

SOFIA VIDEIRA BEGONHA SEQUEIRA MORGADO

Master in Medicine

IDENTIFICATION OF SUB-PHENOTYPES AND PREDICTION OF RESPONSE TO TREATMENT IN MULTIPLE MYELOMA PATIENTS

MASTER IN ANALYSIS AND ENGINEERING OF BIG DATA

NOVA University Lisbon

9, 2023

IDENTIFICATION OF SUB-PHENOTYPES AND PREDICTION OF RESPONSE TO TREATMENT IN MULTIPLE MYELOMA PATIENTS

SOFIA VIDEIRA BEGONHA SEQUEIRA MORGADO

Master in Medicine

Adviser: Cláudia Alexandra Magalhães Soares

Assistant Professor, NOVA University Lisbon

Co-adviser: Ludwig Krippahl

Associate Member Nova Lincs

Examination Committee

Chair: Professor João Pires

Full Professor, FCT-NOVA

Rapporteur: Professor Francisca Leite

Luz Learning Health

Identification of sub-phenotypes and prediction of response to treatment in Multiple Myeloma patients

Copyright © Sofia Videira Begonha Sequeira Morgado, NOVA School of Science and Technology, NOVA University Lisbon.

The NOVA School of Science and Technology and the NOVA University Lisbon have the right, perpetual and without geographical boundaries, to file and publish this dissertation through printed copies reproduced on paper or on digital form, or by any other means known or that may be invented, and to disseminate through scientific repositories and admit its copying and distribution for non-commercial, educational or research purposes, as long as credit is given to the author and editor.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to the following individuals for their invaluable contributions and support throughout the course of this project.

Professor Ludwig deserves special mention for his unwavering guidance and profound expertise. I am deeply thankful to Professor Cláudia for her invaluable insights and contributions, which significantly enriched the quality of this work.

My sincere appreciation goes to Flavio and Ahmed for their generous investment of time, persistent assistance, and invaluable guidance, which were instrumental in the successful completion of this thesis. I extend my gratitude to Janssen, The Pharmaceutical Companies of Johnson Johnson, for their provision of resources and availability.

Lastly, I would like to dedicate a special note of thanks to my brother Edu, for the inspiration, guidance, support, and debugging.

ABSTRACT

Multiple myeloma is an heterogeneous hemathological cancer that affects plasma cells in the bone marrow. Despite recent advances in the treatment of multiple myeloma, the outcome of the patients remains variable. Thus, it is essential to better understand this disease. Identification of sub-phenotypes in multiple myeloma patients and the prediction of response to treatment is crucial for improved individualized patient care.

The primary objctive of this project involved the analysis of a dataset provided by Janssen, The Pharmaceutical Companies of Johnson and Johnson Belgium. The aim was to assess the feasibility of addressing the challenges proposed by the company, which included identifying sub-phenotypes of patients and predicting their response to treatment. Additionally, it was important to ensure that the resulting models were interpretable.

Based on the conducted analysis, it became apparent that the dataset posed several challenges, notably the absence of crucial information. The primary reason for this can be attributed to the significant presence of missing values within the provided features, as well as the lack of inclusion of other crucial features.

This dissertation addresses these stringent restrictions and contains valuable information and discussion on procedures that can be applied to other data sets with comparable limitations. Using an autoencoder novel network architechture, a procedure is defined for dealing with a large number of missing values without resorting to imputation. This new representation of the data may be used for clustering in an attempt to identify patient subphenotypes. Due to the requirement of creating interpretable classifiers, it is not possible to train classifiers using these encoded features. Therefore, a missing value pattern aggregation strategy is recommended. In addition, the importance and implementation of methods that ensure model interpretability are discussed.

Keywords: Multiple Myeloma, Clustering, Classification, Missing values, Model explainability

RESUMO

O mieloma múltiplo é um cancro hematológico que afeta as células do plasma na medula óssea. Apesar dos avanços recentes na terapêutica, a progressão clínica destes pacientes é ainda muito variável. Por esse motivo, é essencial criar uma maior compreensão destes doentes, modo a promover uma melhor abordagem clínica.

Os objetivos deste projeto incluíram analisar o dataset fornecido pela Janssen, The Pharmaceutical Companies of Johnson and Johnson Belgium, para determinar se seria possível responder aos desafios propostos por esta empresa farmacêutica: investigar a presença de sub-fenótipos de doentes, bem como prever a resposta à terapêutica, garantindo a interpretabilidade dos modelos obtidos.

Perante esta análise, tornou-se evidente que o dataset apresentava desafios significativos, incluindo a falta de informação essencial devido à presença de um grande número de valores em falta e à ausência de variáveis potencialmente relevantes.

Assim, esta dissertação inclui uma discussão sobre os procedimentos que possam ser aplicados a dados semelhantes aos nossos, tendo em vista os objetivos importantes que definimos. Foi utilizado um autoencoder para criar uma representação dos dados sem valores em falta e que não seja dependente de imputação. Esta nova representação foi então utilizada para criar clusters na tentativa de identificar sub-fenótipos de doentes. Dado que pretendemos criar classificadores que sejam interpretáveis, não foi possível utilizar estas variáveis codificadas. Assim, definimos uma metodologia baseada nos padrões de valores em falta dos doentes. Por fim, métodos para a obtenção de modelos interpretáveis foram discutidos.

Palavras-chave: Mieloma múltiplo, Clustering, Classificação, Valores em falta, Interpretação de modelos

CONTENTS

List of Figures	ix
List of Tables	xii
1 Introduction	1
1.1 Multiple Myeloma	1
1.1.1 Staging Systems Evolution	2
1.1.2 Current Treatment	3
1.2 Objectives	4
2 Contextualization	6
2.1 State of the art in Multiple Myeloma	6
2.2 Data Preprocessing	9
2.2.1 Exploratory Data Analysis	9
2.2.2 Normalization	10
2.2.3 Missing values	11
2.2.4 Class imbalance	12
2.3 Survival Analysis	13
2.3.1 Censored data	13
2.3.2 The survivor $S(t)$ and the hazard $h(t)$ functions	13
2.3.3 The average survival, the average hazard rate and the median survival	14
2.3.4 Kaplan-Meier	14
2.4 Localized Learning	15
2.4.1 Implementation of Localized learning algorithms	16
2.5 Dimensionality Reduction	20
2.5.1 Principal Component Analysis	20
2.5.2 Self-Organizing Maps	22
2.5.3 Isomap	23
2.5.4 Undercomplete autoencoders	24

2.6	Cluster Analysis	27
2.6.1	Hierarchical Clustering	27
2.6.2	Density-based methods	28
2.6.3	Evaluating clustering results	28
2.7	Classification	30
2.7.1	Logistic Regression	30
2.7.2	Support Vector Machines	30
2.7.3	Ensemble methods	36
2.8	Interpretability	37
2.8.1	Permutation	37
2.8.2	Shap Values	38
2.8.3	Partial Dependence Plots and Individual Conditional Expectation	38
2.9	Limitations	40
3	Exploratory Data Analysis and Preprocessing	41
3.1	General Description	41
3.1.1	Lines of treatment	42
3.1.2	Data collected at diagnosis	43
3.2	Variables Description and Analysis	43
3.2.1	Demographic Data	43
3.2.2	Medical Exams	44
3.2.3	Staging Systems	47
3.2.4	Treatment	48
3.2.5	Patient Care Time Intervals	50
3.2.6	Treatment Response	51
3.2.7	Survival outcome	54
3.2.8	Final variables	56
3.2.9	Missing values	57
3.3	First Line of Treatment	59
3.3.1	Relations between diagnosis and first line of treatment measurements	60
3.3.2	Response to the first line of treatment	61
3.3.3	Missing Values	61
3.4	Outliers	61
3.5	Normalization	63
4	Methods, Results and Discussion	64
4.1	The Challenge of the Missing Values	64
4.1.1	Imputation	64
4.1.2	Removal of Uninformative Instances	65
4.1.3	Autoencoder	66

4.1.4	Encoding All Data	68
4.2	Defining Sub-phenotypes of Patients	72
4.2.1	Relationship Between Sub-phenotypes and the Treatment	74
4.3	Classification	76
4.3.1	Patterns of Missing Values	76
4.3.2	Classification	78
4.3.3	Global Approach	80
4.3.4	Localized Approach	81
4.4	Model Interpretation and Feature Importance	82
4.4.1	Feature Ranking by Feature Permutation	82
4.4.2	The Shap Values	82
4.4.3	Partial Dependence Plots and Individual Conditional Expectation	85
4.4.4	Global and Localized Learning	86
4.5	Limitations	87
5	Recommended procedures	89
5.1	Preprocessing	89
5.2	Autoencoder to solve the missing values limitation	91
5.3	Sub-phenotypes	92
5.4	Prediction of response to treatment	93
5.5	Model interpretation and feature importance	94
6	Conclusions	95
	Bibliography	96
	Appendices	
A	Appendix 1	104
B	Appendix 2	106
C	Appendix 3	107
D	Appendix 4	111
E	Appendix 5	114
F	Appendix 6	116

LIST OF FIGURES

2.1	KNN Imputation	11
2.2	Localized Learning Approach	16
2.3	Multiple prototypes localized learning approaches	16
2.4	Multiple prototypes localized learning approaches	17
2.5	Smooth and Square Kernels	17
2.6	Localized SVM	18
2.7	Profile SVM	19
2.8	The first principal component direction	21
2.9	PCA algorithm	22
2.10	The "Mexican Hat"function of lateral connections from [45]	23
2.11	Geodesic distance [4].	24
2.12	Autoencoder Architecture	24
2.13	Activation functions for neural networks	25
2.14	Sigmoid function	30
2.15	SVM on linearly separable classes	31
2.16	SVM on non-linearly separable classes	33
2.17	SVM and the Kernel Trick	34
2.18	Shap values	38
3.1	Removal of variables	41
3.2	Patients per treatment line	42
3.3	Patients per treatment line	43
3.4	Age at diagnosis	43
3.5	Age at diagnosis per treatment line	44
3.6	LDH levels at diagnosis	44
3.7	LDH levels per treatment line	45
3.8	LDH levels at diagnosis and treatment line	45
3.9	Stage at diagnosis	47
3.10	Staging System Comparison at diagnosis	47
3.11	Staging System Comparison at treatment line	48

3.12	ASCT per treatment line	50
3.13	ASCT per age at diagnosis	50
3.14	Patient care time intervals	51
3.15	Treatment line duration per treatment line number	51
3.16	Time to next treatment per treatment line number	52
3.17	Initial and last response to treatment	52
3.18	Initial and best response to treatment	53
3.19	Last and best response to treatment	53
3.20	Binary response to treatment	54
3.21	Kaplan-Meier curve	55
3.22	Deaths per treatment line	55
3.23	Survival Analysis Results for del17p	56
3.24	Survival Analysis Results for t(4;14)	56
3.25	Survival Analysis Results for del13q	57
3.26	Survival per age at diagnosis	57
3.27	Survival per stage at diagnosis	58
3.28	LDH Correlation	60
3.29	Staging Systems Correlations	61
3.30	Survival per response to first line of treatment	62
3.31	Min-Max Normalization	63
4.1	Autoencoder schema	67
4.2	Autoencoder representation with treatment values	68
4.3	Autoencoder representation with stages values	69
4.4	Autoencoder representation with mutation values	69
4.5	DBSCAN results on autoencoder representation of the data	71
4.6	Hierarchical clustering results on autoencoder representation of the data	72
4.7	AE clinical clusters	73
4.8	Stages shown in the autoencoder representation	74
4.9	Mutations shown in the autoencoder representation	74
4.10	AE clinical clusters	75
4.11	AE clinical vs treatment schema	75
4.12	Selection of Missing Values Patterns	77
4.13	Methodology by group	77
4.14	Nested Cross Validation Schema	79
4.15	Precision KDE	80
4.16	Feature Ranking with Feature Permutation	83
4.17	Feature Ranking with Shap Values	83
4.18	Shap Values for one particular instance	84
4.19	Shap value per value of age at diagnosis	84
4.20	Shap value per value of auto-transplant	85

4.21	Shap Interaction Plot	85
4.22	Feature Ranking with Partial Dependence Plots	86
4.23	PdP and ICE	86
5.1	Recommendations for data preprocessing 1	90
5.2	Recommendations for data preprocessing 2	90
5.3	Recommendations for data preprocessing 3	92
5.4	Autoencoder schema	92
5.5	Autoencoder schema	93
A.1	Mutations at diagnosis and treatment line	104
A.2	Mutations at diagnosis	104
A.3	Mutations at treatment line	105
C.1	Feature contribution to each principal component	107
C.2	PCA Scree Plot	108
C.3	PCA Reconstruction Error	108
C.4	PCA Results	109
C.5	t-SNE Results	110
C.6	Isomap Results	110
E.1	Autoencoder loss function	114
E.2	Nearest neighbors distances for DBSCAN	115
E.3	Dendrogram for Hierarchical Clustering	115
F.1	Shap Values for subphenotype 1	116
F.2	Shap Values for subphenotype 2	116

LIST OF TABLES

1.1	Multiple Myeloma Risk Factors	2
1.2	Multiple Myeloma Staging Systems	2
1.3	Multiple Myeloma Treatment Agents	4
3.1	Presence of Genetic Mutations	47
3.2	Drug classes	48
3.3	Comparison of the survival per age at diagnosis	58
3.4	Variables	59
3.5	Missing values	59
3.6	Correlations in first line of treatment	60
3.7	Missing values	62
3.8	Outliers	62
4.1	Sub-phenotype Comparison 1	73
4.2	Sub-phenotype Comparison 2	74
4.3	Patterns of Missing Values	77
4.4	Incoherent instances	78
4.5	Confusion matrix for the best logistic regression model. PN-predicted negatives, PP-predicted positives, RN-real negatives, RP-real positives	81
4.6	Confusion matrix for the best logistic regression model. PN-predicted negatives, PP-predicted positives, RN-real negatives, RP-real positives	81
B.1	Treatment Responses	106

INTRODUCTION

Multiple myeloma is an heterogeneous hematological cancer that affects plasma cells in the bone marrow. It is considered heterogeneous as the manifestations of the disease vary across patients. Despite recent advances in the treatment of multiple myeloma, the outcome of the patients remains variable. Thus, it is essential to better understand the biology of the disease and its sub-phenotypes to improve patient care. Identification of sub-phenotypes in multiple myeloma patients and the prediction of response to treatment is crucial for individualized patient care and to ensure the most effective treatment for each patient.

In this study, I explore the identification of sub-phenotypes and prediction of response to treatment in multiple myeloma patients. Janssen Belgium has provided me with a dataset to carry out this relevant research. Both a statistical analysis and machine learning algorithms were applied to uncover new insights into the underlying phenomena of this heterogeneous disease and to enhance our understanding of how to best treat these patients.

This chapter will provide a brief summary of multiple myeloma, covering the current categorization staging systems and the most frequent therapies. Furthermore, a section explaining the objectives of this project is included.

1.1 Multiple Myeloma

Multiple myeloma (MM) is the second most common hematological cancer and it is originated in a single clone of plasma cells, located in the bone marrow [40] [20].

This disease has a complex pathogenesis and enormous heterogeneity. The cause of MM is unknown and no single common molecular pathogenic pathway has been identified [3] [74]. However, a few risk factors have been established, including certain occupational exposures [40].

Multiple genetic anomalies have been identified in these cancerous cells. One of the most common genetic abnormalities in MM is the gain of a long arm of chromosome 1 [74]. Other abnormalities that must be searched for when MM is diagnosed include deletion

Table 1.1: Multiple Myeloma Risk Factors [40] [12]

Risk factors
Ionizing radiation
Petroleum products
Pesticides
Organic solvents

of the short arm of chromosome 17 (del17) and translocation between chromosome 14 and chromosome 4(t4-14), 11 or 16 [20]. MM patients with the high-risk cytogenetic abnormalities, such as deletion of part of the chromosome 13 or 17 and translocation between chromosomes 4 and 14, have shown to have a poorer prognosis [63] [26].

Among others, clinical signs of end-organ damage include bone pain and malignant fractures, hypercalcemia, anemia and renal failure [40].

It is considered an incurable disease, although therapy can extend patient survival and improve their quality of life [40] [68] [55]. Currently, the median survival of multiple myeloma patients is 8 years. The major causes of death include progressive myeloma, renal failure, sepsis and treatment related comorbidities [40].

1.1.1 Staging Systems Evolution

1.1.1.1 Staging Systems

The first MM staging system was proposed in 1975 by Salmon and Durie and divided the patients into three categories [23]. It was followed by the International Staging System ISS, which includes two simple laboratory parameters to also define three stages, being stage I the one with lower risk [32]. The criteria for these two staging systems is listed in Table 1.1.1.1.

Table 1.2: Multiple Myeloma Staging Systems Criteria [23] [27] [32]

Stage	Durie-Salmon Criteria	ISS Criteria
I	Hemoglobin value $>10\text{g/dL}$ Calcium level $<10.5\text{mg/dL}$ Low M-component (IgG $<5\text{g/dL}$; IgA $<3\text{g/dL}$) Urine light chain M-component $<4\text{g/24h}$ Normal X-ray or solitary bone plasmacytoma	$\beta 2$ -microglobulin $<3.5\text{mg/L}$ Albumin $\geq 3.5\text{g/dL}$
II	Not I nor III	Not I nor III
III	Hemoglobin level is $<8.5\text{g/dL}$ OR Calcium value is $>12\text{mg/dL}$ OR High M-component (IgG $>7\text{g/dL}$; IgA $>5\text{g/dL}$) OR Urine light chain M-component $>12\text{g/24h}$ OR Advanced lytic bone lesion	$\beta 2$ -microglobulin is $\geq 5.5\text{mg/L}$

The Revised-ISS added cytogenetic aberrations (the presence or absence of some already identified high risk mutations) and LDH (Lactate Dehydrogenase) levels as risk determining parameters [62].

These systems demonstrated their value in identifying a small minority of patients at high risk. Nevertheless, they have significant limitations, including limited accuracy in predicting patient survival, namely when considering intermediate-risk patients, and

insufficient capacity to discriminate between risk groups [3] [60] [61] [30]. Moreover, most high-risk patients, those with sub-optimal response to therapy or early relapse, seldom carry the identified high risk characteristics [76].

Additionally, the cytogenetic abnormalities chosen for the R-ISS may not have been the best to include in this staging system [60].

1.1.1.2 Alternative Staging Systems

Alternative risk scores have already been developed, combining several clinical and molecular markers. Multiple strategies have already surpassed the classic risk scores by better estimating patient outcome, as is exemplified below [3] [68].

Firstly, the combination of ISS or R-ISS and unsupervised machine learning clustering offers excellent prospects for risk stratification improvement in MM [60].

One other strategy included administering highly effective drug combinations to patients with high-risk factors (elevated blood levels of plasma cells or high risk cytogenetics). Alternatively, adapting the treatment intensity and duration to the sustained negative minimal residual disease has been explored [61].

On the one hand, it is thought that therapy response variability is largely due to chemoresistance linked to changes in gene expression. Interpreting these gene expression changes is thus essential to assessing the potential efficacy of antimyeloma therapies in the future [3].

Gene expression profiling may help identify high-risk disease and improve treatment sensitivity prediction, thus helping to choose better treatment schemes [68] [74]. Several distinct genetic profiles have previously been established and are gaining prominence in risk stratification [48]. Yet, they are still very heterogeneous and unhelpful to optimize therapy. Thus, most have been restricted to clinical trials [61].

One AI-supported Modified Risk Staging, which included six, simple to obtain, clinical parameters, has shown promising results as it outperformed the ISS in overall survival and was equivalent in determining disease progression-free survival [25].

Finally, the IAC-50 model, which includes ISS stage, 46-gene expression and first-line therapy data, has shown to be useful in predicting MM survival and which of the treatment combinations maximizes each patient long-term survival [61].

These alternative stages were not included in the data used for this project, possibly due to the fact that they are not as generally used as the classic ones.

1.1.2 Current Treatment

Patients with symptomatic or progressive multiple myeloma require systemic therapy to control the myeloma and supportive care to control the symptoms and complications.

The therapy is determined by the age and comorbidities of the patient. Some patients may be ineligible for high-dose treatment and transplantation [40].

An induction treatment phase is followed by high-dose treatment with autologous stem cell transplantation and consolidation in the patients that are transplant candidates. All patients undergo maintenance treatment in order to prolong remissions. On further progression, relapsed disease treatment must be administered [40].

