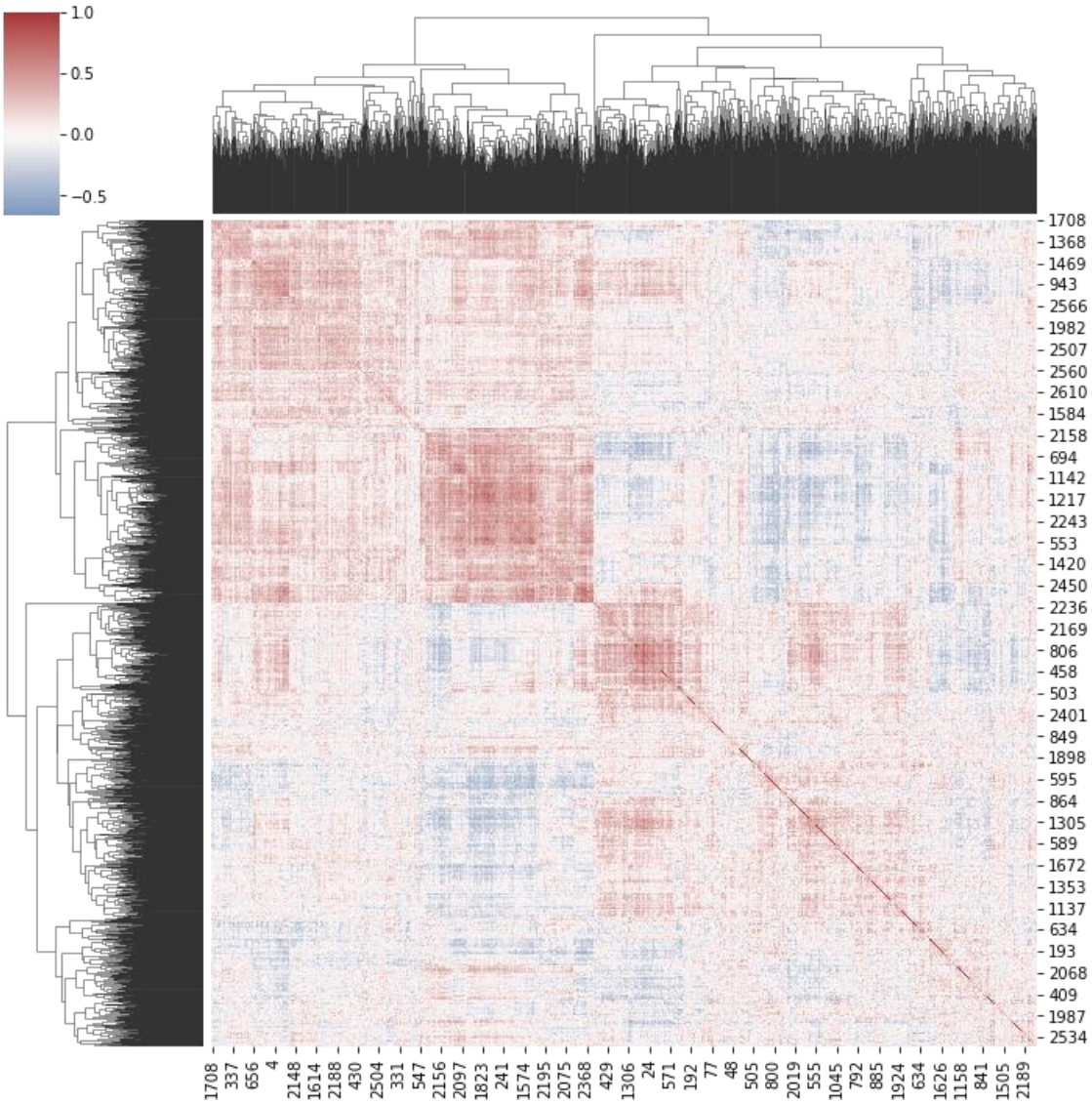
Supplementary figure 11A-B: (A) Overlap of top 100 most important proteins selected by the embedded and wrapper feature selectors. (B) Histogram showing the percentage of missing values the selected features originally contained.