The variety of therapeutic alternatives and the wide inter individual variance in response to therapy makes good patient care challenging [55][65].

Table 1.1.2 shows a list of possible treatment agents for multiple myeloma. Various drug combinations may be given to the patients. The combination of lenalidomide, bortezomib, and dexamethasone is one of the preferred induction regimens since it yields near to a 100% response rate [40].

Table 1.3: Multiple Myeloma Treatment Agents. Adapted from [40]

Class	Agent	Indications
Immunomodulatory agents	Thalidomide	New Diagnosis/Relapse
	Lenalidomide	Relapse
	Pomalidomide	Relapse
Proteasome Inhibitor	Bortezomib	New Diagnosis/Relapse
	Carfilzomib	Relapse
	Isazomib	Relapse
Antibodies	Daratumumab	Relapse
	Elotuzumab	Relapse
Histone deacetylase inhibitor	Panobinostat	Relapse
Alkylating agents	Melphalan	New Diagnosis/Relapse
	Cyclophosphamide	New Diagnosis/Relapse
	Bendamustine	Relapse
Steroid	Dexamethasone	All stages
	Prednisone	All Stages

1.2 Objectives

The primary purpose of this work was the analysis of a dataset provided by Janssen, The Pharmaceutical Companies of Johnson and Johnson Belgium, in order to extract the maximum amount of information from it, as well as to compare and discuss the methods utilized. Our objective included determining the quality and usefulness of the data provided for addressing the challenges proposed by the company, which included identifying sub-phenotypes of patients and predicting their response to treatment, while ensuring interpretable results.

First, the data was examined to determine and address its inherent limitations.

Specifically, to create a profile of the multiple myeloma patients present in the data, we clustered patients using different methods, such as hierarchical clustering or density based methods. The clustering methods were compared in order to choose the best clustering method for this context. The goal of this step was also to try to find natural groups in the data and understand the major characteristics of each cluster.

Furthermore, supervised learning techniques were used in order to try to create a model that predicts the response to treatment based on patients' features. Again, we aim

to compare the methods used and discuss their advantages and disadvantages, as well as evaluating their interpretability.

Due to the characteristics of the data, the primary purpose of this report is to specify and evaluate the multiple methodologies applied, rather than the results themselves.

CONTEXTUALIZATION

2.1 State of the art in Multiple Myeloma

Prognostic and predictive biomarker models that reliably anticipate patients' care trajectories and drug response are becoming increasingly essential in both clinical practice and academic research, and specifically in MM [76][65] [80]. Additionally, massive amounts of high-dimensional data has been produced on MM, including genetic, serology, histology and radiology data [3].

Genetic research is one prominent area. It is crucial to identify variations in gene expression that lead to observed individual variations in response to therapy, and incorporate it with non-omics data to create predictive models [55][65]. Numerous computerized methods for MM immunophenotypic research have already been used [3].

A brief review of the state-of-the-art regarding analyzing data about multiple myeloma, as well as similar data, such as clinical data on other cancer types, will be provided in this chapter. This includes data pre-processing, including dealing with correlated variables and missing values, dimensionality reduction, as well as clustering and classification algorithms.

Data Preprocessing When referring to data pre-processing and statistical analysis, several methodologies have demonstrated to be useful in MM studies. Multiple variables have shown an independent correlation with survival, making univariate analysis advantageous [68][30]. These variables include age, performance status, C-reactive protein, lymphocyte count and ISS stage [71].

Furthermore, removal of highly correlated features based on their correlation coefficient may be useful [76].

In the creation of prognostic models, missing data is a frequent challenge [71]. There are multiple ways to deal with it. The easiest solution is to remove observations or variables with missing data, however this results in a significant loss of information in the medical field [1].

When the number of missing values is low, a simple imputation of the missing values

with the median value of the parameter was used in clinical data of MM patients [25]. Multiple imputation has the potential to enhance the reliability of medical research. Three methods of multiple imputation by chained equations were compared in generated data on breast cancer and melanoma. Polytomous regression yielded the best results, while using random forest had the largest variance, the most biased results and was very sensitive to noise. Predictive mean matching was nearly as accurate as the former, and had the advantage of a shorter processing time [24]. Other methodologies will be discussed in the next chapter.

For class imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was used in a dataset of MM patients [76].

Dimensionality reduction Principal Component Analysis (PCA) was performed as a part of creating a staging system to predict the risk of relapse in multiple myeloma patients undergoing autologous stem cell transplantation. It was applied to clinical and laboratory data and the number of principal components selected was chosen in order to preserve 95% of the variance in the data, followed by clustering analysis [30].

Other examples in the oncology field include principal component analysis usage in breast cancer patients to project individual gene sets onto lower dimensional space, and the first principal component median was used as threshold for high risk patients [88]. A self-organizing map applied to clinico-pathological features of breast cancer patients was used in order to project the data into a low-dimensional space to allow a better subsequent performance of a clustering method [75].

Clustering and classification Clustering algorithms, Random Forests, Gradient Boosting Decision Trees, SVM, and Naive Bayes are a few of the ML approaches that have previously been used for the computational diagnosis of MM [3]. They will be mentioned more broadly below.

Unsupervised machine learning techniques enable the transformation of multidimensional data into simplified and reproducible clusters, assisting researchers in maximizing the value of data and reducing costly healthcare resources [60] [41]. Nevertheless, clustering medical data faces the challenge of dealing with a lot of disease markers available [41]. As explained above, in many cases dimensionality reduction is applied to pre-process the data. Some examples of unsupervised techniques used include:

Using a K-adaptive partitioning algorithm and a rule-based supervised decision-tree classifier, a novel inexpensive staging system based on age, albumin, β 2-microglobulin, calcium, glomerular filtration rate, and hemoglobin has been developed. The K-adaptive partitioning technique was utilized to establish new thresholds for the parameters, while the decision-tree was utilized to develop risk staging criteria. This classifier outperformed the ISS for overall survival prediction and was comparable in predicting progression free survival [25].

Also, univariate K-means has been used to cluster patients in order to generate cut-off values for numerical variables based on the limit values for each cluster [30].

DBSCAN was used to create clusters of patients based on their clinico-pathological features [75].

A Gaussian Mixture Modeling fitter with an Expectation Maximization algorithm (GMM-EM model implemented in the “Mclust” algorithm version 5.4.7) was used to create clusters with different survivals. It shows to be particularly useful in the R-ISS intermediate-risk patients. Time-dependent AUCs revealed that the precision in survival prediction of the unsupervised clusterization was clearly superior to ISS and R-ISS [60].

After applying principal component analysis to the features, spectral clustering was used to obtain two groups of patients and their distinct survival patterns were studied. In order to make this clustering useful for clinical applications, as it is simple to be memorized, a Fast and Frugal Tree (FFT) was fitted using the two clusters obtained as the labels for the patients. The FFT showed competitive results comparing to the random forest. [30].

Random forest and random survival forests were applied to identify high risk patients (patients with suboptimal response to induction or early relapse), to predict the response to proteasome inhibitor treatment and to predict survival outcomes in MM [55][76][64][61].

Classification models, including ensemble models such as gradient boosting and random forests, SVM and artificial neural networks have been used breast cancer data after clustering using K-means [87].

Furthermore, transfer learning approaches have been applied to enable higher performance results from machine learning algorithms such as transfer SVM and transfer Adaboost [80].

Finally, convolutional neural networks and other DL methods were utilized to detect MM bone lesions in CT, MRI, and PET scans [3].

Survival analysis For survival analysis, the correlation between each variable and overall survival in MM patients was examined using univariate cox-regression. Also, time-dependent AUCs and c-indexes were used [60].

In the next sections, we will describe the machine learning methods that will be applied in this project. First, we will mention the exploratory data analysis and data preprocessing methods used in order to address the major problems we found in our data: the presence of missing values, class imbalance and highly correlated features. It will be followed by methods of feature analysis and dimensionality reduction. The localized learning approach will be discussed. Different methods of cluster analysis will be described and compared, followed by the description of classification methods. Finally, a brief explanation of survival analysis will be presented.

2.2 Data Preprocessing

2.2.1 Exploratory Data Analysis

Single feature evaluation First, each variable must be independently assessed. Its distribution may be examined: a histogram can be used for visualization, or some simple features can be presented, such as the central tendency, the mode (the number of clusters), the spread (variability), the tails (how quickly the probabilities drop off towards the tails of the distribution), and the existence of outliers [22].

It is essential to identify the presence of missing data and to establish if there is a pattern associated with them. Imputation is a method for dealing with missing values in which these values are determined and included in the data [25].

It may be also useful to compare the values from one variable in different subgroups of data. The Kruskal-Wallis is a non-parametric statistical test that is used to determine if there is a significant difference in the median of more than two groups, and it is a generalization of the Mann-Whitney U test, that is used to compare pairs of groups. It is useful when the assumptions of normality and equal variances are not met and, thus, the ANOVA or the T-test cannot be used, as is this case.

Multiple feature evaluation There are several ways of evaluating the relationship between different features, which may be useful for further feature understanding and selection.

For the purpose of evaluating feature correlation, we may begin to employ pairwise feature plots for visualization. A scatter plot is the easiest way to examine the relationship between two variables. Introducing random noise into the data -jittering- reverses the impact of the rounded measurements that may be present in the data. (The hexbin plot is an alternative to scatter plot, usefull for very large datasets, but it makes the outliers invisible). Additionally, the relationships between variables may also be characterized by a percentiles plot [22].

Then, some correlation metrics, that measure the strength of the relationship between two variables, may be employed. In order to compare variables that are not expressed in the same units, we may transform each value to a standard score and use the Pearson product-moment correlation coefficient or transform each value to its rank and use the Spearman rank correlation coefficient [22].

The Pearson correlation is an easy to compute linear correlation measurement that relies on the covariance, which measures the tendency for two variables to vary together. It is expressed in Equation 2.1:

$$\rho = \frac{Cov(X, Y)}{S_X S_Y} \quad (2.1)$$

where $Cov(X, Y)$ is the covariance between the two variables and S_X and S_Y are the standard deviations of each variable.

One of its limitations is the fact that it only measures linear relationships. For that reason, in order to use this method one should always evaluate the features scatter plot. Also, it is not robust in the presence of outliers or in variables whose distribution is not roughly normal [22].

Mutual information (MI), on the other hand, is a quantitative measure of the relationship between two variables that is sensitive to dependences that do not manifest in the covariance. The value of MI is non-negative and greater values correspond to stronger dependencies between the variables, with 0 indicating independence between the variables [46][57]. For this reason, it may provide useful information for confirming the results obtained using the Pearson correlation.

It might be advantageous to exclude strongly correlated features based on a previously selected threshold [76] [58].

2.2.2 Normalization

Normalization plays a pivotal role in data analysis and machine learning. By bringing features into a consistent range, normalization facilitates fair comparisons and enhances algorithmic performance. It mitigates bias towards certain variables, preventing them from having an excessive impact simply due to their wide ranges.

Most machine learning algorithms require feature scaling for optimal performance, and it may also accelerate convergence.

This master thesis report discusses two normalization methods: Z-score normalization and min-max normalization.

The Z-score normalization, or standardization, transforms features to have a mean of 0 and a standard deviation of 1, and can be expressed by the following equation:

$$x_{new} = \frac{x - \mu_x}{\sigma_x} \quad (2.2)$$

It effectively handles outliers when compared to other methods. While it is advantageous to many machine learning algorithms, it can distort relationships if the data is not normally distributed.

On the other hand, min-max normalization converts the data into a scale that ranges from 0 to 1. The new value can be calculated, for each feature, as

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.3)$$

This method is sensitive to outliers, which require careful consideration. Nevertheless, this approach is useful when variables are not normally distributed. Binary variables prevail in our data set, and they will remain unchanged with the min-max normalization. Furthermore, the min-max normalization will provide us with a transformed data set that is easier to interpret

2.2.3 Missing values

It is essential to identify the presence of missing data and to establish if there is a pattern associated with it. One first approach is to calculate and evaluate the percentage of missing values in each variable [71].

Imputation is a method for dealing with missing values in which these values are determined and included in the data [25]. When the number of missing values is low, a simple imputation of the missing values with the median value of the parameter has shown to be useful in clinical data of MM patients [25]. Imputing the missing values with the mean of the observed values or with the last observation, or even using a missing category indicator, are commonly used methods. Nevertheless, this may lead to serious bias [77].

Multiple imputation has been used across a number of studies. It includes using the assumed distribution of the observed data to estimate values for missing data such that multiple data sets are created by imputing the missing values and the results are later combined [83] [77]. However, to acquire valid results, the assumption of the distribution must be made with caution. Also, assumptions such that the data is missing at random or is normally distributed must be dealt with care. Finally, it is important to consider that this is an computationally expensive approach [77].

Multiple imputation has the potential to enhance the reliability of medical research. Three methods of multiple imputation by chained equations were compared in generated data on breast cancer and melanoma. Polytomous regression yielded the best results, while using random forest had the largest variance, the most biased results and was very sensitive to noise. Predictive mean matching was nearly as accurate as the former, and had the advantage of a shorter processing time [24].

On the other hand, it is possible to use predictive models to estimate the missing information based on the other variables in the dataset. Nevertheless, it is also computationally expensive and requires a large amount of data. One example is using K-nearest neighbors. First, the data is projected in a multidimensional space, defined by the most correlated features to the feature to impute. Then, the value that the closest neighbors take for the original variable with missing data are assigned to the value missing. Figure 2.1 provides a diagram with this process.

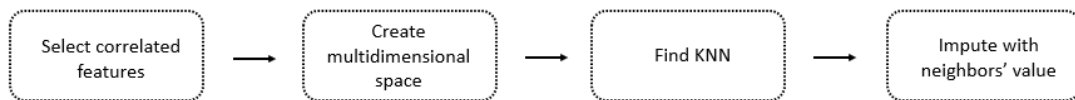


Figure 2.1: K-Nearest Neighbors imputation scheme

Finally, an autoencoder may be employed in order to deal with the missing data problem. Specifically, denoising autoencoders have been utilized for single imputation. The

idea is to leverage from a mask-based attention mechanism in order to concentrate on the observed data. The encoder undergoes pre-training utilizing a denoising technique, wherein a randomly selected portion of the inputs is intentionally set to zero for each sample. This method was found to be useful and even outperform state-of-the-art algorithms, including when there is a large number of missing values. One limitation includes high computational cost due to the large number of parameters to adjust [1] [79]. The general theory underlying denoising and undercomplete autoencoders is explained in the dimensionality reduction chapter.

2.2.4 Class imbalance

Class imbalance is a challenge that must be evaluated in the dataset, as it is crucial to solve it to ensure that the classification model does not become biased towards the majority class. There are multiple ways of addressing this issue [56].

First, one solution would be oversampling, which can be done by simply duplicating instances of the minority class or, for example, with the Synthetic Minority Oversampling Technique (SMOTE) [76]. These methods have the disadvantage of possibly introducing overfitting in the minority class, as well as, by augmenting the number of samples given to the algorithms, augmenting the computational time.

In contrast, undersampling is a technique that involves randomly decreasing the number of instances in the majority class in order to achieve a balanced class distribution. One potential drawback of this method is it decreases the quantity of data utilized for training the algorithms. This disadvantage becomes particularly relevant when the initial dataset is limited in terms of size and has been shown to perform worse than oversampling in a number of studies [56].

Furthermore, some classification algorithms present in the sklearn library already provide incorporated ways of dealing with class imbalance. For instance, logistic regression and support vector classifier have a parameter to balance the data, where the “balanced” mode uses the values of the label y to automatically modify weights so that they are inversely proportional to class frequencies in the input data. The sample weights will be multiplied by these weights. This method is easy to implement and guarantees model robustness without the need to resort to oversampling.

2.3 Survival Analysis

Survival analysis comprehends the statistical procedures for which the outcome variable is the time until a certain event occurs. It is useful for the study of disease evolution and treatment response [19].

Some patterns are common in survival analysis. Cancer patients have surviving curves with a UBNE (Used Better than New) pattern, which means, usually, their life expectancy increases the longer they survive the disease [Downey2011].

Its major goals include interpreting and comparing the survivor and hazard functions, as well as evaluating the relationship of explanatory variables to survival time, which may be accomplished using the Cox proportional hazard approach, for example [19]. All these concepts will be discussed below.

2.3.1 Censored data

Censored observations are those for which we do not know the exact survival time. They correspond to patients who, for example, withdraw from the study, those who are lost during follow-up and those who do not suffer the event until the end of the study [44].

Most data is right censored, which include, patients for which the event has not happened before the end of the study or that withdraw from the study. In this case, we know that the survival time is at least the last observation, but it is not possible to know the exact value. Left censored observations occur when the event occurs before a time t , but we do not know the exact time. Finally, interval censored occurs if the survival time is within a certain unknown time interval [44] [19][43].

There are three assumptions to be considered about censoring: it must be independent, random and non-informative [19].

2.3.2 The survivor $S(t)$ and the hazard $h(t)$ functions

The survivor function $S(t)$ provides the probability that an individual will outlive a certain time t . When using real data it will be a step function so that $S(0) = 1$ and $S(\infty) = 0$ [19].

$$S(t) = P(T > t) \quad (2.4)$$

The hazard function $h(t)$, also called conditional failure rate, provides the instantaneous potential for the occurrence of the event per unit time, assuming the individual has lived up to time t . It is a rate that corresponds to a probability per unit of time and it ranges from 0 to ∞ and is expressed by the equation 2.5 [19].

$$h(t) = \lim_{\Delta t \rightarrow 0} \frac{P(t \leq T < t + \Delta t | T \geq t)}{\Delta t} \quad (2.5)$$

There is a relationship between $S(t)$ and $h(t)$ and they can be derived from each other [19].

2.3.3 The average survival, the average hazard rate and the median survival

The average of survival times is denoted by T and can be calculated as the mean of the survival times of all data, excluding the censored data [19].

The average hazard rate, h , corresponds to the division of the total number of events by the sum of the observed survival times, as is represented in equation 2.6. The higher the h value, the lower the probability of the group of surviving [19].

$$h = \frac{\#events}{\sum_{i=1}^n t_i} \quad (2.6)$$

The median survival time corresponds to the value of "t" for which the total probability of survival after a period of time is 0.5 [43].

2.3.4 Kaplan-Meier

The Kaplan-Meier estimate is a useful and simple way of calculating the proportion of participants who survive during a specific period of time [43].

The Kaplan-Meier curve corresponds to the probability of surviving in a given period of time, considering small time intervals. Some assumptions are made: censored observations, as well as patients that are recruited later in the study, have similar surviving prospects as the patients that continue to be followed; and we assume that the event happens during the interval of time chosen for the study [43].

2.3.4.1 The Log-Rank Test and the Cox proportion hazard model

The Log-Rank test is used to evaluate if two or more Kaplan-Meier curves are statistically equivalent. The null hypothesis is that there is no overall difference between the two curves and the test statistic and the significant can be evaluated using a chi-square distribution with one degree of freedom [if the test statistic is less than the critical value (on one degree of freedom chi-square) than there is no significant difference] [19][43].

$$LogRank = \frac{(O_1 - E_1)^2}{E_1} + \frac{(O_2 - E_2)^2}{E_2} \quad (2.7)$$

Nevertheless, the log-rank test does not allow to test the effect of the independent variables. For that purpose, we may use the cox proportion hazard model [43].

The strategy used is similar to the one used when fitting a logistic regression. The objective is to determine the value of the coefficient, test for its significance and calculate a confidence interval for the effect. To test the significance, there are the Wald and the log likelihood statistics [19]. [for more info see pp. 100-119 of [19]]

The correlation between each variable and overall survival in MM was examined using univariate cox-regression in [60].

2.4 Localized Learning

The purpose of localized learning is to develop several models for the different regions distributed over the input space. Each model is trained only with the nearby instances adaptively, in order to capture the local characteristics [6]. The objective is to construct a learning system that captures useful information from particular subsets of data, in order to model the truth in that area, rather than describing a global phenomenon, by estimating a function [13][38][81]. Localized learning imposes a trade-off between the model capacity and the number of training instances, or locality [15].

Localized learning assumes that it is sometimes impossible to identify a single accurate classification rule that is applicable to the entire population [13] [38]. Furthermore, in real-world issues, data is not uniformly distributed in the input space, which introduces a challenge for a global learning system and may result in superior performance from a localized learning strategy [15][7][2].

Localized learning has several advantages, as it entails this distinct learning paradigm that has been demonstrated to be advantageous and superior to global learning in several applications [13][38]. It can be more effective and more computationally efficient in some scenarios [6] [15]. Some studies have shown that the time complexity for training and testing is reduced, and accuracy is similar or superior, as a result of localizing in several datasets [34].

The reduction of the number of instances upon which the predicted value depends creates models that are more flexible, as its assumptions must only be valid for subsets of the population; more accurate, by utilizing only the instances corresponding to each subdomain; and more interpretable [13][82].

It has the advantage of being computationally less expensive, by adapting the complexity of the model to the properties of the data in each area [13][7]. Moreover, computations regarding different regions may be done in parallel [82]. Finally, local learning algorithms have been effective in addressing class imbalance and ambiguity [7].

Naturally, localized learning also has some drawbacks. For instance, it loses a global perspective on the data and may overlook a data pattern [38]. Moreover, localized models are not always superior or equivalent to global ones. For instance, there may be an issue with overfitting when the underlying true model is simple but has a great deal of noise [13].

A simplified scheme of the localized algorithms is shown in Figure 2.2.

One important aspect is to differentiate local and localized learning. Local learning algorithms are those that are focused on a specific region in the input space. The k-nearest neighbors (KNN) method is the simplest local algorithm [6][81].

On the other hand, localized learning consists of using a global algorithm applied to a subset of the data [6]. Examples of global learning algorithms that may be used for localized applications include classification trees, logistic regression and support vector machines (SVM) [13]. Localized learning and its examples will be discussed below.

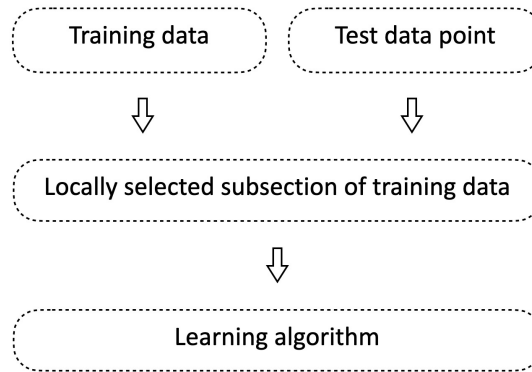


Figure 2.2: Localized Learning approach [6].

2.4.1 Implementation of Localized learning algorithms

There are several ways to implement localized learning by using local algorithms.

The simplest methods are the instance-based techniques, in which a local classification rule is constructed for each training instance based on the adjacent training examples. One example is the KNN, as is shown in Figure 2.3 [13][15].

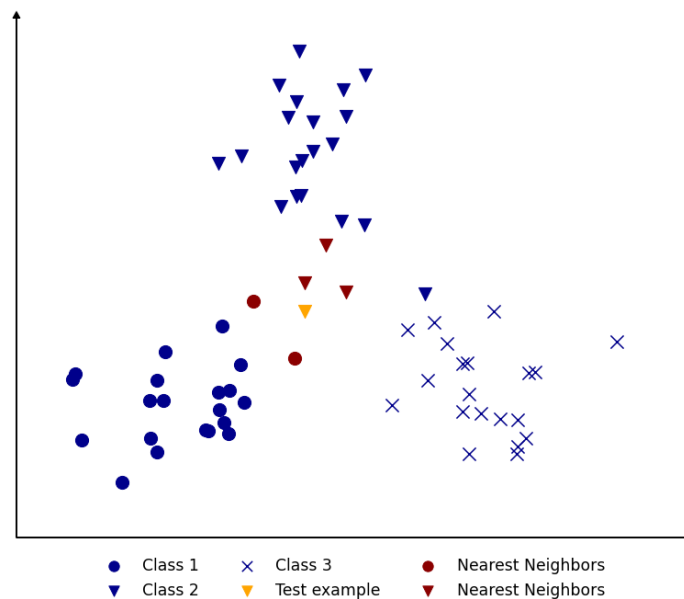


Figure 2.3: Instance based approach using the 5 nearest neighbors. A testing point (yellow triangle) is classified as the majority of its k-nearest neighbors (red triangles and dots).

Other group consists of multiple prototype approaches, which utilize prototypes to represent training data and create partitions of the predictor space. They have the advantage of reducing the memory amount required for instance-based techniques and being less costly computationally. One example to achieve this approach is to use K-means clustering [13] [16]. This method is shown in Figure 2.4.

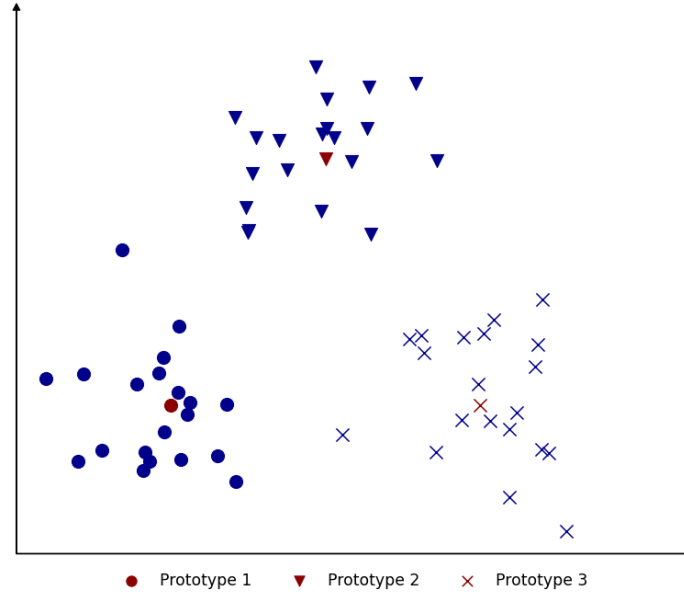


Figure 2.4: Multiple prototype approaches represent the training data using a number of prototypes. The blue dots, triangles, and arrows each indicate data points represented by their respective prototype.

Local kernel techniques, such as local kernels SVM, provide a compromise between global and observation-specific approaches. These Kernels can be square, if they provide uniform weight to a subset of the data points and exclude of the others, or smooth, if they give decaying weights, dependent on the distance, as is shown in Figure 2.5. The Gaussian kernel and the Parzen windows algorithm are examples of smooth kernels [13][15].

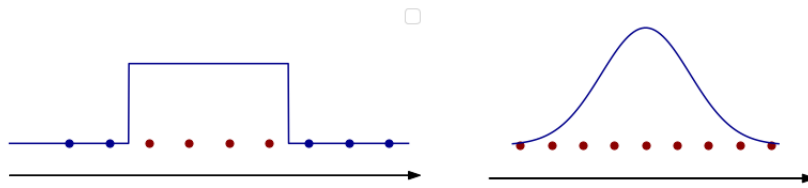


Figure 2.5: Smooth and Square Kernels. On the right, there is an example of a square kernel, in which only some points (in red) are considered, with the same weight, and the remaining are discarded (blue). On the left, a smooth kernel is used, which gives decaying weights to all the data points (red), depending on their distance. Adapted from [6].

Other example of local kernel techniques includes a localized multiple kernel learning method. It is comprised of several weighted sums of many kernels in distinct input areas. This model has higher accuracy than global multiple kernel learning and reduces the model complexity by storing few support vectors [33].

Recursive partitioning algorithms divide the input space into hyper-rectangles before fitting each one with a classifier. An illustration are classification trees [13].

Finally, discriminant adaptive techniques involve classifiers that give greater weight to training data near the class border [13].

Localized learning as shown to perform well in multiple situations. One example of localized learning applied to oncology includes the combination of K-means with consensus clustering with cohort-specific decision trees for classification in breast cancer patients [5].

2.4.1.1 Localized SVM

Global SVM is susceptible to overfitting when the number of training instances is low due to their large VC dimensions, but also is very expensive computationally for large datasets [16]. Some localized alternatives include the Localized SVM (LSVM) and the Profile SVM (PSVM). They have both shown to perform better than global SVM for several datasets [16].

LSVM builds a SVM model for each testing instance based on the training examples in its vicinity. Therefore, it has a high computational cost. It is sensitive to the neighborhood size, similar to KNN. Additionally, it is not flexible, as the neighborhood size is fixed for all test instances [16].

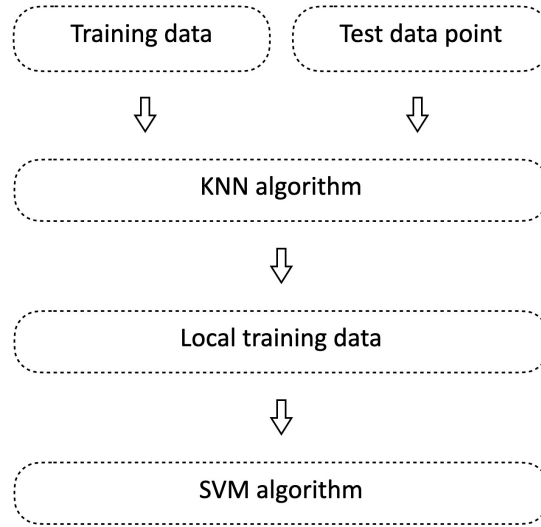


Figure 2.6: Localized SVM [6]

On the other hand, PSVM divides the training set into clusters, creates a linear SVM model for each cluster independently and assigns each testing example to its closest cluster, classifying it using the correspondent model. It is less computationally expensive by reducing the number of created models, and has shown to perform similarly to LSVM [16].

PSVM has similar time complexity to global SVM, while including the expense of clustering and SVM implementation, since it reduces the rapidly growing complexity of

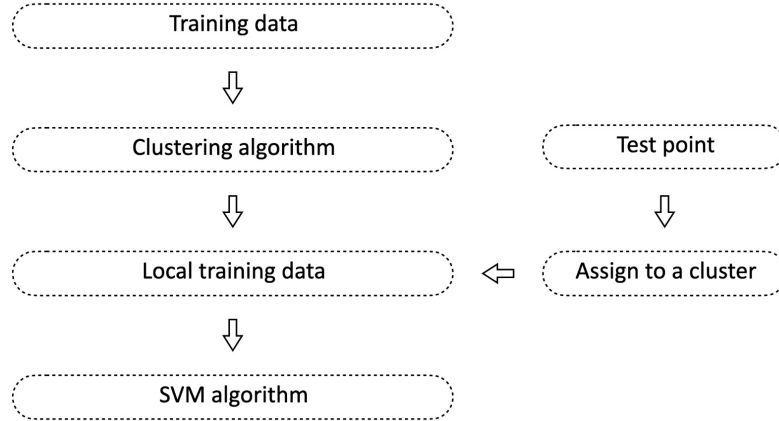


Figure 2.7: Profile SVM

global SVM as the dataset size increases. Furthermore, it increases classification accuracy by incorporating local data features [21].

PSVM may employ a number of clustering techniques, for example K-means or "MagK-means", a variant of the traditional K-means algorithm that minimizes the class imbalance within each cluster [16].

Advantages of localized SVM include better performance than global SVM in some data sets [16] [7]. Additionally, localized SVM generates small SVM models, mitigating the issue that the complexity of SVM rapidly increases as the number of training instances grow. Also, localized SVM stores less support vectors than global SVM [34]. [7].

2.5 Dimensionality Reduction

The aim of dimensionality reduction is to reduce the dimension of a high-dimensional data set, although preserving the essential properties of the full data matrix [86]. Dimensionality reduction methods have been used for data on MM and other cancers [30]. Examples include principal component analysis on gene signatures and self-organizing maps on clinicopathological features[87].

Dimensionality reduction is useful as a preprocessing step for various reasons. Firstly, simpler models are more robust on smaller datasets. By lowering the input dimensions, the complexity of the learning algorithm is reduced. It eliminates unneeded input, hence decreasing the expense of subsequently removing it. Moreover, with fewer characteristics, it is easier to comprehend the data's underlying mechanism. Dimensionality reduction is also important for plotting and visually analyzing data. Finally, dimensionality reductions helps us deal with the curse of dimensionality [4].

Dimensionality reduction is possible since not only certain features on the data set may be irrelevant to our data analysis, but the "intrinsic" dimensionality of the data may also be smaller than the number of accessible features [4].

Feature selection and feature extraction are the two primary techniques for reducing dimensionality. In feature selection, the most informative subset of features is chosen, and in feature extraction, a new set of features is created by combining the original ones [4].

Manifold-based learning is used for feature extraction. It assumes that data lies in a densely sampled manifold to be unrolled. These algorithms may be linear, as is PCA, or non-linear, such as Isomap and t_SNE [9], and are detailed below.

2.5.1 Principal Component Analysis

One unsupervised feature extraction method is the Principal Component Analysis (PCA). This is a multivariate technique which entails a sequence of data transformations in order to acquire new uncorrelated data variables. The principal objective of this technique is to provide dimensionality reduction by maximizing the data variability [4] [70]. PCA is useful for data visualization [4].

The direction with the largest projected variance is called the first principal component, as is shown in Figure 2.8. The orthogonal direction that captures the second largest projected variance is called the second principal component, and so on [86].

In order to obtain the principal components (PC), the eigenvectors and eigenvalues of the data covariance matrix must be calculated. The eigenvectors correspond to the PC, whereas the eigenvalues give us the retained variance value for each PC. In other words, the first PC can be obtained by determining the eigenvector of the covariance matrix of the sample corresponding to the largest eigenvalue. PCA algorithm scheme is shown in Figure 2.9. The linear combinations of the original variables with the PC give us the new variables [4].

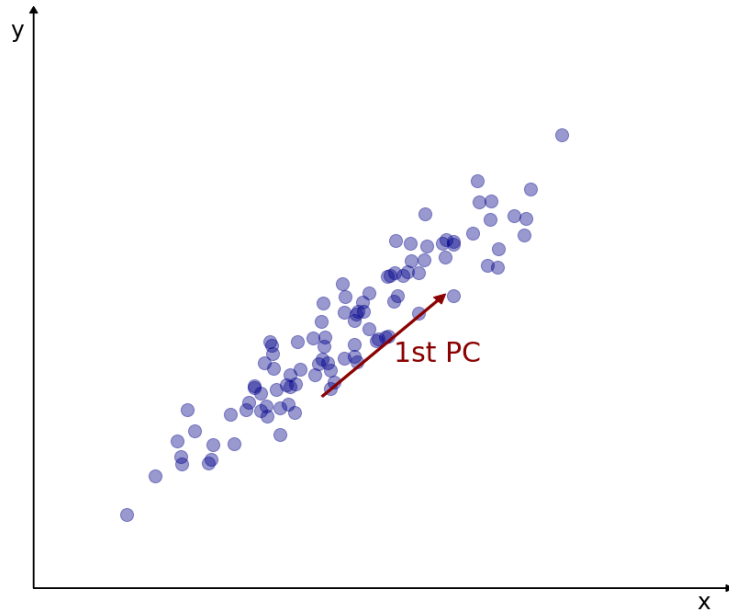


Figure 2.8: Principal Component Analysis. The direction of the first principal component is represented by the red arrow.

Data preprocessing before PCA application is important as it is sensitive to outliers [4]. Additionally, PCA must not be utilized when there are more variables than samples, as information about the original variables may be lost [37].

2.5.1.1 Selecting the number of principal components

Some PCs have little contribution to the variance and may be discarded. There are several ways to decide which PCs can be removed [4].

The Kaiser method includes ignoring the eigenvectors whose eigenvalues are inferior to the average input variance, which means, the eigenvectors which contribute less to the variance than the mean of the original features. If the variables were normalized prior to the PCA, then PC with a corresponding eigenvalue superior to 1 should be selected [4] [37].

Also, PC can be selecting by using the cumulative proportion of explained variance, by choosing a threshold [86].

It is possible to generate a Scree Graph that depicts the variance as a function of the number of eigenvectors, in descending order of their associated eigenvalues. The PCs after the "elbow", the point where the curve slope decreases sharply, becoming nearly horizontal, shall be discarded [4]. However, this technique may be subjective and inadequate for PC selection. This approach is commonly employed as a complementary method [50].

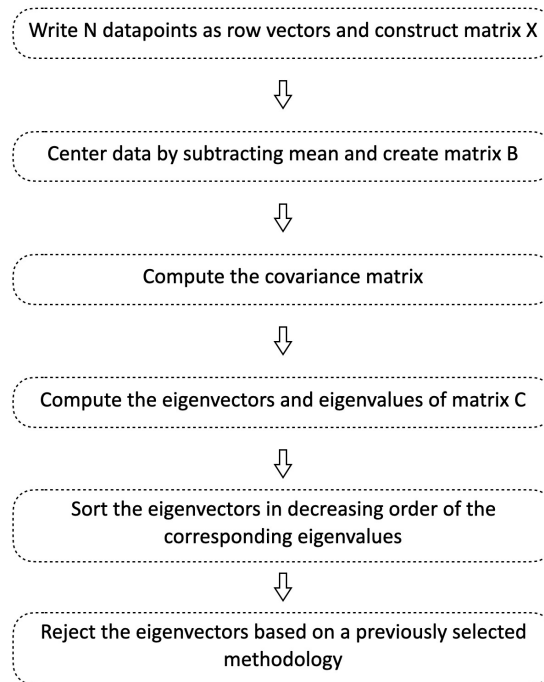


Figure 2.9: PCA algorithm. Adapted from [53].

2.5.2 Self-Organizing Maps

The Self Organizing Map (SOM) is a competitive learning algorithm inspired by the ordered sensory signals mapping in the cerebral cortex [53].

These maps are capable of preserving topological relations, also called topology preservation, while reducing the size of the representation space [45] [53].

SOM have shown to be capable of describing topological relations of input signals in a one or two dimensional representation [45].

The usual arrangement in SOM includes neurons arranged in 2D (although 1D or even 3D are also possible as this depends on the intrinsic dimensionality in the data), with lateral connections between themselves, so that neurons that are together should represent similar features. This lateral interaction is described as the "Mexican Hat", as is shown in Figure 2.10. The winning neuron attracts the neurons in the network that are closest to it, repels the ones that are further away, and disregards the most distant points [53].

The SOM algorithm consists of [53]:

1. **Initialization** - selecting the number of neurons, the number of dimensions and the initial values for the weight vectors (for example, randomly)
2. **Learning** - repeating the following steps until the map stops changing or a maximum number of iterations is met.

For each data point, select the best-matching neuron (similarity by Euclidean distance). Update that neuron weight vector.

Update the weight vector of all other neuron.

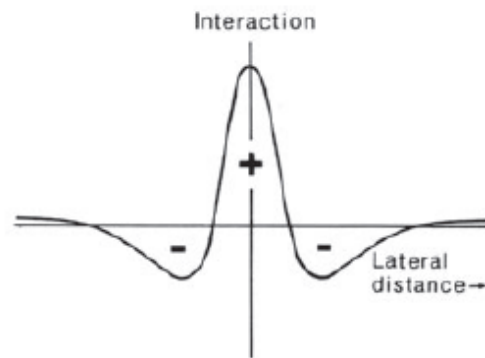


Figure 2.10: The "Mexican Hat" function of lateral connections from [45]

Reduce the learning rate and adjust the neighborhood function.

3. **Usage:** for each test point, select the most similar neuron.

The neighborhood function must be modified such that in a first phase, called ordering, when the network is randomly unordered, the neighborhood is large, but as the algorithm begins to fine-tune in a second phase, called convergence, it shrinks. This decrease in the neighborhood size occurs at a rate that may be chosen [53].

2.5.3 Isomap

Isomap, known as Isometric Mapping, is a non-linear, unsupervised, graph-based feature extraction algorithm. [9]. It aims to preserve the distances between points inside a manifold, regardless of how far apart the points are [53].

It estimates the distance along a manifold, which is called geodesic distance, and applies a multidimensional scalling (MDS) for dimensionality reduction [4].

Assumptions include that the manifold is a smooth convex region and the embedding is an isometry [39]. [Explain this better]

The geodesic distance is approximated by the sum of the distances between the points along the manifold. To calculate this distance, Isomap creates an undirected weighted graph whose nodes correspond to the data points. Two nodes are connected if their distance is less than some threshold, ϵ , and the edge weight corresponds to the Euclidean distance between those points. The geodesic distance corresponds to the length of the shortest path between two nodes. Then, a MDS is applied to the geometric distances [4].

This algorithm performs poorly when the manifold has holes [39].

The parameter ϵ needs to be carefully tuned to prevent topological instability resulting from wrong connections: if it is too small, the graph may be divided in more than one component, but if it is too large, edges not representing the manifold may be created and create a higher dimensional space [4] [9].