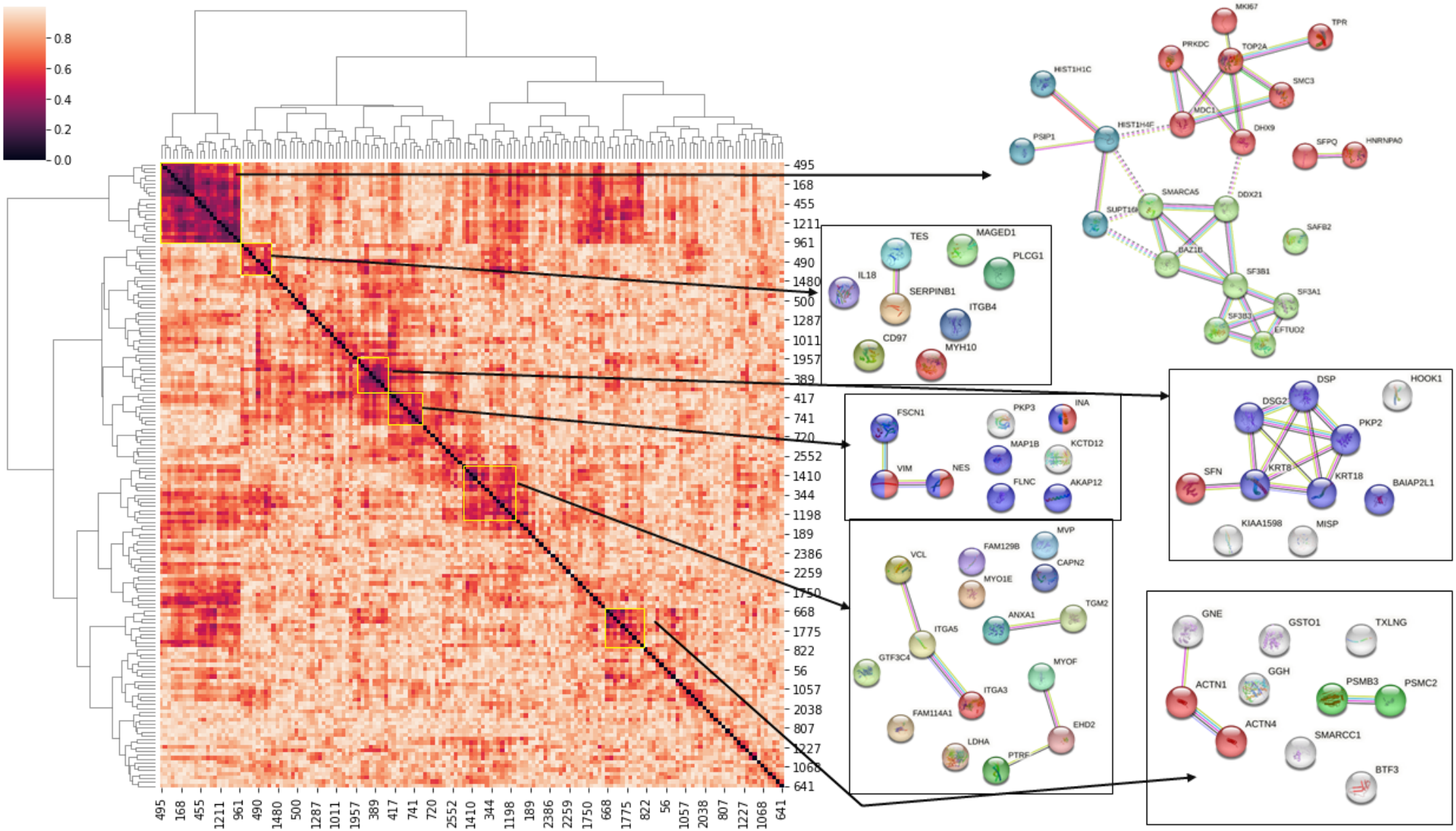
Supplementary figure 12: Top 100 features selected by at least 3 out of five methods were chosen and compared after imputing the quantile dataset with the five different imputation strategies. Numbers on the left side indicate the agreement between feature selectors. Lower numbers indicate lower agreement.