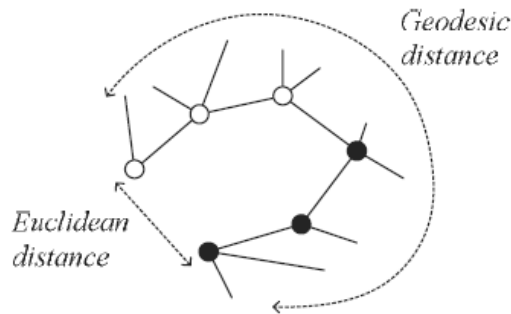


Figure 2.11: Geodesic distance [4].

2.5.4 Undercomplete autoencoders

The autoencoder is a type of deep neural network trained to reconstruct its own input. It consists of a input layer representing the original data, a series of hidden layers representing the transformed features, and a final, output layer, that closely resembles the input layer. It is a method of nonlinear feature extraction based on unsupervised learning, as no class labels are provided [18][10].

The primary objective of the autoencoder is to acquire an encoding of the data in the deep layers that more accurately represents the information contained in the data. This encoding may have higher or lower dimensionality than the original data. Undercomplete autoencoders produce a data encoding with lower dimensionality than the original data. Undercomplete autoencoders produce an informative data representation that can be used for multiple purposes, such as data compression and feature extraction [18].

The autoencoders consist of two major sequential components: the encoder and the decoder, as is shown in Figure 2.12.

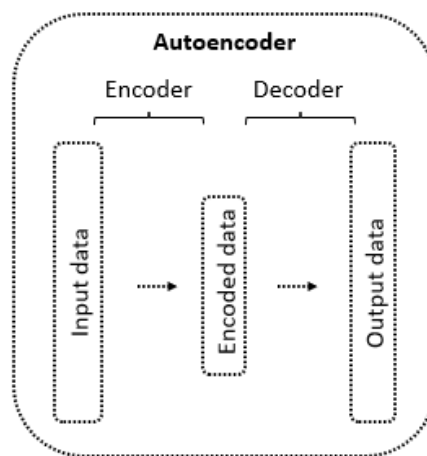


Figure 2.12: Autoencoder Architecture

The two components may be linear operations, in which case the obtained result would

be the same representation as the results of PCA. If they are not linear, the autoencoder is able to learn a non-linear manifold of the data [18]. As mentioned, autoencoders can be used for dimensionality reduction, and have even demonstrated lower reconstruction errors than linear methods such as PCA [10].

In order to prevent getting the identity operator for the encoder and decoder, several regularization methods may be used. The most common includes imposing a bottleneck by making the dimension of the data representation obtained by the encoder smaller than the original data [18].

An autoencoder is often trained using one of the many backpropagation variants, typically the stochastic gradient descent method[10].

One particular important aspect of constructing an autoencoder is the choice of activation function. The activation functions play an important role in the training of neural networks. There are several options for the activation functions, including the sigmoid, the hyperbolic tangent, rectified linear unit (ReLU), amongst others [8]. These three activation functions are shown in Figure 2.13.

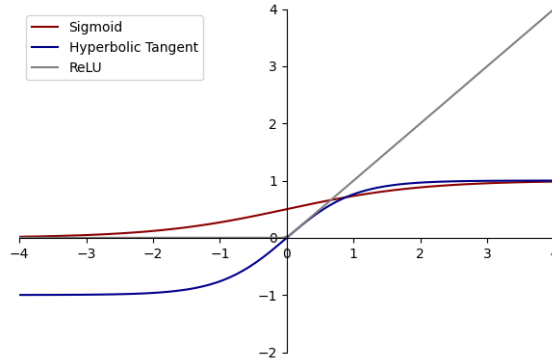


Figure 2.13: Activation functions for neural networks

The sigmoid activation function is a non-linear function that receives input from $(-\infty, +\infty)$ and translates it to the range $[0;1]$, which may cause the gradient to vanish in deep networks. It is defined as

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2.8)$$

The hyperbolic tangent has similar structure to the sigmoid function but the output ranges from -1 to 1. It can be defined by

$$\tanh(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (2.9)$$

Finally, the ReLU is the most common activation function in deep learning applications, as it has demonstrated superior training performance. It can be defined as

$$\text{ReLU}(x) = \max(0, x) \quad (2.10)$$

It is basically the combination of two linear activation functions: the identity function for all the positive values, and 0 for all the negative input values. For this reason, it is simple and is capable of increasing the representation sparsity of the network, as it often outputs 0. Moreover, many modifications of the original ReLU function have been used [8].

2.6 Cluster Analysis

The purpose of clustering is to group the data based on similarities in the input features, doing so without being provided with the data labels [14][53]. Cluster analysis is an unsupervised learning approach used to split a set of items into homogeneous groups, such that two random objects belonging to the same cluster are more similar to each other than two random objects belonging to different clusters [14][84].

Cluster analysis may be utilized for exploratory data analysis, to acquire insight into the nature of the data, including the identification of relationships between instances, patterns, hidden features and outliers. In addition, it may be utilized to generate a natural classification of the data based on its characteristics. Cluster analysis may also give a more compact representation of the data and aid in identifying the most representative items in a dataset [84].

The measure of similarity or dissimilarity used to produce the clusters may be diverse. For example, one may consider mutual similarity between objects, or similarity of objects to prototypes of the clusters [84].

In the medical field, these techniques enable the transformation of multidimensional data into simplified and reproducible clusters, assisting researchers in maximizing the value of data and reducing costly healthcare resources [60] [41]. On the other hand, clustering medical data faces the challenge of dealing with the large number of disease markers available [41].

The use of clustering methods in medical research includes clustering of disease risk factors, clustering of illness symptoms, and clustering of gene expression data [66]. Some examples of methods used for clustering in the medical field include K-means, fuzzy C-means, hierarchical, DBSCAN, Expectations Maximization algorithm and consensus clustering [67][66][85]. Several cluster analysis methods have been used specifically in multiple myeloma, including Gaussian Mixtures models and spectral clustering [60][30].

2.6.1 Hierarchical Clustering

Hierarchical clustering algorithms comprise a collection of traditional clustering techniques that consist in successive aggregation or division of the instances and their subsets. This process produces a dendrogram, a tree-like structure that represents these divisions or aggregations [84].

Hierarchical clustering algorithms may be divided into agglomerative and divisive algorithms. In the former, the algorithm begins with as many clusters as there are datapoints and merges related clusters until there is only a single cluster. In the latter, the algorithm begins with a single cluster and splits it until each cluster includes only a single instance [4]. Using this technique is considerably less common than using agglomerative clusters [84].

In the agglomerative algorithm, at each iteration, the two closest clusters are merged.

Furthermore, the distance may be defined in several ways, including single-link (the smallest distance between all pairs), complete-link (the largest distance between all pairs), the average link (the average distance between all pairs), the ward link (minimizes the sum of squares of distances between objects and the corresponding centroid), amongst others [4][84].

Using the differences between levels in dendrograms, the number of clusters in hierarchical clustering may be determined by maximizing the distances between clusters [4].

The hierarchical clustering methods also present some shortcomings. First, if an object has been misclassified, it may not be shifted afterwards to the correct cluster. Also, they lose their clarity with larger datasets [84].

2.6.2 Density-based methods

For density-based methods, a cluster is defined as an area of high density of instances surrounded by low density areas [84].

One example of these methods is DBSCAN. With DBSCAN, all points belonging to a cluster are mutually density-connected. It requires two initialization parameters: the maximal distance between internal points and the number of neighbors. There are methods to help decide the best maximal distance [84].

This algorithm presents with some advantages comparing to the ones explained above. Firstly, it deals with outliers, as they are located in low density areas and are not assigned to any cluster. It allows clusters with irregular and convex shapes. There is no need to know the number of clusters a priori. Finally, this algorithm is only slightly sensitive to the order in which the points are added to the clusters [84].

Limitations include the need for different clusters to have similar densities. Also, the algorithm strongly depends on the definition of distance chosen [84].

HDBSCAN is yet another density-based algorithm. It introduces an advantage, since there is no need for all clusters to have similar densities [84].

2.6.3 Evaluating clustering results

A crucial step in cluster analysis comprises evaluating the cluster results. There are multiple ways to evaluate the obtained clusters and select the algorithm that functions best for our data. They can be separated into external and internal validity measures.

External validity measures compare the outcomes of two partitions of the data set. One example is the Rand Index. They require a partition to compare the results of our clustering. Therefore, they are not appropriate in the majority of cases, where this label is not provided.

On the other hand, the internal validity measures rely on properties inherent to the data. These methods rely on some concepts that characterize the characteristics of effective clustering. They essentially measure cluster compactness and separation, meaning, points

should be close to the points in the same cluster and distant to those in different clusters. They are crucial for clustering evaluation when the comparison with a label is not possible, as they provide quantitative measures to assess the quality of the clusters formed during the clustering process. These metrics enable informed decisions on which clustering algorithm or parameters are most appropriate for the specific data. They include the Silhouette score, the Calinski-Harabazs and the Davies-Bouldin indexes.

The Silhouette score is considered one of the best performing validity measurements. It defines clusters compactness as the distances between all points in the clusters and separation as the distances between the points in one cluster and the points in the closest cluster. The Silhouette score consists of the average of the silhouette values of all points in the data. The silhouette value can be obtained as

$$s(x_i) = \frac{b(x_i) - a(x_i, c_j)}{\max(b(x_i), a(x_i, c_j))} \quad (2.11)$$

where $a(x_i, c_j)$ is the average distance of a point to all the points in the same cluster and $b(x_i)$ is the minimum over the average distances of a point to all points in another cluster. The Silhouette score is in $[-1, 1]$ and higher values indicate better clustering.

The Davies-Bouldin index defines compactness as the distance of a point to the centroid of the cluster and separations between centroids distances. Its value is non-negative and the objective is to minimize it.

Finally, when using the Calinski-Harabazs Score, compactness is defined as the distance of the points to its centroid, similarly to the Davies-Bouldin score, whereas separation here is defined as the distance of the centroid of the cluster to the centroid of the entire data set. Its value is non-negative and the objective is to maximize it.

2.7 Classification

Supervised learning models receive the input vectors and their matching target vectors and can be divided into regression and classification. In regression model, the target value is continuous; in classification, it is discrete. [14].

The objective of classification is to assign the input vector to one of the discrete classes, partitioning the input space into regions, separated by decision boundaries [14].

Several classification methods have been used to improve MM diagnosis, identify high risk patients, predict the response to treatment and survival outcomes. These include SVM, decision trees, random forests and random survival forests [3][55][76][64][61]. Decision trees perform well in disease prediction and have the added advantages of being transparent in how they model the data [5].

Additionally, convolutional neural networks and other DL methods were utilized to detect MM bone lesions in CT, MRI, and PET scans [3].

2.7.1 Logistic Regression

Logistic regression is linear model used for two-class classification problems, rather than regression. It takes advantage of the logistic sigmoid function

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.12)$$

which varies around a threshold value, but is almost constant away from this value. It varies from 0 to 1, as is shown in Figure 2.14.

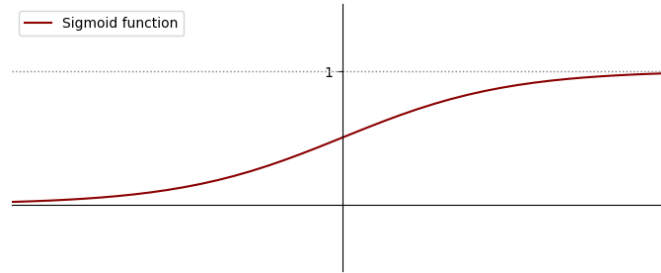


Figure 2.14: Sigmoid or logistic function.

The maximum likelihood is then used to determine the parameters of the logistic regression model [14].

2.7.2 Support Vector Machines

Support Vector Machines (SVM) can be used for classification as well as regression. They are margin optimization methods that allow the model to be written as the sum of the influence of the support vectors. The support vectors are the subset of the training instances that contribute to the decision boundary [4].

The optimization problem of SVM is a convex optimization one, therefore, the global solution can be obtained by solving it analytically. The optimal separating hyperplane is the one that maximizes the margin between the different classes, as is shown in Figure 2.15 [4].

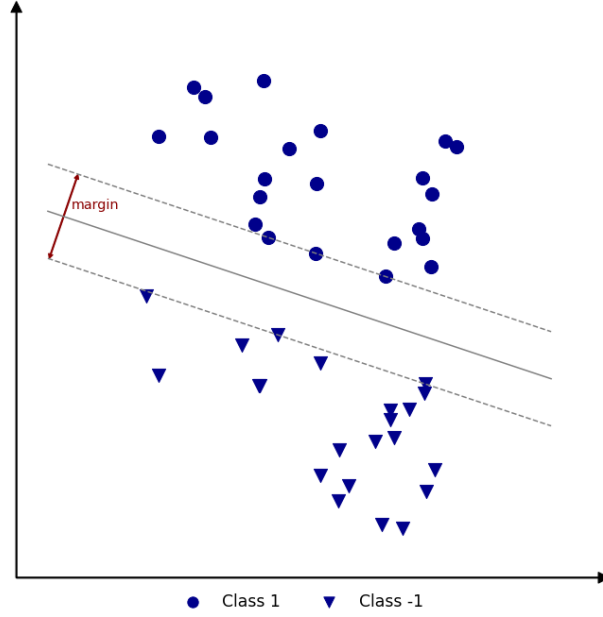


Figure 2.15: SVM is a maximum margin classifier. Applied to a dataset composed of two linearly separable classes, the objective is to maximize the margin between them. Adapted from [4] [6].

This distance can be written as

$$\frac{|w^T x^t + b|}{\|w\|} \quad (2.13)$$

where w is the weight vector, x^t is a support vector and b is the bias [86] [4]. We want this distance to be at least some value ρ . We multiply it for the class label, $y \in \{-1, 1\}$, in order to always obtain a positive value that we want to maximize:

$$\frac{y^t(w^T x^t + w_0)}{\|w\|} \geq \rho, \forall t \quad (2.14)$$

The problem in trying to maximize ρ is that there is an infinite number of solutions to this problem. For that reason, we may impose a margin greater than one by scaling $\vec{w}$. This can be demonstrated by

$$y^t(w^T x^t + w_0) \geq \rho \|w\| \quad (2.15)$$

where $\rho \|w\| = 1$. Then, we may minimize the $\|w\|$ to maximize the distance ρ [4]. The new objective function is

$$\min \frac{1}{2} \|w\|^2 \quad (2.16)$$

Subject to

$$y^t(w^T x^t + w_0) \geq 1 \quad (2.17)$$

The method of Lagrange multipliers (λ) is used to solve constrained optimization problems, which corresponds to finding maxima and minima of multivariable functions subjected to constraints [14]. We may formulate this problem by creating a function L , that combines the function we want to minimize and its constraint

$$\begin{aligned} L &= \frac{1}{2} \|w\|^2 - \sum_{t=1}^N \alpha^t [y^t(w^T x^t + w_0) - 1] \\ &= \frac{1}{2} \|w\|^2 - \sum_t \alpha^t y^t (w^T x^t + w_0) + \sum_t \alpha^t \end{aligned} \quad (2.18)$$

where α is the Lagrangian multiplier and N is the number of data points. Some of the Lagrangian multipliers are 0, which correspond to the points that are not support vectors.

The function L derivative may be used to find its minimum with respect to w (Equation 2.19) and w_0 (Equation 2.20) and to be maximized in respect to $\alpha^t \geq 0$ [4].

$$\frac{\delta L}{\delta w} = 0 \Rightarrow w = \sum_t \alpha^t y^t x^t \quad (2.19)$$

$$\frac{\delta L}{\delta w_0} = 0 \Rightarrow \sum_t \alpha^t y^t = 0 \quad (2.20)$$

Using these values, the dual optimization problem can be expressed by

$$L = \sum_t \alpha^t - \frac{1}{2} \sum_t \sum_s \alpha^t \alpha^s y^t y^s (x^t)^T x^s \quad (2.21)$$

where x^t and x^s represent two datapoints, α^t and α^s are the corresponding Lagrangian multipliers and y^t and y^s are the respective labels.

The dual formulation is useful when we have data in a high dimension, as it depends on N , the sample size, and not on the data dimensionality, or when we want to use the Kernel Trick [4].

2.7.2.1 Non-linearly separable classes

Moreover, in real-world applications classes may overlap. In this case, where the data will not be linearly separable, we look for the hyperplane that incurs the least error.

To do this, we define non-negative slack variables, ξ^t , which represents the distance of the misclassified point to the margin. When $0 < \xi^t < 1$, the data point is correctly classified but is in the margin, whereas when $\xi^t \geq 1$ the point is misclassified [6] [4].

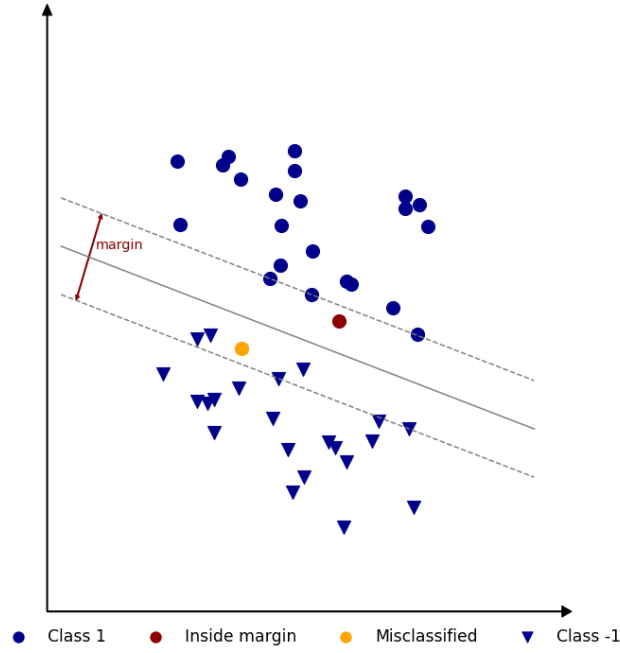


Figure 2.16: SVM applied to a dataset composed of two non-linearly separable classes. The red dot represents a data point that is correctly classified but is inside the margin. The yellow dot represents a misclassified data point. Adapted from [4] [6]

We define soft error as the sum of all the values of this slack variables, and a penalty term C , that compromises between the misclassification tolerance and the margin maximization, may be added,

$$L_p = \frac{1}{2} \|w\|^2 + C \sum_t \xi^t \quad (2.22)$$

subject to the constraint:

$$y^t(w^T x^t + w_0) \geq 1 - \xi^t \quad (2.23)$$

Adding this penalty, Equation 2.18 can be written as

$$L = \frac{1}{2} \|w\|^2 + C \sum_t \xi^t - \sum_t \alpha^t [y^t(w^T x^t + w_0) - 1 + \xi^t] - \sum_t \mu^t \xi^t \quad (2.24)$$

where μ^t are the new Lagrange parameter to guarantee the positivity of ξ^t . The derivative in respect to ξ^t must also be set to 0

$$\frac{\delta L}{\delta \xi^t} = C - \alpha^t - \mu^t = 0 \quad (2.25)$$

and since $\mu^t \geq 0$, then $0 \leq \alpha^t \leq C$. We must then subject Equation 2.21 to this in order to deal with non-linearly separable classes [6] [4].

2.7.2.2 The Kernel Trick

Another possibility when dealing with non-linearly separable classes is to perform a non-linear transformation to the data to map it in a new higher dimensional space and then use a linear model in this new space.

The d -dimensional x space may be mapped to the k -dimensional z space, as is shown in equation 2.26.

$$z = \phi(x), \text{ where } z_j = \phi_j(x), j = 1, \dots, k \quad (2.26)$$

We use the dual formulation as it depends on N and not on k , which may be larger than N , and get

$$L = \sum_t \alpha^t - \frac{1}{2} \sum_t \sum_s \alpha^t \alpha^s y^t y^s \phi(x^t)^T \phi(x^s) \quad (2.27)$$

subject to the previously explained constraints.

With the kernel trick, we may substitute the inner product of the feature map functions, $\phi(x^t)^T \phi(x^s)$, which is expensive in high dimensional feature maps, by a kernel function $K(x^t, x^s)$. This way, there is no need to map the instances in the new space, instead a positive kernel function may be directly applied to the input space [4], as is shown in Figure 2.17.