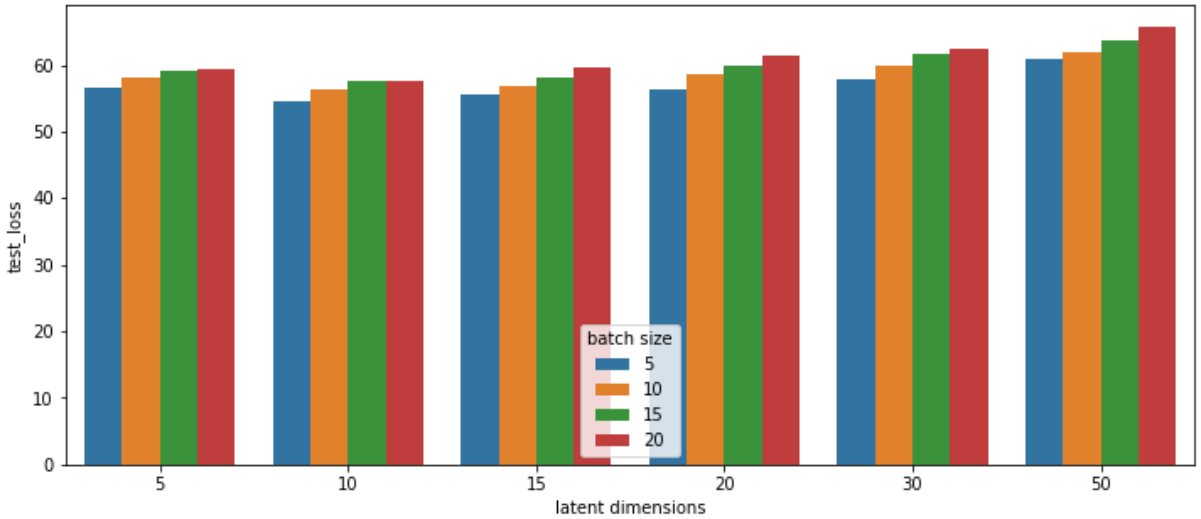
Supplementary figure 13: Top 100 features selected by at least 3 out of five methods were chosen and compared after normalising the dataset with each described technique and imputed with LOD-imputation.



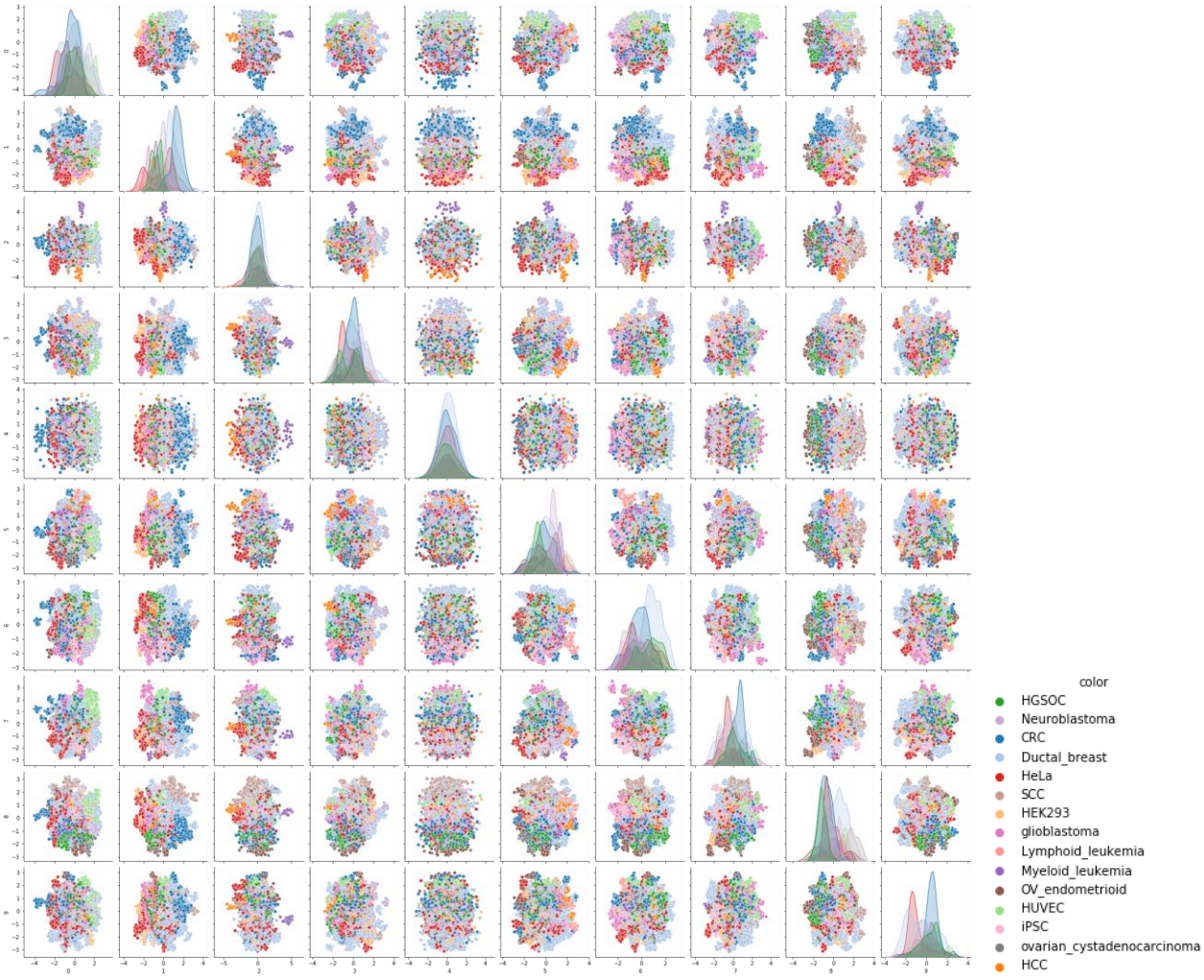
Supplementary figure 14: Cluster heatmap of all pairwise correlation of the quantile dataset.