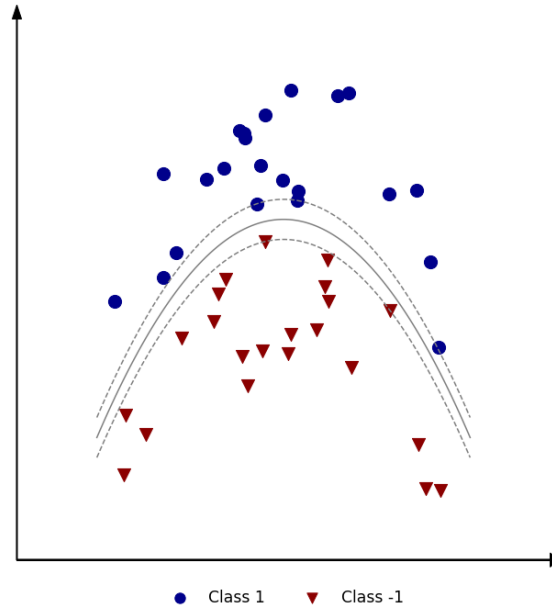


Figure 2.17: SVM applied to a dataset with non-linearly separable classes, using a polynomial kernel of degree 2. Adapted from [4].

Summarizing, this may be written as:

$$L = \sum_t \alpha^t - \frac{1}{2} \sum_t \sum_s \alpha^t \alpha^s y^t y^s K(x^t, x^s) \quad (2.28)$$

Another major advantage of using the Kernel trick is that it is independent of the input space dimensionality [6].

2.7.2.3 Applications, Advantages and Disadvantages

SVM have demonstrated utility in a variety of applications. Text classification, image recognition, including face or handwriting recognition, geological and environmental sciences, finance, insurance and bioinformatics are among examples. In particular, in bioinformatics, they have been utilized for cancer diagnosis, protein interaction and homology identification, gene expression categorization, survival analysis and medication screening. [17] [69] [54]. They have already demonstrated state-of-the-art results in pattern recognition, regression estimation and time series prediction in reasonably sized datasets [35] [53].

SVM have several advantages. Firstly, their sparseness. As a result of the minimal number of non-zero parameters, the discriminant function is only dependent on a small portion of the input data set, the support vectors, which allows fast computations [42] [54]. On the other hand, SVM are robust algorithms, as they perform well in the presence of outliers or extreme data points [54].

They are capable of finding a unique global solution. Since they provide a convex optimization problem, they do not get stuck in local minima [69] [54]. Additionally, they can deal with complex and high-dimensional dependence structures, large quantities of dissimilar data and complex functions if the suitable kernel can be constructed [54] [69] [42].

They have a certain flexibility, since the discriminant can lie in a infinite number of positions. Nevertheless, this may lead to overfitting [54]. Also, they are nonparametric, therefore, they do not assume any prior knowledge about the data [54].

Nonetheless, SVM have some disadvantages as well. They do not work well on extremely large datasets. The memory space grows quadratically with the training set size, and, given twice as much data, the algorithm will take four times as long to run [59].

Moreover, Kernel selection is challenging while implementing SVM. Using a Kernel that permits data separation without introducing too many dimensions would be the optimum option. Typically, this is accomplished by trial and error, which may be time-consuming. However, this issue appears to be less significant than it first appears because different Kernels provide comparable outcomes [35] [69] [59].

Finally, another question that rises is when to choose the dual formulation. The dual representation of the problem showed that it can be successfully used in very high dimensional feature spaces [53] [17], however it is worse for a large number of instances in a low-dimensional space [6].

2.7.3 Ensemble methods

Ensemble methods explore the idea of decision by committee, meaning, more than one learner are joined to make decisions.

Boosting is the most popular ensemble method. It is powerful and was firstly created for classification problems. The idea is to combine a collection of poor learners, which are defined as those who are only better than chance, in order to obtain better results [Marsland] [Bishop].

In boosting, classifiers are trained in sequence using a weighted form of the data set. These weights are obtained based on the performance of the previous classifiers, such that points that were misclassified are given a higher weight as those that were correctly classified. Once all the classifiers have been trained, a weighted majority voting scheme is used to make the final predictions [Bishop].

2.7.3.1 Histogram-Based Gradient Boosting Classifier

Histogram-Based Gradient Boosting classifier is one example of a tree-based ensemble method based on the boosting methodology that is inspired on the LightGBM().

In the first step, this estimator bins the input into groups, used to leverage from histograms instead of all the continuous values when building the trees. It uses the log-loss loss function for classification. Since it uses histograms, it does not require sorting the samples in each node, as the traditional GradientBoostingClassifier(), but rather only once in the beginning of the boosting process, making it faster.

It has built-in support for the missing values, which avoids the need for an imputer or other ways of dealing with the missing values. During training, the tree grower learns at each split point whether samples with missing values should go to the left or right child, based on the potential gain. When predicting, samples with missing values are assigned to the left or right child consequently. It also provides support for categorical features [49].

2.8 Interpretability

Interpretability is crucial in the medical field to provide insight on several aspects of a model. Firstly, for it to be easy to understand, but also to provide understanding on what factors are important for the predictions, for examples, to provide understanding of the underlying causes of diseases.

There are several methods to provide model interpretability, that will be mentioned below [11] [36].

Firstly, it is important to separate transparent and opaque models, also generally called "black boxes". Simpler models, such as logistic regressions and decision trees, are often transparent and easily interpreted. The coefficients may provide feature importance in linear models, and understanding the rule-based learners as the decision trees may lead to understanding of the decision of the model. Nevertheless, this is not always true. For examples, decision trees applied to a large number of features or with a large number of nodes may be too complex for human interpretation. Anyway, in the case where interpretability is crucial, these models should be used as a first approach. Nevertheless, simpler models may show inferior performance than more complex ones. For that reason, more complex models may be needed to improve the classification results [11]citeHenelius2014.

There are several kinds of explanations that may be useful. Firstly, feature importance approaches are an indirect way to explain a model that enable quantitative feature ranking. Feature ranking may, however, disregard the importance of highly correlated features. Simplification of opaque models using simple surrogate models may lead to the creation of rules that approximately show how the models makes their predictions. Visualizations are one of the most powerful interpretation tools and may provide simplifications on the behavior of the model that are easy to interpret. They may provide information on some specific aspects of the model, such as the decision boundary [11] [28].

As was mentioned, there are several methods to provide model explainability. However, implementing them individually may lead to missing important information. Moreover, there is no established and unique way of combining these techniques.

Some techniques will be exposed below.

2.8.1 Permutation

Another approach to increase interpretability of the "black box" models was proposed, initially in the context of random forests, but that is model agnostic and can be applied to any type of predictive model. It is called permutation importance, and results of randomly permuting the values for each independent feature and comparing the effect in the performance of the model to the model applied to the original data. It assumes that, if one feature does not contribute to the prediction, then permuting their value will have no effect on it [36].

Since this approach is used in one feature at the time, assuming the feature are

independent, it may ignore important feature interactions. In the case of correlated features, permuting the values of one of them may have little impact on the performance of the model, which will lead to a score identical to non-important attributes, although this may not be the case [36]. It is important to notice that, in the context of medical data, complex feature interactions are often present, which is a raising concern in our case.

2.8.2 Shap Values

The Shap (Shapley Additive Explanations) values are a widely used approach for model explainability and are based on the Shapley value from cooperative game theory [72].

The Shapley value is a measure of the marginal contribution of each player to a coalition playing a game. It is a way to determine the credit each player should receive when playing in a coalition. This is determined by, starting with a single player, adding each player at random, and calculating the amount which is adherence contributes to the value of the coalition [73].

When considering the adaptation to the Shap in machine learning, these values measure the contribution of each feature to the classification. For each instance, they explain how to get from the base value $E[f(z)]$ that would be predicted if we did not know any features to the current output $f(x)$, by averaging the results from all orderings possible. A simplification of this is shown in Figure 2.18 [52].

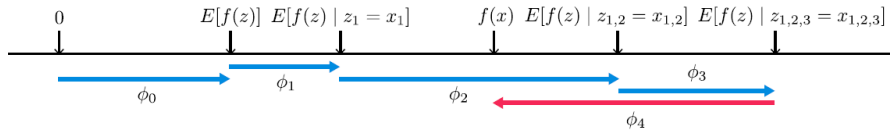


Figure 2.18: Shap values [52]

The Shap value attributed to each feature results from averaging the results obtained from each of these instances [52].

One important assumption made is that of feature independence. It has the disadvantage of only taking into account linear relations, and may mislead when is the only method used. Also, it is computationally expensive [72].

2.8.3 Partial Dependence Plots and Individual Conditional Expectation

Since visualization of the function obtained from all the original features is not possible due to the dimensionality in question, it is important to use other visualization tools. The partial dependence plots (PDPs) can be used to help interpret any opaque model. The single variable PDPs show the changes in the predictions of the model when a specified feature varies and the remaining ones are constant. This method accounts for the influence of other variables, and may even show the form of the relationship between the feature and the outcome. When considering categorical variables, bar plots may be used. [28].

Furthermore, the PDP can be misleading in the case of a strong dependence between features [29].

The individual conditional expectation (ICE) operates on instance level, showing the decision boundary of the model as a function of a single feature. The averaging ICE plot gives us the PDP.

There are some limitations on the use of machine learning in medical applications that must be addressed. Firstly, a substantial knowledge gap exists on the causal relationships on the correlations that ML systems might extract from vast sets of data [78]. Several AI techniques lack transparency in their elaboration processes [3].

Regarding the data, there is the problem of high dimensionality due to the amount of clinical, genetic and therapeutic data available [68]. Additionally, Data collected as part of routine care may be biased against particular patient racial/ethnic groups [31].

On the other hand, being a retrospective study, it suffers from a high risk for bias [58].

2.9 Limitations

There are some limitations on the use of machine learning in medical applications that must be addressed. Firstly, a substantial knowledge gap exists on the causal relationships on the correlations that ML systems might extract from vast sets of data [78]. Several AI techniques lack transparency in their elaboration processes [3].

Regarding the data, there is the problem of high dimensionality due to the amount of clinical, genetic and therapeutic data available [68]. Additionally, Data collected as part of routine care may be biased against particular patient racial/ethnic groups [31].

On the other hand, being a retrospective study, it suffers from a high risk for bias [58].

EXPLORATORY DATA ANALYSIS AND PREPROCESSING

The primary purpose of this chapter is to acquaint the reader with the data, by illustrating patterns and relationships, as well as any issues that need to be addressed.

Descriptive statistics and data visualizations will be offered to provide the reader with an overall understanding of the structure and content of the data. Some issues regarding this data will be addressed. Namely, the presence of variables that do not carry any information (variables that take no values or only one value) or that carry repeated information (variables that have the same values). Also, information will be extracted from some variables to create more informative ones. The presence of missing values and outliers will be addressed.

3.1 General Description

The original data contains 309 features and 7069 observations. Due to the longitudinal aspect of this dataset, these observations correspond to 2404 patients with measurements taken at various times.

The first analysis of this dataset included finding and removing variables that will not be usefull for further studies. This process is shown in Figure 3.1.

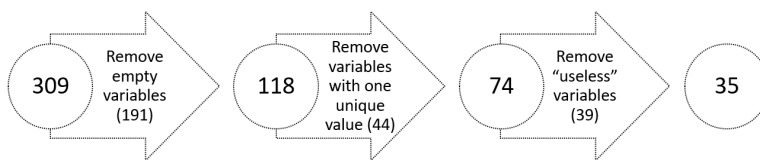


Figure 3.1: Schema showing the process of removal of variables

On the original dataset, 191 variables had no values and were removed from the dataset. Also, there are 44 variables that take only a single value. These were also removed as they have no utility for our objectives.

Additionally, 39 variables were removed. They consisted of variables related to dates and variables that were directly dependent on other variables (for example, the year of birth when we also have the age).

The remaining 35 variables were studied and are described in the following sections.

The data was retrospectively collected from patients that were diagnosed between July 1993 and June 2014. The study ended on the 16th march 2022.

3.1.1 Lines of treatment

A patient with multiple myeloma may be subjected to a series of treatments in order to control the disease. For that reason, as explained above, each patient may have multiple observations registered at different times, corresponding to the time of each treatment line.

The distribution of the total number of lines of treatment each patient was submitted to is shown in Figure 3.2.

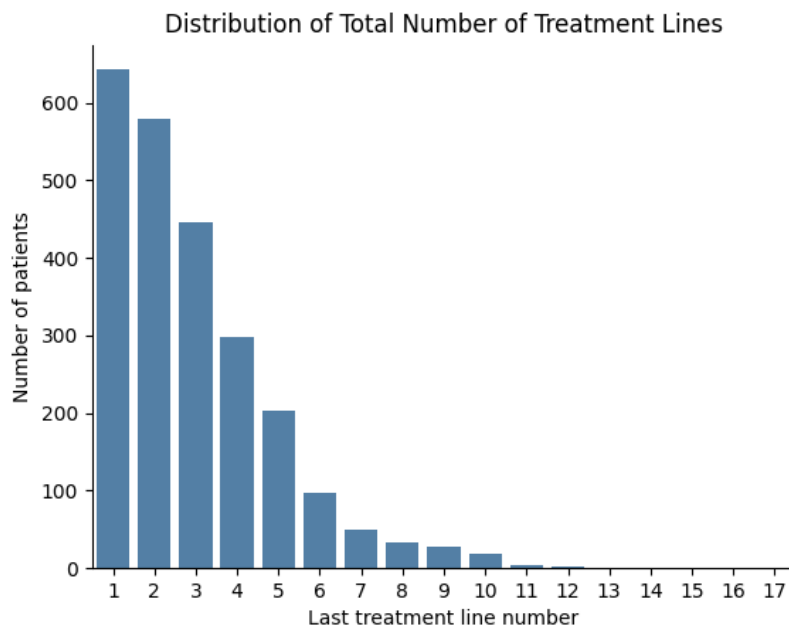


Figure 3.2: Distribution of patients by total number of treatment lines

The mean number of treatment lines is 2.94. As Figure 3.2 shows, the number of patients decreases as the total number of treatment line increases.

For simplicity purposes, in the remaining exploratory data analysis, except when mentioned otherwise, the 437 patients (18.18%) that had five or more treatment lines were grouped together, as is shown in Figure 3.3.

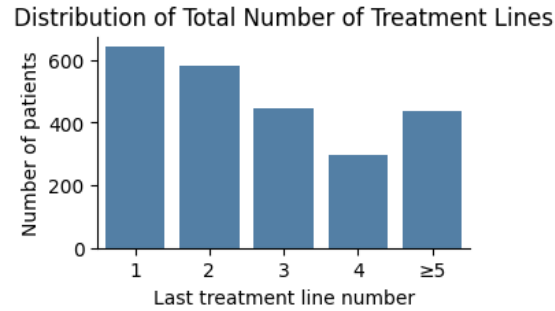


Figure 3.3: Distribution of patients by total number of treatment lines with censored values at ≥ 5

3.1.2 Data collected at diagnosis

There are nine variables in this data set that were collected at the time of diagnosis. Thus, they assume one unique value for each patient and do not vary through each line of treatment.

These variables consist of sex, age, LDH value, the presence of three genetic mutations and the scores for the Durie Salmon and the ISS staging systems at diagnosis.

3.2 Variables Description and Analysis

3.2.1 Demographic Data

In order to guarantee confidentiality, the patient sex was substituted by a randomly generated variable by the Janssen Belgium team. The observations include 1205 males (50.12%) and 1199 females (49.88%). Due to the nature of this variable, it will not be used for the machine learning methodologies applied below.

The mean age at diagnosis was 63.45 years. The distribution of age at diagnosis is shown in Figure 3.4.

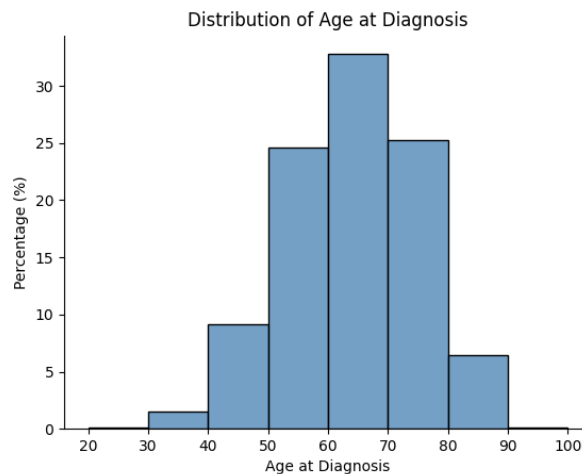


Figure 3.4: Distribution of patients by age at the time of diagnosis

The age at each line of treatment was registered. An empirical cumulative distribution function is shown in Figure 3.5. We do not observe any specific pattern or differences in the distribution of age for each treatment line.

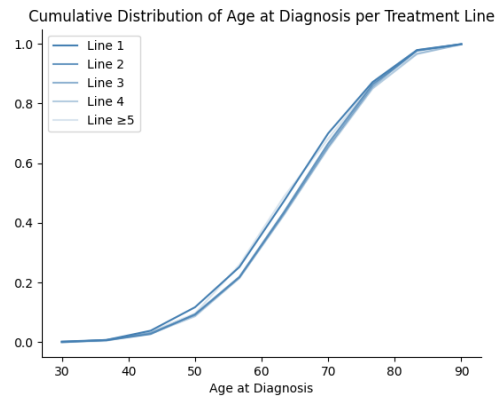


Figure 3.5: Empirical CDF of patients by age at the time of diagnosis per total number of treatment lines

3.2.2 Medical Exams

Regarding results of medical exams, this dataset contains data related to LDH measurements and genetic testing.

3.2.2.1 LDH Levels

LDH measurements are taken at the time of diagnosis and at each treatment line.

Regarding the levels registered at diagnosis, 566 patients have the LDH level registered and 1838 (76.46%) have this value missing. Reference values of LDH are in the interval between 45 and 90 U/L (SI). 549 (97.8%) patients have an elevated LDH. The mean level of LDH at diagnosis is 280.6 U/L. Its distribution is shown in Figure 3.6.

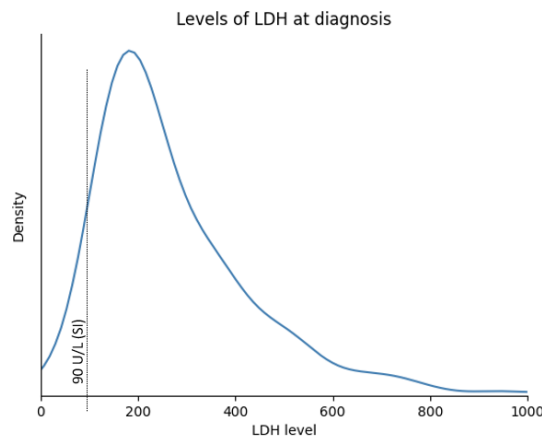


Figure 3.6: Kernel Density Estimation plot of LDH levels at the time of diagnosis

The levels of LDH at each treatment line have 74.00% (5231 observations) missing values. Specifically, 41.06% of the patients have no LDH measurements at any line treatment and 16.72% have this value registered for all the treatment lines. The remaining ones have some values registered and some missing values.

Figure 3.7 shows the boxplots of the LDH values per treatment line. Using the Kruskal-Wallis test, it was shown that there is a significant difference between these 5 groups. Nevertheless, when applying the Mann-Whitney test on consecutive groups, no significant difference was obtained.

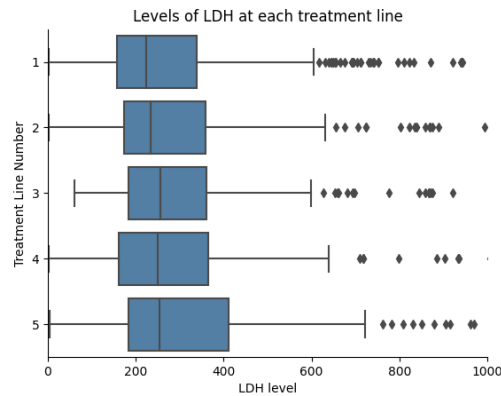


Figure 3.7: LDH levels at the time of each treatment line

Finally, Figure 3.8 shows the relationship between LDH at diagnosis and LDH at each treatment line. It is possible to observe most values are the same for LDH at diagnosis and at first treatment line, which may lead us to believe these may be values corresponding to the same measurements.

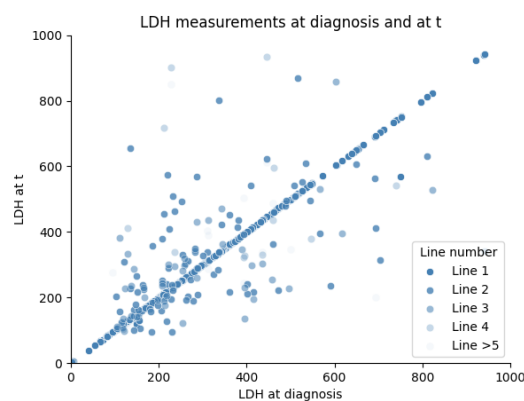


Figure 3.8: Comparison of LDH levels at the time of diagnosis and each treatment line

3.2.2.2 Genetic Mutations

Three genetic mutations related to the presence of multiple myeloma were studied: the deletion of the long arm of the chromosome 13 (del13q), the deletion of the short arm of

the chromosome 17 (del17p) and the transposition between chromosomes 4 and 14 (t4-14). In some cases, the genetic testing took place at the time of diagnosis. Others, occurred at the time of one of the treatment lines, that will be referred as time t.

In our data, we can find the values "Yes", corresponding to the presence of that mutation, "No", corresponding to its absence, "Missing", which is a value registered in data representing the missing information; also, there are missing values (None), which in this case were filled with the value "MV" for this analysis. The distinction between "Missing" and "MV" is that "Missing" was recorded by the doctors who gathered the data, but "MV" was an actual missing value filled during this data analysis.

Figure A.1 in Appendix 1 compares the values registered for each mutation at the time of diagnosis and at the time t. As one may observe, there are no contradictions regarding the presence or absence of each mutation in time.

In addition, there are missing values ("MV") or the value "Missing" in both the diagnostic and t time measurements; nevertheless, some of these values correspond to the presence or absence of the mutation in the other measurements.

Also, the sum of the missing values and the value "Missing" at diagnosis and at t is the same for all mutations. This may be due to the fact that the genetic test includes testing for the three mutations at the same time.

This hypothesis can be confirmed by looking at Figure A.2 and Figure A.3 in Appendix 1. There are 6138 missing data and 691 "Missing" values among the diagnostic observations for all mutations. There are 5260 missing observations and 1312 "Missing" values at time t. The data shows that testing for one mutation entails testing for the other two.

In addition, the "Missing" value appears to indicate that the genetic test was not completed, as it is consistent within the three mutations. In light of this, the value "Missing" shall hereafter be regarded as a missing value. Finally, for each mutation, there is a total of 240 and 497 observations registered at diagnosis and at the time of the treatment lines, respectively.

Since it is expected that the presence of a mutation is stable over time, as our data confirms by the absence of contradictions between measurements in time, we may create one unique column per mutation. For this, we can take all the values that were collected at diagnosis and fill the missing values with the data that was collected during the treatment.

After this, we created a new dataframe with these 3 columns and each patient id, and grouped by patient to obtain a single value for each mutation and patient.

The results show that there were 499 patients that were genetically tested for all the mutations. 134 patients had at least one mutation. Of these, 105 had a unique mutation, 22 tested positive for two of the mutations, and 7 tested positive for all the mutations.

Table 3.2.2.2 shows the results of the genetic tests per patient. The most prevalent mutation in this population was the deletion of the long arm of chromosome 13.

Table 3.1: Number of patients presenting with positive and negative genetic tests for each mutation.

Mutation	Positive	Negative
Deletion 13q	91 (18.24%)	408 (81.76%)
Deletion 17p	44 (8.82%)	455 (91.18%)
Translocation 4-14	35 (7.01%)	464 (92.99%)

3.2.3 Staging Systems

The data includes the disease stage of each patient based on the Durie-Salmon Staging System (DS) and the International Staging System (ISS), both at diagnosis and at each treatment line.

The variables regarding the stage at diagnosis have no missing values, although they may take the value "Unknown". Figure 3.9 shows the diagnosis stage distribution.

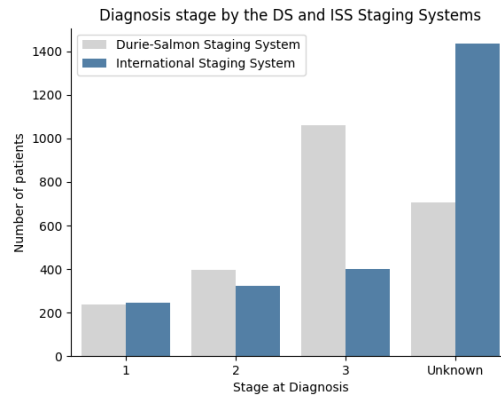


Figure 3.9: Stages at diagnosis based on the Durie-Salmon and the International Staging Systems

The heatmap presented in Figure 3.10 demonstrates that the results from each staging system at diagnosis are frequently different.

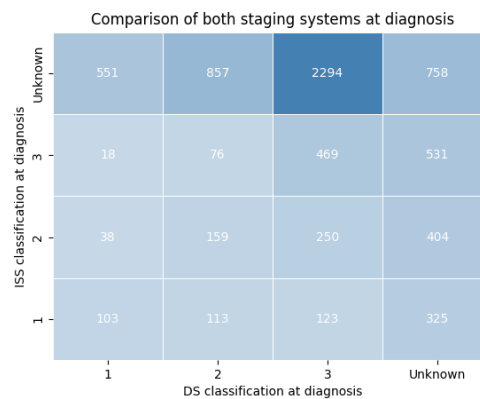


Figure 3.10: Comparison of the Durie-Salmon and International Staging System stages at diagnosis

The variables regarding stage at each treatment line both have 4596 (65.02%) missing values. They also may take the value "Missing". Figure 3.11 shows the distribution of the values of the Durie-Salmon and International staging systems per treatment line.

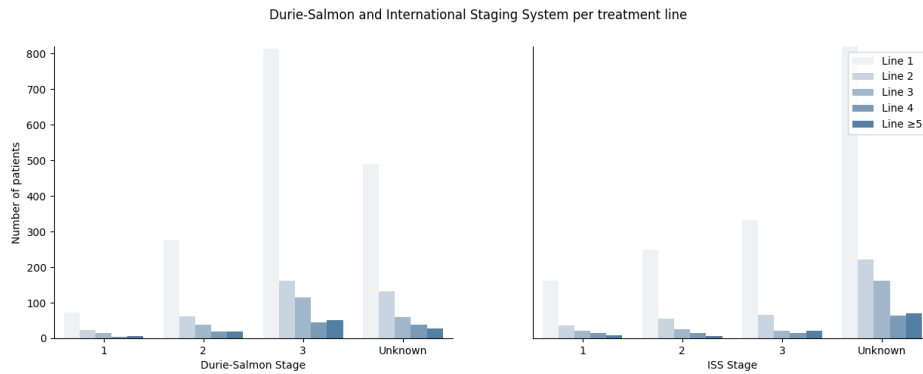


Figure 3.11: Comparison of the Durie-Salmon and International Staging System stages at t(time of treatment line)

3.2.4 Treatment

3.2.4.1 Drug Classes

A variable providing a list of drug classes for each observation is included in the data. This variable contains combinations of six drug classes: "corti" for corticosteroids; "Cyto" for cytostatic drugs; "Imids" for immunomodulatory agents; "PI" for proteasome inhibitor; "Mab" for antibodies; and "Unknown" class. The combinations of these drug classes result in 33 unique values for this column. Also, this variable has 52 (0.74%) missing values.

Table 3.2.4.1 displays the frequency with which each drug class appeared in the drug class list.

Table 3.2: Drug classes included in all the treatment lines

Drug Class	Included	Not included
Corticosteroids	6075 (85.93%)	942 (13.33%)
Cytostatic	4565 (64.58%)	2452 (34.69%)
Immunomodulatory agents	2497 (35.32%)	4520 (63.94%)
Proteasome inhibitor	3035 (42.93%)	3982 (56.33%)
Antibodies	1 (0.0%)	7016 (99.25%)
Unknown	1033 (14.61%)	5984 (84.65%)

To facilitate the use of the information carried by this variable, six dummy binary variables were created, one for each medication class. For each observation, these columns were filled with 1 or 0, depending on whether the drug class was present in the observed drug class combination. For the observations where the value of the original column was missing, these columns were filled with the value "MV".

There is an unique observation where there is the administration of a drug from the antibodies class. It is the case of a 67 year old patient that is being treated in the first

line of treatment. Data regarding the LDH and the presence of mutations is missing. There is only information regarding the ISS stage at diagnosis, which was stage I. This treatment was part of a combination with an immunomodulatory agent and a corticoid. The response to treatment was stable disease. For now, we did not identify any special findings regarding this observation. For this reason, and also due to the very low number of occurrences, the column regarding the prescription of antibodies will be removed.

The original variable containing combinations of drug classes, as well as the column regarding the administration of an unknown drug, were dropped.

3.2.4.2 Drug Regimen

The data also includes a variable that specifies the name of the treatment applied. These treatments include drug names and the autologous peripheral blood stem cell transplant. Similarly to the drug classes, this variable contains 783 unique values that are combinations of 61 treatments. Due to the high number of combinations this variable takes, we decided to use only the drug classes.

3.2.4.3 Regimen Sequences

There are two variables that relate to the regimen sequences. They have information on the sequence of treatment applied to each patient. Nevertheless, they take 1450 and 1908 unique values respectively. For the same reasons explained above, they will not be used in the remaining methods applied to this data.

3.2.4.4 Autologous Peripheral Blood Stem Cell Transplant

This dataset contains two variables that indicate whether or not an autologous peripheral blood stem cell transplant (ASCT) was conducted in each treatment line. Since they contain exactly the same information, one of these columns was dropped.

This dataset contains 778 (32.36%) individuals who have undergone at least one ASCT and 1626 individuals who have not. Of the former, 641 (26.66% of total patients) received one transplant, whereas 137 (5.7%) received two transplants.

Figure 3.12 shows the percentage of patients that had an Autologous Peripheral Blood Stem Cell Transplant performed in each treatment line. As we may notice, the proportion of transplants decreases as the number of treatment lines increases.

Figure 3.13 shows the distribution of the realization of transplants per age of diagnosis.

3.2.4.5 First Line of Chemotherapy

The data contains information on whether a first line of chemotherapy occurred. A total of 342 (14.23%) patients did not have a first line of chemotherapy. The remaining 2055 (85.48%) underwent chemotherapy, and of these, more than 80% underwent chemotherapy treatment in their first line of treatment.

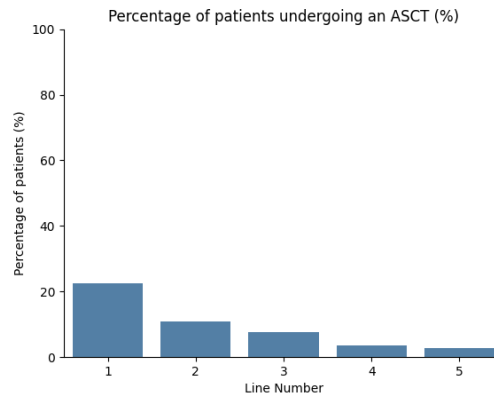


Figure 3.12: Percentage of patients which had an Autologous Peripheral Blood Stem Cell Transplant in each treatment line.

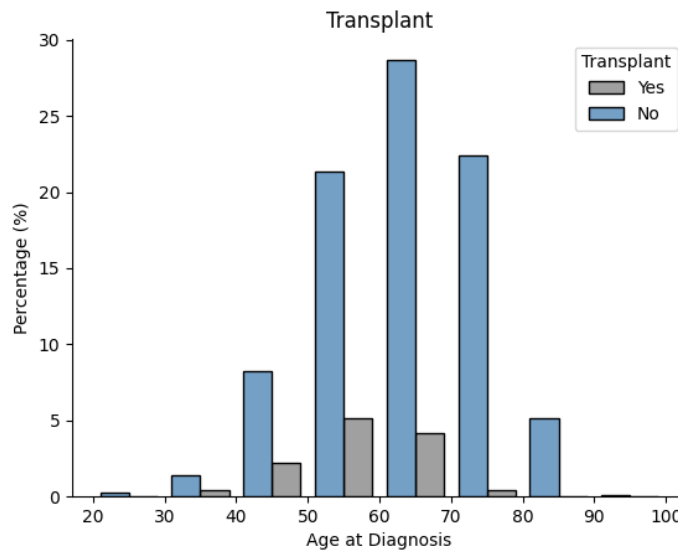


Figure 3.13: Number of patients which had an Autologous Peripheral Blood Stem Cell Transplant per age at diagnosis.

3.2.5 Patient Care Time Intervals

Included in the data, there is information regarding the treatment duration and the time until the start of the following line of treatment. Figure 3.14 shows the distribution of the overall time intervals. The mean line duration is 210.20 days and the mean of the time until the start of the next line of treatment is 417.71 days.

Figure 3.15 shows the distribution of treatment line duration in respect to each of the treatment line numbers. The Kruskal Wallis test was used to compare the median of these 5 groups and the result showed that there is a significant difference among them (p -value = 0.0), meaning, there is at least one statistically significant different group. The Mann Whitney test was used to compare pairs of consecutive groups and the result showed there was a significant difference in all of them.

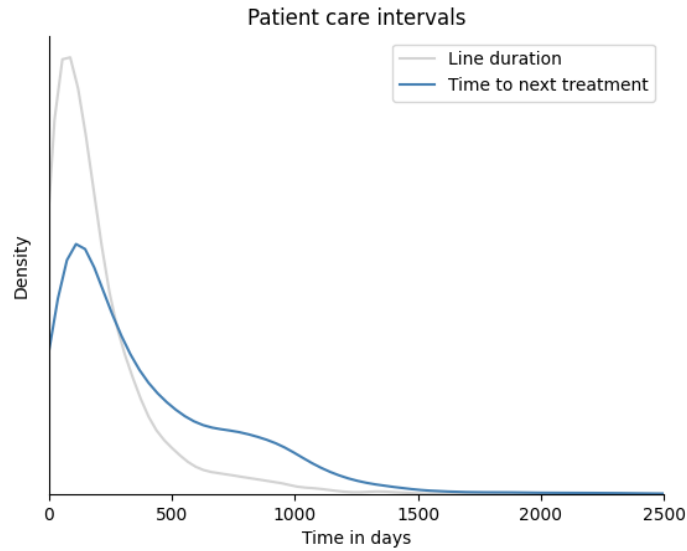


Figure 3.14: Time in days for the duration of treatment and interval of time until the next treatment

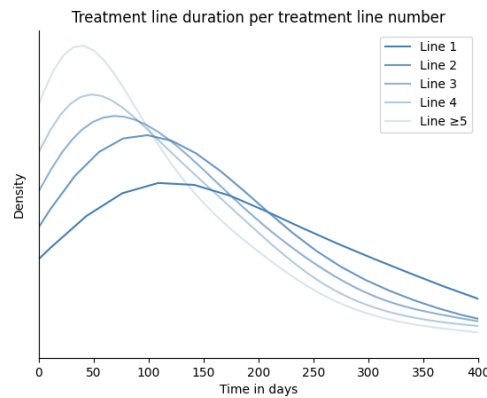


Figure 3.15: Treatment line duration per treatment line number

Again, regarding the time to the next line of treatment, both the the Kruskal Wallis test and the Mann Whitney test showed that there is a significant difference among these (p -value = 0.0) and a significant difference between all consecutive pairs, respectively. Figure 3.16 shows their distribution.

3.2.6 Treatment Response

The variables that indicate the response to treatment include the initial response, the last response and the best response to the treatment.

The first and last response to treatment take the values "CR", meaning complete response; "nCR", near complete response; "VGPR", very good partial response; and "PR", corresponding to partial response. The best response includes these four values and also "MR", minimal response, "SD", stable disease; "PD", progressive disease and "NAP",

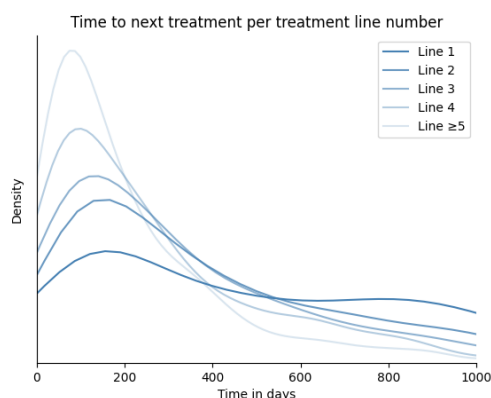


Figure 3.16: Time to next treatment per treatment line number

response unknown. The criteria that dictate these values is shown in Table B in the Appendix.

The variable first response and last response have 3784 (53.53%) missing values each, and the best response has 315 (4.46%) missing values.

Figure 3.17 shows the comparison between first and last response. It shows that the missing values in the variable initial and last response correspond to the same observations. Also, 2848 (86.70%) of the observed responses do not change from the initial to the last response and 437 (13.30%) do change, and only 1.23% have a worse last response than better response.

Relation between initial response and last response

Last Response				
	CR	nCR	VGPR	PR
	Initial Response			
	CR	nCR	VGPR	PR
PR	7	5	14	1751
VGPR	1	3	513	153
nCR	3	176	20	46
CR	408	26	53	106

Figure 3.17: Comparison between initial and last response to treatment

Figure 3.18 shows the comparison between the initial and the best response. As expected, the initial response is always equal or worse than the best response.

Additionally, the responses corresponding to the values "MR", "SD", "PD" and "NAP" in the best response always correspond to missing values in the initial response.

When excluding the missing values, the initial and the best response differ in 427 (13.00%) observations.

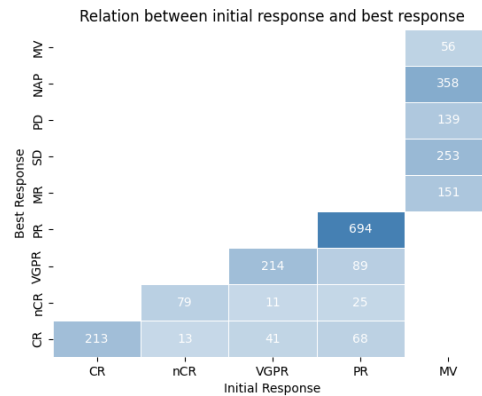


Figure 3.18: Comparison between initial and best response

Finally, figure 3.19 shows the comparison between the initial and the best response. The same phenomena is happening with the missing values. Also, the final and best response are different in a 61 (1.85%) observations.

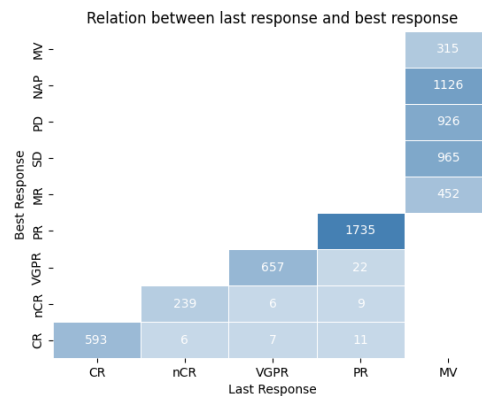


Figure 3.19: Comparison between last and best response

Due to the similarities between all these columns, as well as the presence of a large number of missing values in the initial and last response, the best response will be used to evaluate the response further on.

To facilitate the classification task that will be later performed, a new binary column was added. Its values derive from the best response column such that the response is considered 0 for the "MR", "SD" and "PD", which are considered lack of response in the literature, and 1 for all the other non-missing values. Figure 3.20 shows the values this variable takes per line number.

For simplicity purposes, for the remainder of the study, we will consider that the value "NAP" corresponds to a simple missing value.

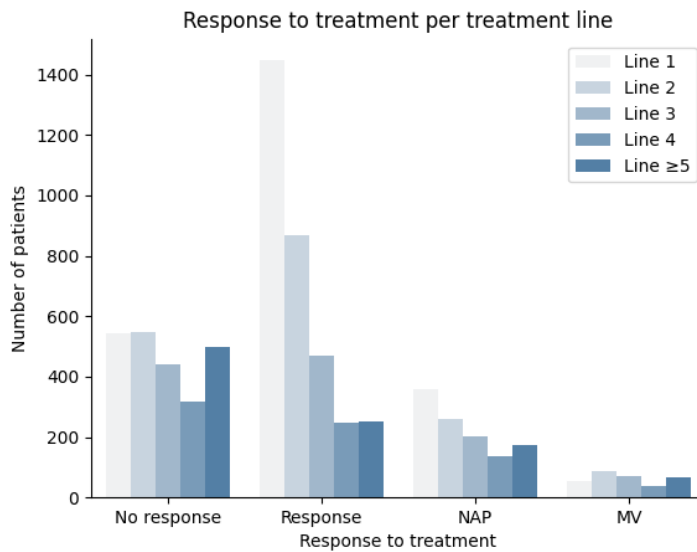


Figure 3.20: Binary variable for best response. NAP - unknown response; MV - missing value.