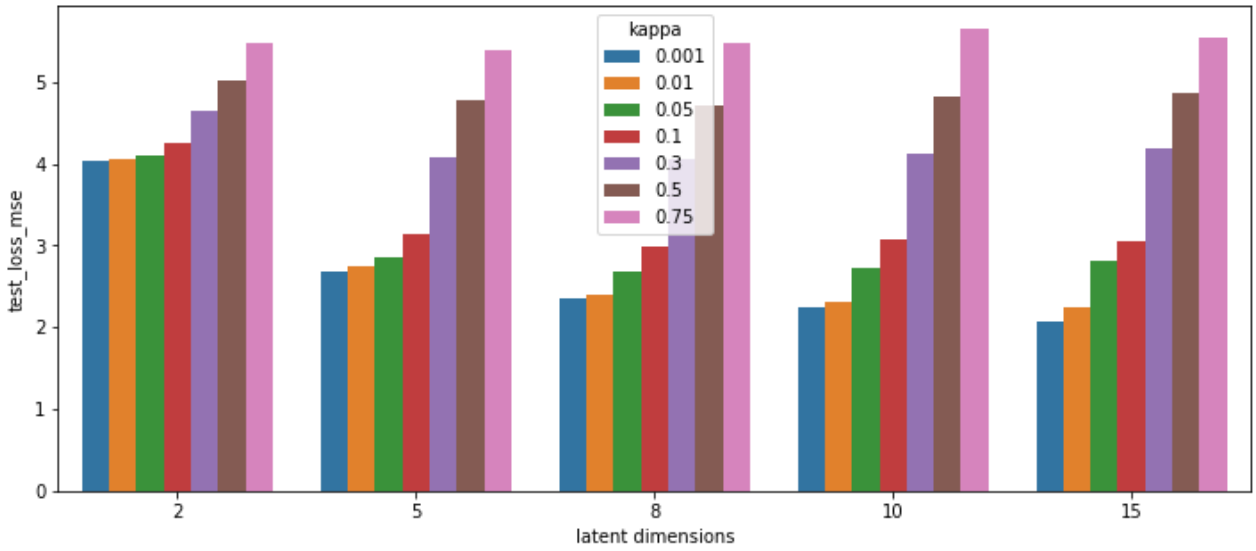
Supplementary figure 15: A cluster heatmap of the pairwise correlations from the feature selected quantile dataset. The yellow boxed clusters were searched in StringDB to discover enriched protein-protein interactions. The String networks are shown next to the the cluster heatmap. Only high confidence interactions (score > 0.7) are shown.