3.2.7 Survival outcome

There are five variables in the data regarding survival. Two of them signal if the patient survived until the end of the study. Since they have repeated information, one of them was deleted. The variable kept takes the value 1 in case of death, and 0 in case of censored data (if death was not registered until the end of the study or until the patient withdrew from the study). Another pair of variables includes the time since the diagnosis until death occurred, or, if it did not, until the end of the study. Again, one was dismissed. Finally, the last one includes information on which line of treatment death occurred.

In total, 1736 (72.21%) patients died during follow-up. Figure 3.21 shows the Kaplan-Meier curve for the survival.

Figure 3.22 shows the percentage of patients that died in each line of treatment.

The survival analysis was performed for the variables regarding each of the genetic abnormalities. The following Kaplan-Meier curves show the probability of survival in the first five years after diagnosis of the patients with and without each genetic abnormality.

The log rank test showed there is a statistically significant difference between the two groups with or without del17p (p-value = 0.003), but no difference for t(4;14) and del13p, respectively (p-value of 0.91 and 0.68).

Diagnosis age and survival The mean age at diagnosis of the group that survived and the group of people that died during follow-up was 63.03 and 64.54 years, respectively. To compare the median of these two groups (64 and 65 years), the Mann Whitney test was performed and showed a statistically significant difference (p-value = 0.0).

Figure 3.26 shows the Kaplan-Meier curve for the 10 year survival by the age at diagnosis in decades.

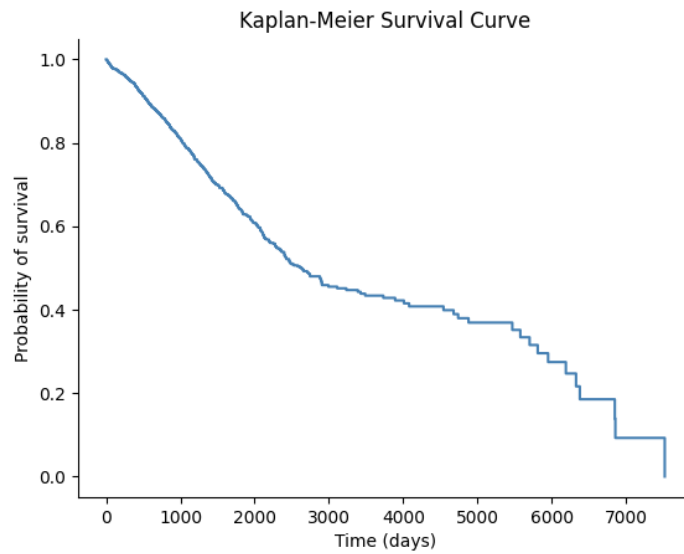


Figure 3.21: Kaplan-Meier curve for survival time since the time of diagnosis.

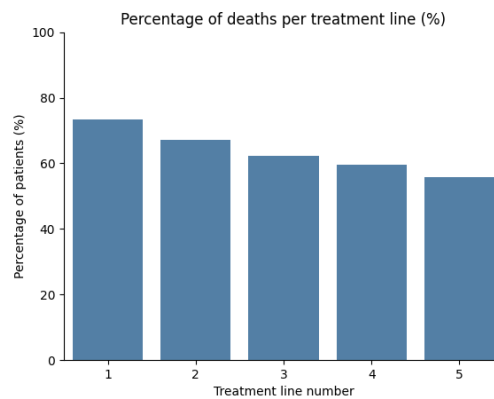


Figure 3.22: Deaths per treatment line

In order to compare the results obtained in this Kaplan-Meier curve, a log-rank was applied. First, a multivariate log-rank test was performed and the test statistic result was 57.09, with a p-value of 0.0. This indicates that there is a statistically significant and large difference in survival for at least one diagnosis age group. For this reason, the univariate log-rank test was applied to consecutive pairs of groups, and the results are registered in Table 3.2.7.

Observably, there is no statistically significant difference in survival between three of the groups: those diagnosed between the ages of 30 and 40, 40 and 50, and 80 and 90. On the other hand, the larger difference is between the 60 and 70 decades.

Diagnosis stage and survival Figure 3.27 shows the Kaplan-Meier survival curve for the 10 year survival for the diagnosis stage given by both staging systems.

Again, a log-rank was used to compare the findings obtained from this Kaplan-Meier

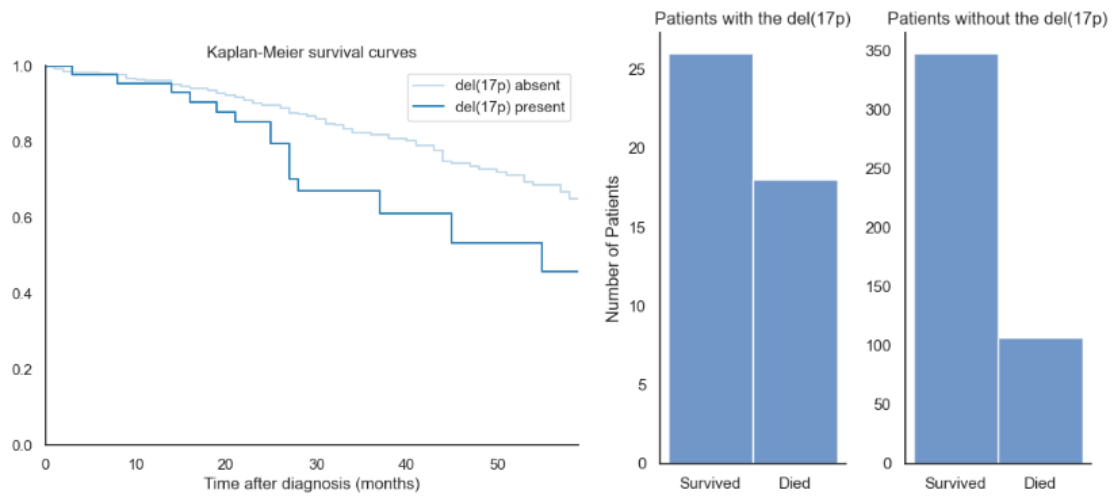


Figure 3.23: Survival Analysis Results for del17p

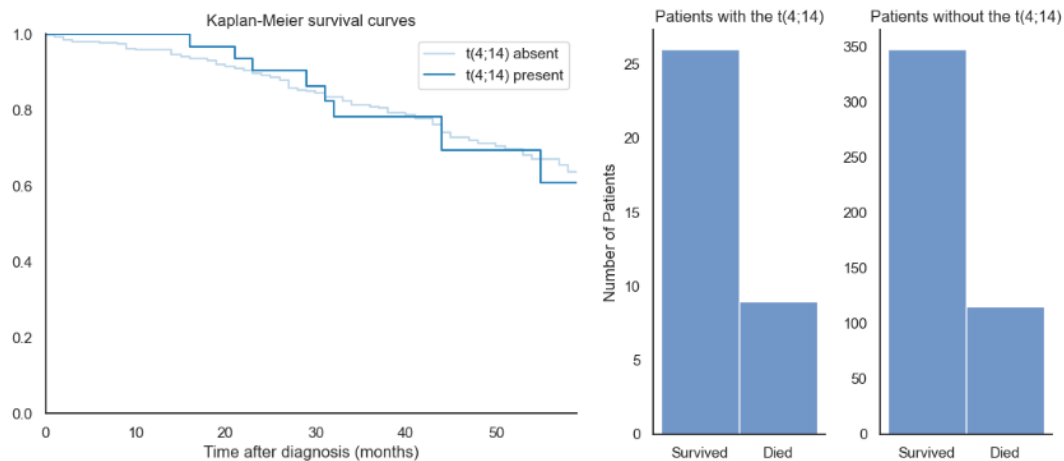


Figure 3.24: Survival Analysis Results for t(4;14)

curve. A multivariate log-rank test was conducted and showed a survival difference based on Durie Salmon and International stages (p-value = 0.0).

3.2.8 Final variables

The final variables are described in Table 3.2.8. The variables include those that carry information regarding the clinical status of the patient, the treatment, and the response to treatment.

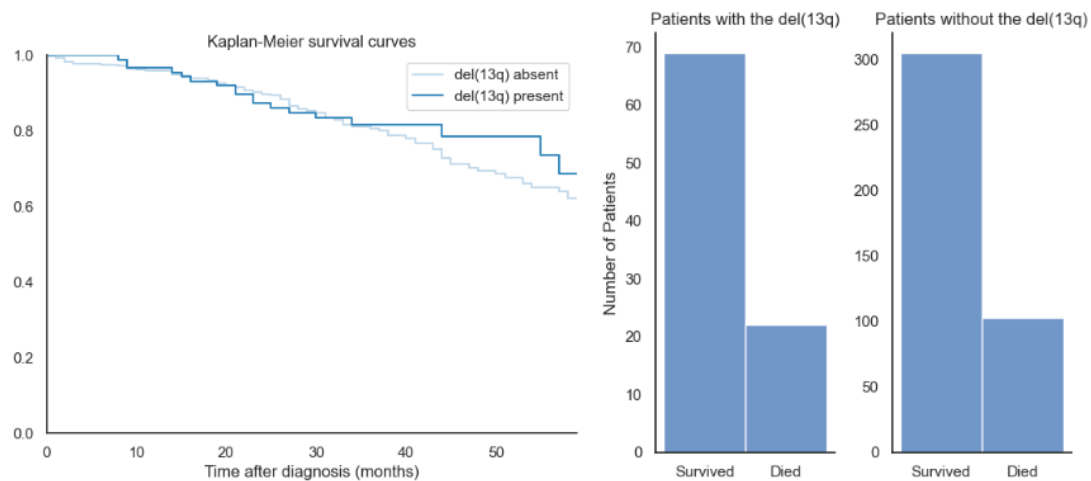


Figure 3.25: Survival Analysis Results for del13q

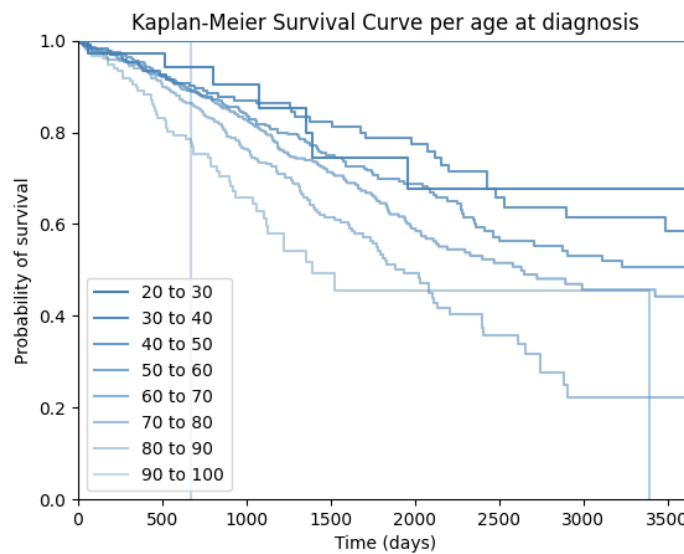


Figure 3.26: Kaplan-Meier Survival curve per age at diagnosis

3.2.9 Missing values

After the changes explained above, the data has 28 variables, 13 of which do not have missing values. Table 3.2.9 shows the number and percentage of missing values in the remaining 15 variables.

Depending on the percentage of missing values, the importance of the variable and the reason behind the missing values, these variables will be addressed differently.

For the 4 variables regarding the drug classes, the number of missing values is rather low (0.75%). For that reason, the missing values will be imputed with the most common

Table 3.3: Log-rank test results for the survival comparison between consecutive age (in decades) at diagnosis.

Decades	Test Statistic	p-value
20 - 30	4.073749	0.043555
30 - 40	0.000095	0.992242
40 - 50	1.008210	0.315332
50 - 60	5.157071	0.023152
60 - 70	8.114415	0.004391
70 - 80	5.854430	0.015538
80 - 90	0.199490	0.655133

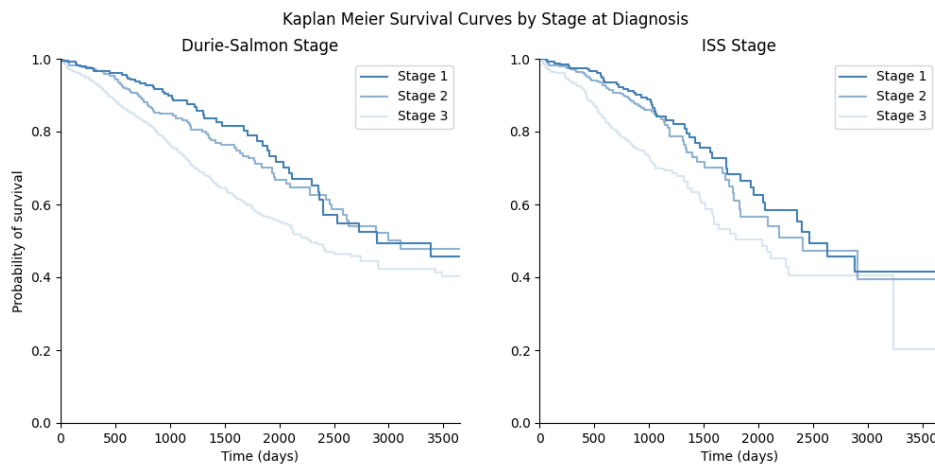


Figure 3.27: Kaplan-Meier Survival curve per stage at diagnosis

value in each variable.

For the variables that indicate the presence or the absence of each genetic mutation we may notice that there is a very high number of missing values. For that reason, a new value will be created for these variables that represents the missing value. Since we are dealing with binary variables that take the values 0 and 1, missing values will be represented as 2.

Due to some considerations made, the remaining variables with missing values will still be dealt with. Firstly, regarding the variables related to the best and any responses, although they have a fairly low number of missing values, we should take into account the importance of this variables, as they may be used as our target variables. The staging system categories, as well as the LDH level, both at diagnosis and treatment, have a very large number of missing values, making imputation with the mean value or the most common value very biased.

The exploratory data analysis presented in this chapter included all the data provided. Nevertheless, in the next chapters, only the data from the first line of treatment was used. This is due to constraints that were found regarding the amount of data available, especially taking into account the number of missing values present.

Table 3.4: Final variables meaning and type

Variable	Meaning	Type
line_number	Treatment line number	int(ordinal)
age_diagnosis	Age at diagnosis	int(ordinal)
age_lot	Age at treatment	int(ordinal)
diag_ldh_serum_num	Serum LDH value at diagnosis	float
ldh_serum_num	Serum LDH value at treatment	float
diag_del13q_cat	Presence of the mutation del13q	int(binary)
diag_del17p_cat	Presence of the mutation del17p	int(binary)
diag_t_4_14_cat	Presence of the mutation t4-14	int(binary)
diag_durie_salmon_cat	Durie-Salmon Stage at diagnosis	int(ordinal)
durie_salmon_cat	Durie-Salmon Stage at treatment	int(ordinal)
diag_iss_cat	ISS Stage at diagnosis	int(ordinal)
iss_cat	ISS Stage at treatment	int(ordinal)
corti	Treatment with corticoids	int(binary)
cyto	Treatment with cytotoxic	int(binary)
Imids	Treatment with immunomodulatory agents	int(binary)
PI	Treatment with proteasome inhibitor	int(binary)
first_line_chemo_fl	Treatment with chemotherapy	int(binary)
auto_sct_fl	Treatment with ASCT	int(binary)
line_duration	Duration of treatment line	int
time_to_next_trt	Interval between treatment lines	int
best_response	Best response registered to treatment	int(ordinal)
any_response	Best response registered to treatment	int(binary)
status_fu	Occurrence of death	int(binary)
status_next_trt	Occurrence of death in the treatment line	int(binary)
time_to_os	Time until death	int

Table 3.5: Variables with missing values

	Missing values	Percentage (%)
corti	52	0.74
Cyto	52	0.74
Imids	52	0.74
PI	52	0.74
best_response	1441	20.38
any_response	1441	20.38
diag_durie_salmon_cat	2018	28.55
diag_iss_cat	4460	63.09
ldh_serum_num	5231	74.00
durie_salmon_cat	5344	75.60
diag_ldh_serum_num	5904	83.52
iss_cat	6022	85.19
diag_del13q_cat	6469	91.51
diag_del17p_cat	6469	91.51
diag_t_4_14_cat	6469	91.51

3.3 First Line of Treatment

The data from the instances corresponding to the first line of treatment were selected for further analysis, due to constraints regarding the missing data.

Table 3.6: Pearson's correlations and mutual information for variable at diagnosis and at the first line of treatment

Feature 1	Feature 2	Correlation	MI
age_diagnosis	age_at_lot	0.99	3.11
diag_ldh_serum_num	ldh_serum_num	0.99	4.86
diag_durie_salmon_cat	durie_salmon_cat	0.88	0.67
diag_iss_cat	iss_cat	0.97	1.04

3.3.1 Relations between diagnosis and first line of treatment measurements

One important aspect is that there is little time between the diagnosis and the first line of treatment. For that reason, with the objective of feature selection, correlations and the mutual information between variables regarding the same information registered at diagnosis and at the first line of treatment were calculated using the sklearn library. For this, the Pearson correlation was used. Out of all the correlations and mutual information values calculated, the most relevant ones are shown in Table 3.3.1.

It is possible to observe the high values of correlation between the age at diagnosis and at the first line of treatment, the measurements of LDH at diagnosis and at the first line of treatment, as well as the stages for both staging systems. Furthermore, the values of mutual information suggest high dependency between these pairs of variables. In order to better understand these correlations, some visualizations are provided.

Figure 3.28 shows that, for most patients, LDH takes the same values at diagnosis and at the first line of treatment. As was mentioned previously, it seems that these values may even correspond to the same measurement, and were registered in both fields.

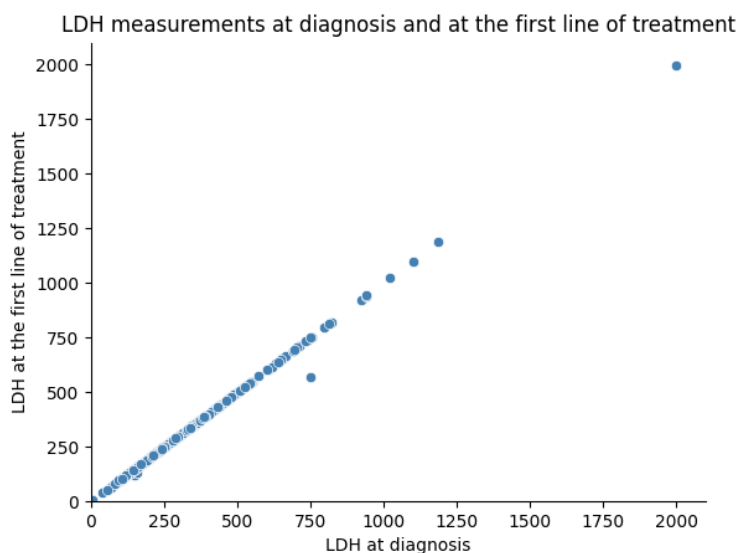


Figure 3.28: Relations between LDH at diagnosis and at the first line of treatment

Figure 3.29 shows two heatmaps with the relation between each stage at diagnosis and at the first line of treatment. It is possible to observe that these values are highly

correlated. At the same time, it is also visible that only very rarely the stage at the first line of treatment is lower than the stage at diagnosis.

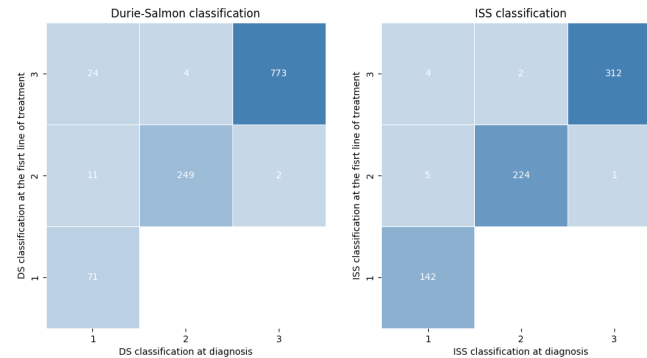


Figure 3.29: Relations between stages at diagnosis and at the first line of treatment

For these reasons, these variables were used for imputation of the missing values, and were then discarded. For the age, since there are no missing values, only the age at diagnosis was considered and the age at the first line of treatment was discarded.