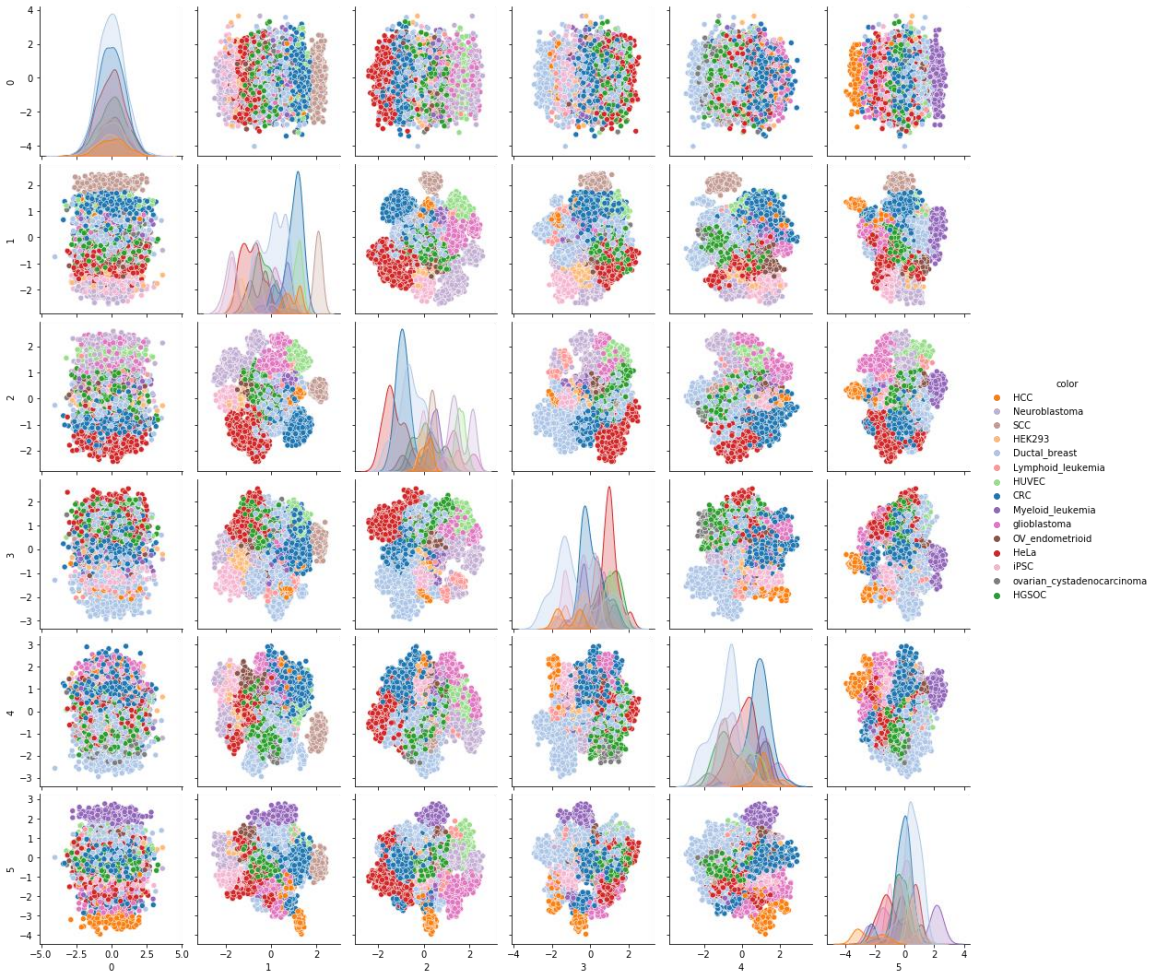
Supplementary figure 16: Loss computed on the test set after 100 epochs of training for VAE1



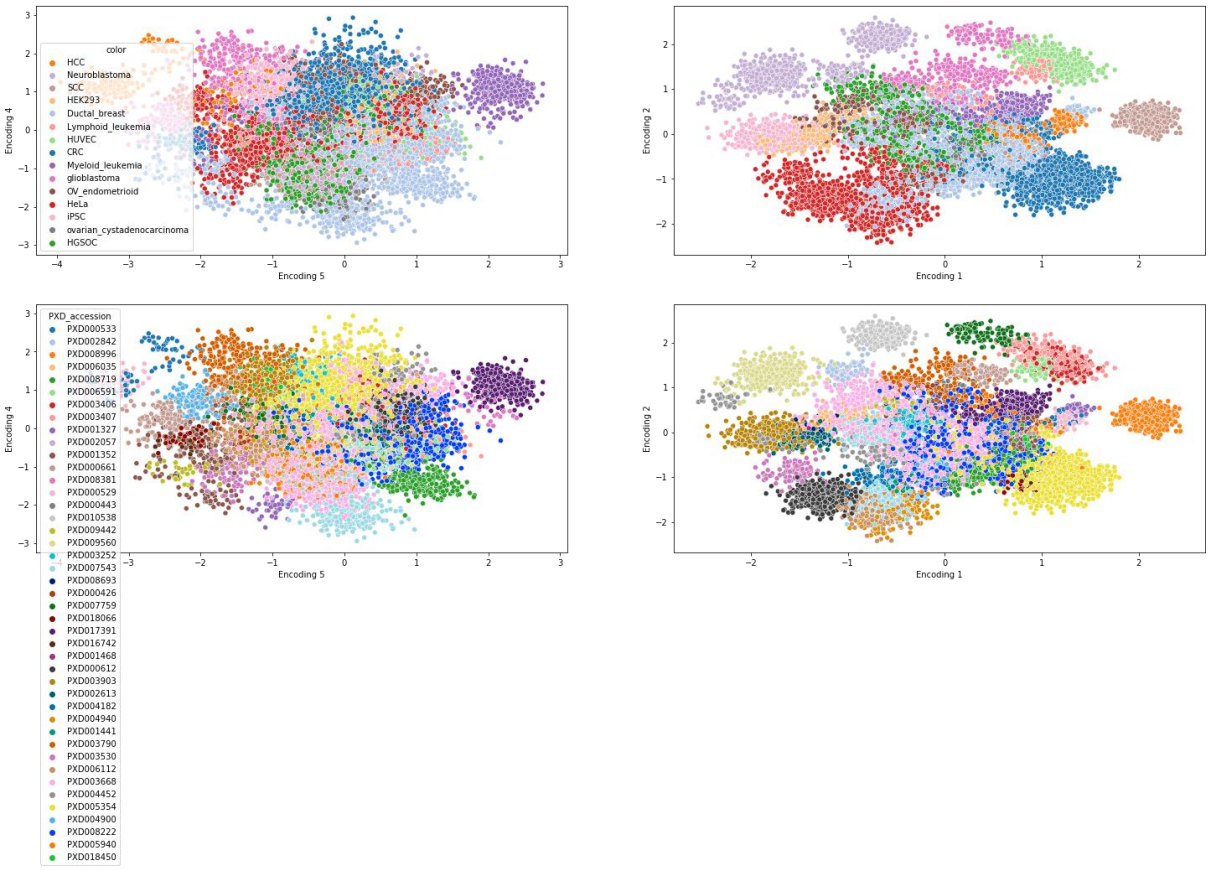
Supplementary figure 17: Pairwise plots of the latent variables from the quantile dataset after 20 resample cycles. Colours indicate the cell line groups.



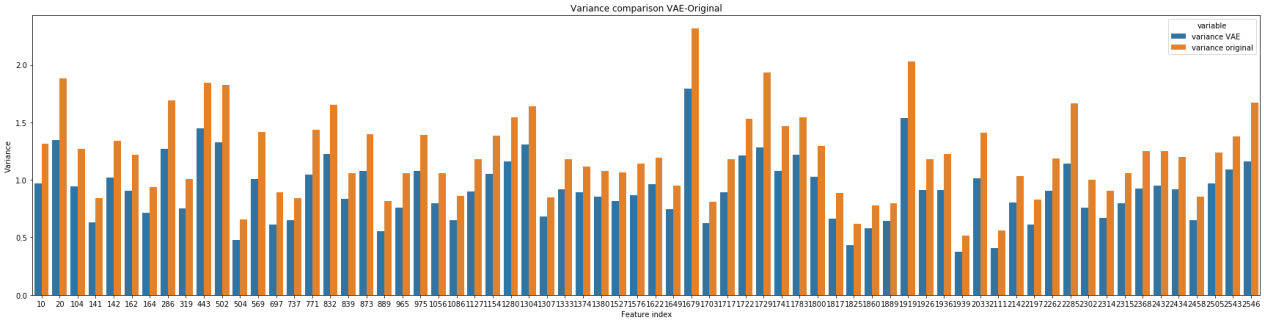
Supplementary figure 18: Reconstruction loss computed on the test set after 100 epochs of training for VAE2 with different settings for kappa.



Supplementary figure 19: Pairwise plots of the latent variables from the feature selected quantile dataset after 20 resample cycles. Colours indicate the cell line groups.



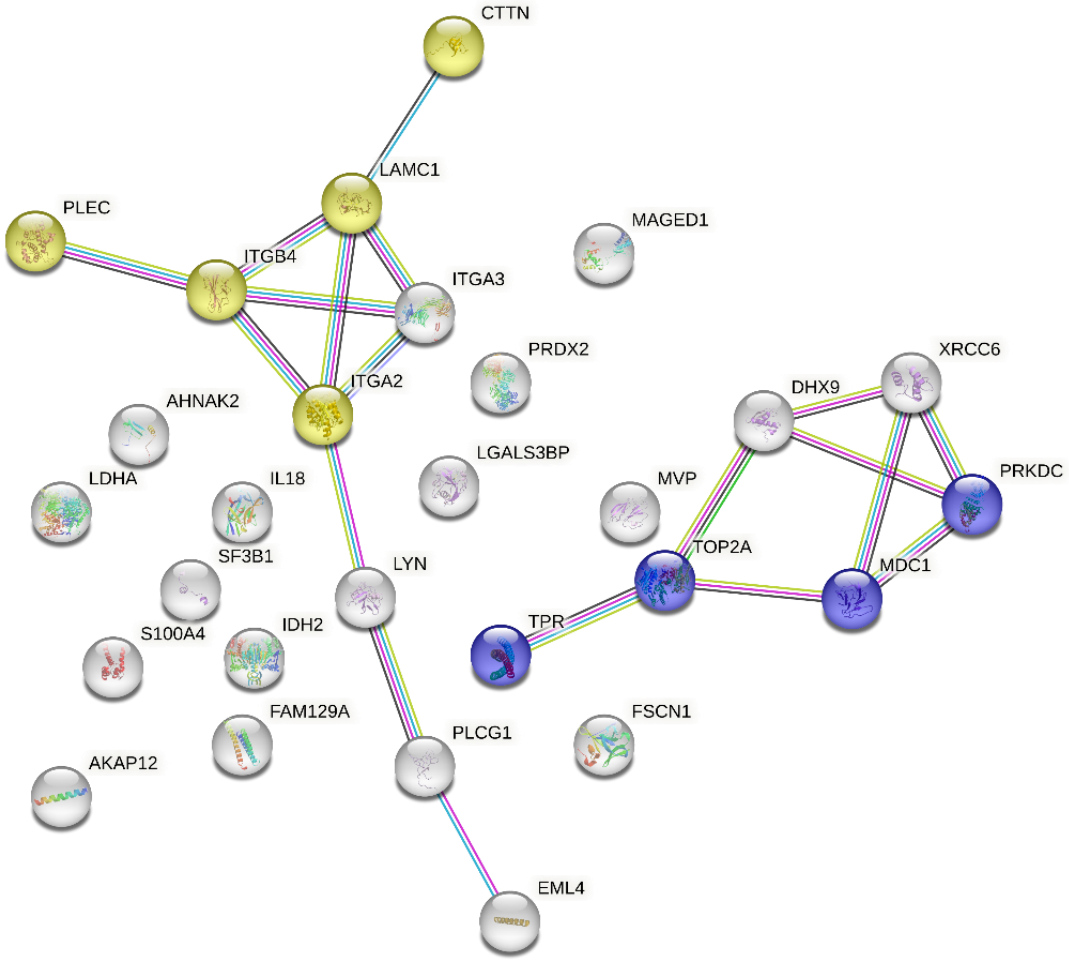
Supplementary figure 20: Scatterplots of the encodings after passing the feature selected quantile-dataset 20 times through VAE2. The upper two plots are coloured by cell line group and the lower two by project.



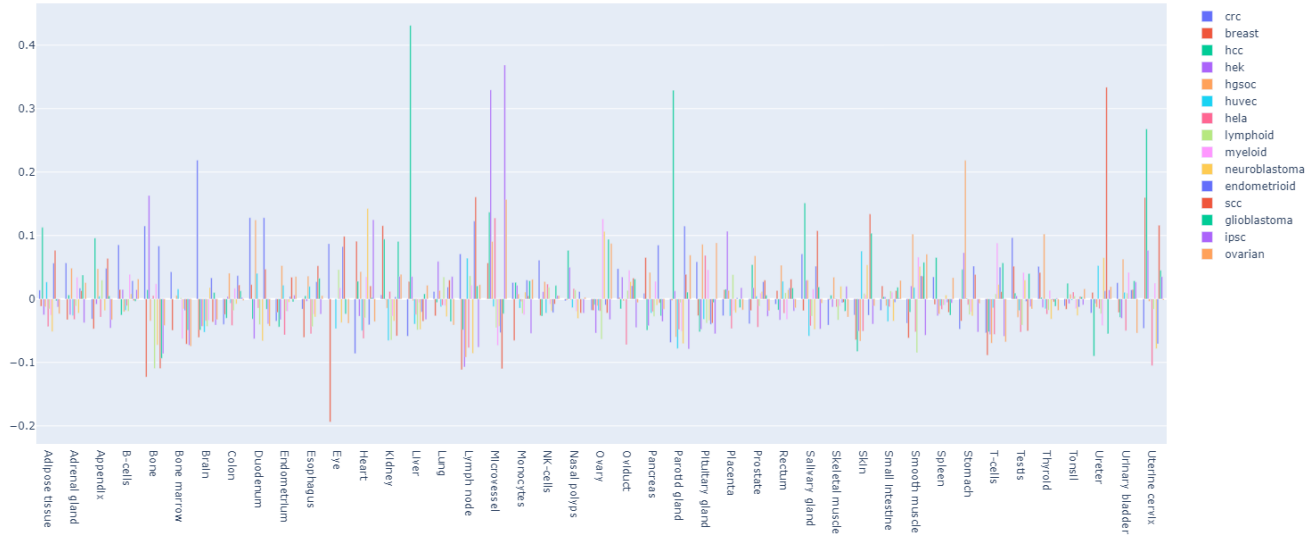
Supplementary figure 21: Bar charts showing the variance of the features that were significantly different between the original and VAE-reconstructed dataset.

Supplementary table 1: True and wrong predictions when performing leave-one-project-out cross validation with the optimised workflow and logistic regression model.

PXD	True labels	Wrongly predicted as
PXD006591	6 lymphoid leukemia	3 myeloid leukemia
PXD001352	3 colorectal cancer 3 myeloid leukemia 3 ductal breast cancer	2 hepatocellular carcinoma 2 hepatocellular carcinoma /
PXD009442	2 ductal breast cancer	2 HGSOC
PXD003252	8 ovarian endometrioid	2 HEK293
PXD003790	12 glioblastoma	1 iPSC, 2 HGSOC, 2 ductal breast cancer



Supplementary figure 22: String interaction network of the most important proteins for classification that are cancer related as defined by THPA.



Supplementary figure 23: Global figure showing the tissue specificity defined by the tissue classifier described in Claeys et al[28] in relation to the cell line group-specific proteins.