3.3.2 Response to the first line of treatment

Regarding the binary variable containing the response to treatment, 1447 (61.63%) patients are positive responders to the first line of treatment and 543 (23.13%) patients are negative responders, revealing a possibly important class imbalance. The remaining patients have missing values regarding response.

The relation between the response to the first line of treatment and survival was evaluated and the Kaplan Meier curve is shown in Figure 3.30. The log rank test showed there is a significant difference in the survival between these two groups ($p\text{-value}=0.0$).

3.3.3 Missing Values

Table 3.3.3 shows the number and percentage of missing values in each variable for the data regarding the first line of treatment after these modifications. It demonstrates that, even for the first line of treatment, there is still a very large number of missing values that must be addressed.

3.4 Outliers

Since the data from the first line of treatment was selected to be later used for clustering and classification, the presence of outliers was searched for.

Initially, the input continuous variables were analyzed for the presence of outliers. The outliers were defined as

$$outlier < Q1 - 1.5 * IQR \text{ OR } outlier > Q3 + 1.5 * IQR \quad (3.1)$$

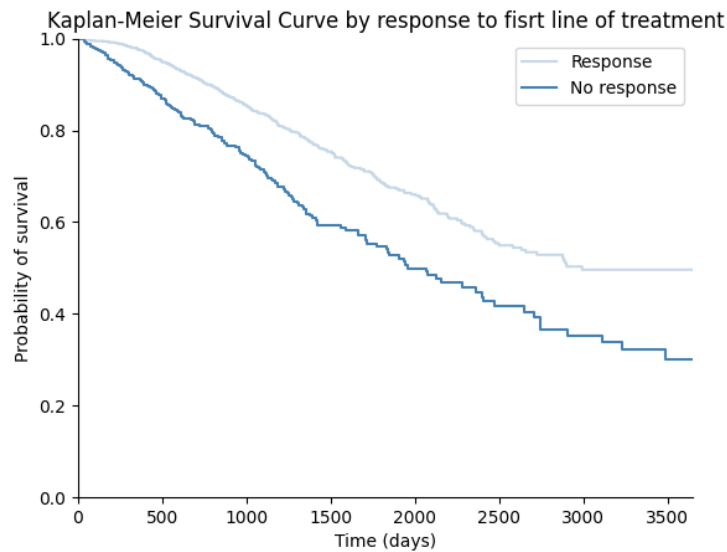


Figure 3.30: Survival per response to first line of treatment

Table 3.7: Variables with missing values on the first line of treatment

	Missing values	Percentage (%)
corti	7	0.29
Cyto	7	0.29
Imids	7	0.29
PI	7	0.29
best_response	414	17.22
any_response	414	17.22
durie_salmon_cat	707	31.28
iss_cat	1435	59.69
ldh_serum_num	1838	76.46
diag_del13q_cat	2095	87.15
diag_del17p_cat	2095	87.15
diag_t_4_14_cat	2095	87.15

Table 3.8: Outliers

Column Name	Min	Q1 - 1.5 * IQR	Q3 + 1.5 * IQR	Max	Outliers
age_diagnosis	26	32	96	92	5
diag_ldh_serum_num	2.39	-105.65	603.02	4425.30	42

where Q1 represents the first quantile, Q3 the third quantile and IQR represented the interquantile range. The results are shown in Table

Regarding age, 5 outliers were found. The age of these patients was comprehended between the value 26 and 32. Despite being younger than the rest of the population, it seems plausible that patients of this age exist. In addition, there are 42 outliers for the LDH values, with LDH values varying from 603.02 and 4421.30. There are no outliers with negative values for both these variables, what would indicate an error in the measurement. As a result of clinical reasoning indicating that the so-called outliers are plausible values rather than data errors, it was decided not to remove or alter them.

3.5 Normalization

The present project aimed at determining the most appropriate normalization method for the data under analysis. The data from the first line of treatment was selected for this step.

After careful consideration, the min-max normalization was selected over the Z-score normalization. Not only were the variables not normally distributed, but the majority of them were also binary or categorical variables, which are simpler to interpret after min-max normalization as opposed to Z-score normalization. The implementation of the min-max normalization was carried out by utilizing the `MinMaxScaler()` class from the `sklearn` library.

The present study employed a scaling technique that was calibrated on the training data in order to find the scaling parameters. Subsequently, these were applied on both the training and test sets in a separated manner. Figure 3.31 synthesizes this process.

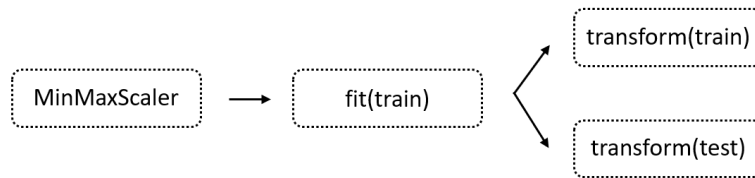


Figure 3.31: Min-Max normalization scheme

When cross-validation was used, the scaling parameters were again calculated using the training set from each fold, and implemented to the train and test set separately.

METHODS, RESULTS AND DISCUSSION

4.1 The Challenge of the Missing Values

As mentioned previously, a major limitation of the data set provided is the sparsity of the data it contains, as there are several variables with a high percentage of missing values. Unfortunately, this is a serious limitation that is frequently present in medical data. In this project, several approaches to this challenge are discussed, implemented and compared.

4.1.1 Imputation

Imputation techniques were employed to address the issue of missing values in our data set. Initially, we explored two simple approaches for the continuous variables: imputing missing values with the mean and imputing a random value within the range of the mean plus and minus the standard deviation. For the categorical variables, imputation with the most common value was implemented.

Upon analyzing the results obtained from our imputation methods, we identified two significant limitations. Firstly, there were features that contained more missing values than observed values. For that reason, they would be represented by a majority of imputed values rather than real observed values. Consequently, the imputed data set lacked real observations, potentially affecting the validity of further analysis. Secondly, a substantial alteration in the distribution of the variables was observed.

The application of dimensionality reduction and cluster methods was conducted on the imputed data, and a comprehensive analysis revealed that the resulting data representation did not yield meaningful outcomes. A concise overview of these methods is provided in Appendix 3, although this particular approach was later abandoned.

Furthermore, the K-nearest neighbors approach for imputation was addressed. This approach was implemented and evaluated on our data set. Several limitations were encountered. Primarily, when attempting to select the features for constructing the multidimensional space, we observed low correlation values for almost all variables. As a result, it was not possible to establish a reliable space for projecting our data points and identifying nearest neighbors based on the assumption that these points would have

similar values in the variable to input than the closest data points. Furthermore, even when considering only the most highly correlated features, certain data points exhibited numerous missing values within these variables. Consequently, precise projection of these points within the new multidimensional space became impossible. Due to these challenges, we made the decision to discontinue the utilization of this approach.

These identified limitations demonstrate the necessity of cautious interpretation and consideration when utilizing imputation techniques to handle missing values, specially when dealing with a large amount of missing data. The implications of imputed values, particularly their potential deviation from true observations and influence on data distributions, must be taken into account to ensure the integrity and reliability of subsequent analyses and findings. We concluded that imputation was not a practical solution for the missing values in our data set.

4.1.2 Removal of Uninformative Instances

Due to the limitations encountered using imputation, alternative approaches were explored. One option was to remove uninformative instances, meaning, those that carry very little information due to the missing values.

One important assumption underlying this method was that the missing values were absent at random. For example, this methodology would be implausible if, for example, we believed that missing information corresponded to tests the physician decided not to perform, as they would not benefit the patient. Nevertheless, the variables contained in this data are crucial to the management of patients with multiple myeloma, and are all mentioned in numerous international guidelines. Therefore, we hypothesized that the information missing in our data corresponded to data the physician had access to, but did not include in the data tables at random.

First, it was determined how many patients were missing how many of these fields. The 7 patients with missing information on the treatment were removed from further analysis, leaving 2397 patients for consideration. For this methodology, 7 clinical variables were considered, which included age at diagnosis, LDH values, the ISS and DS stage, as well as the three mutation variables. Since the age at diagnosis was present for all patients, only the remaining six variables were considered by this methodology.

The Appendix Table D displays the number of patients with the number of missing values per feature. This table shows small decreases in the percentages of missing values for most variables as the patients with higher counts of missing values are removed. This demonstrates that the missing values are distributed across patients, meaning, the majority of patients have some missing values as opposed to a small number of patients having a large number of missing values. It is clear that in order to achieve low percentages of missing values, the vast majority of patients must be eliminated, which would result in a massive loss of data, which is inapplicable to this project.

Moreover, we may notice that the presence of missing values in the mutation variables

is correlated. Consequently, removing patients based on the number of missing values they possess will create a bias in favor of the mutation variables. This is easy to understand as the percentage of missing values in these variables decreases constantly and significantly more rapidly than in other variables.

For these reasons, this method was abandoned. Alternative approaches were explored. Based on the denoising autoencoder concept for imputation, a methodology that would use the autoencoder without imputation was proposed. The details and implementation of this approach are elaborated upon in the subsequent section.

4.1.3 Autoencoder

Inspired by the autoencoder approach outlined in the existing literature, another methodology was implemented. This involved constructing an undercomplete autoencoder to both solve the missing values challenge and for dimensionality reduction, following an approach used in low-rank models [udell2016generalized]. This autoencoder employed a matrix indicating the missing values as a mask for the loss function.

Following normalization of the variables, the missing values were imputed with the value -1, as it was not present in any of the variables and was readily distinguishable. The mask is a matrix with the same dimensions as the original data set. It assigned a value of 0 to any original matrix element equal to -1, representing missing values, and a value of 1 to all other elements.

A customized loss function entailed the sum of the product of the squared error by the mask matrix, as shown in Equation 4.1

$$loss = \sum_{n=1}^N \sum_{m=1}^d (y_{nm} - \hat{y}_{nm})^2 * mask \quad (4.1)$$

where N is the number of observations and d is the number of features, y_{nm} is the observed value in the data set for the n -th observation and m -th feature and $\hat{y}_{nm}$ is the corresponding reconstruction. This methodology permits disregarding the values associated with missing data by omitting the reconstruction error measurement on the missing values.

This loss function was chosen because it is useful for measuring the dissimilarity between the input and output, which should be as similar as possible, it is straightforward to interpret and computationally efficient. However, it is essential to note that it is sensitive to the variables different dimensions, so input data must be normalized.

Additionally, the reconstructed data set obtained by the decoder still has the limitation containing a large number of values that were imputed, rather than the original values. For this reason, the lower-dimension representation of the data obtained by the encoder was utilized for further analysis, as is shown in Figure 4.1.

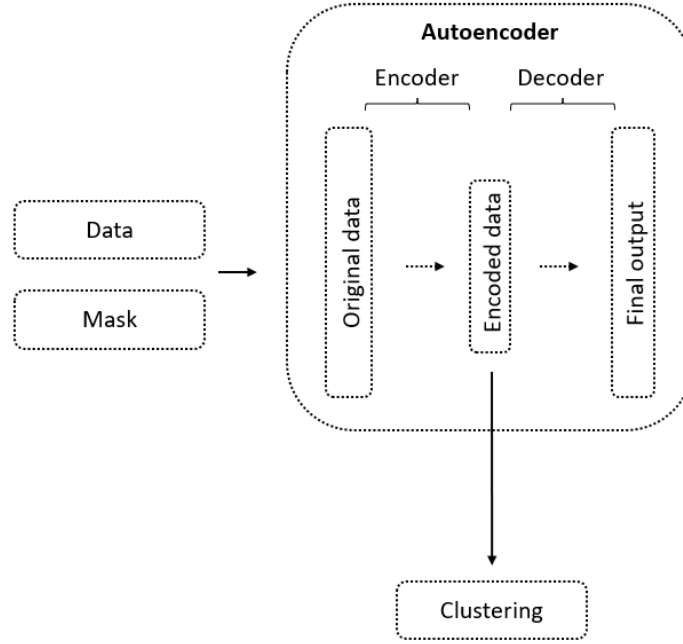


Figure 4.1: Autoencoder schema

The major advantages of this approach is that not only do we not use imputed values, but the data representation also has a lower dimension, eliminating the need for other dimensionality reduction techniques.

In terms of implementation, the Keras library was used as it offers a simple, high-level API to create deep learning models, which is efficient and can be easily combined with pandas and numpy.

Regarding the general architecture, as is suggested in the literature, a symmetric encoder and decoder were implemented.

The KerasTuner was utilized for hyperparameter optimization due to its easy-to-configure framework. A random train and validation sets were used. The chosen search algorithm was Hyperband. The He Normal initializer was selected. It draws samples from a truncated normal distribution centered on 0 and with a standard deviation corresponding to $\sqrt{\frac{2}{\text{number_input_units}}}$. Included among the parameters to select was the optimizer. The "adam", "sgd" and the "rmsprop" were compared. Also, the number of nodes and activation function were compared for each dense layer, including 'relu', 'sigmoid' and 'tanh', except for the last dense layer of the encoder, for which a 'linear' activation function was chosen. The remaining parameters, such as the learning rate, were selected manually.

After the hyperparameters were selected and the autoencoder was trained, it was used to encode the entire dataset and generate a new 3-dimensional dataframe.

This methodology was applied in two different settings, as is explained below.

4.1.4 Encoding All Data

Initially, the autoencoder was used on the data containing both clinical and the treatment variables. This aimed to demonstrate how to encode a complete data set with missing values in order to acquire a representation with fewer dimensions and no missing values. Also, analysis of the results will show some particularities regarding this method.

Figure E.1 in the Appendix illustrates the value of the loss function during training of the autoencoder. This is useful because it allows us to identify significant aspects of the training process. For examples, a too rapid learning rate resulting in an erratic curve.

For visualization purposes, the data points were projected using autoencoder results. An initial visual analysis of the results was conducted.

For this purpose, plots were created depicting the distribution of the values of the original variables in this new encoded space. Figure 4.2 shows how the treatment variables are distributed. To facilitate comprehension, the axes of some plots may be differently positioned.

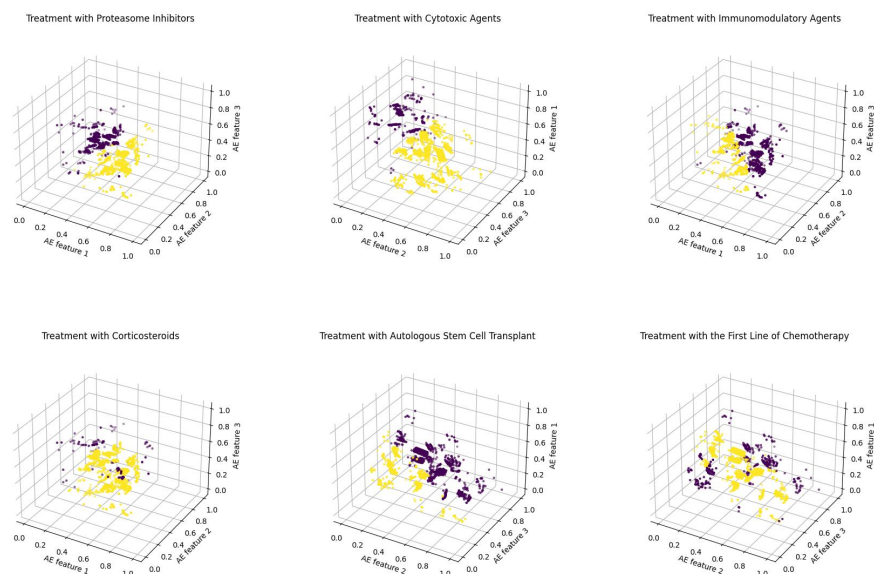


Figure 4.2: Data representation obtained with the autoencoder showing the values of the treatment variables.

Observably, the new arrangement of the data points separates the different values for most treatment variables. The exceptions include the corticoid therapy, with which the vast majority of patients were treated, resulting in a significant imbalance between the values of this variable; and the flag for the first line of chemotherapy.

On the other hand, it is more difficult to distinguish the values of the clinical variables

in this new space. Figure 4.3 displays the variables associated with the staging systems. It demonstrates that the three phases present in each staging system cannot be separated.

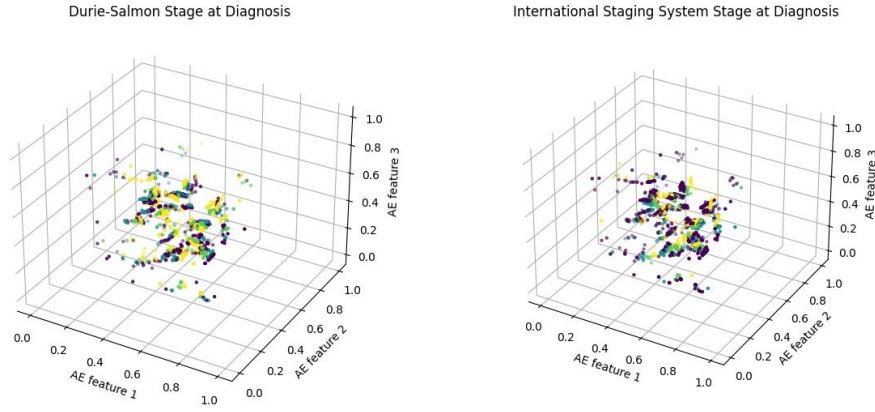


Figure 4.3: Data representation obtained with the autoencoder showing the values of the staging systems' variables

Additionally, several variables, such as the measurement of the LDH values and the mutations, have a large number of missing values. Figure 4.4 shows the data points that have values for the mutations. The points with missing values are missing from this representation. It is possible to see that the presence and absence of the mutations are not well separated in space.

We may conclude from this experiment, as is quite empirical to understand, that larger number of missing values will limit the amount of contribution a feature will have in the new representation of the data.

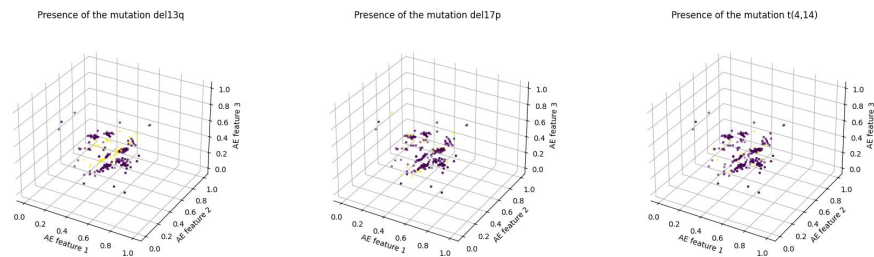


Figure 4.4: Data representation obtained with the autoencoder showing the values of the mutation variables without the missing values

4.1.4.1 Clustering Encoded Data

The data obtained with the usage of this autoencoder was later clustered. For this, the representation obtained with the autoencoder was first normalized. Both DBSCAN and hierarchical clustering were applied and the results were compared. These methods were used due to the data structure: there seems to be no well-defined, spherical clusters, and, for that reason, K-means was excluded.

The epsilon parameter for the DBSCAN was selected using two distinct methodologies. Firstly, the 10 nearest neighbors of each point were identified and the distances from each point to all its nearest neighbors were calculated and sorted in ascending order. Then, we examine the point where there is an elbow, which indicates a distance beyond which the nearest points are very distant, i.e. zones of low density of instances. Figure E.2 in the Appendix shows these sorted distances. This point of inflection is visible to be somewhere close to 0.06. However, this technique has the drawback of being overly subjective. Therefore, a different technique was employed to confirm the results.

The second method consisted of finding the hyperparameters that would maximize the Silhouette Score. Epsilon values between 0.01 and 0.2 were tested and the Silhouette score was calculated. The results indicated that an epsilon of 0.07 would yield the highest Silhouette score. This value was utilized to create the clusters.

This value of epsilon was used and it led to the formation of twenty clusters, as is shown in Figure 4.5.

On the other hand, for the implementation of the hierarchical clustering, the linkage method and the number of clusters had to be specified. First, for the selection of the linkage method, the most common linkage methods were tested, including "single", "complete", "average", "weighted", "centroid", "median" and "ward". The method that permitted greater distances between clusters was chosen. The "ward" method was chosen to generate a dendrogram in order to determine the best number of clusters, again, defined as those that guarantee larger distances between clusters. Figure E.3 in the Appendix shows the dendrogram produced, showing the best number of clusters to guarantee more separability was 2.

Using these values, agglomerative hierarchical clustering was implemented, using the Scikit Learn library, and the results are displayed in Figure 4.6.

Table 4.1.4.1 comparison of the results obtained using DBSCAN and hierarchical clustering.

	DBSCAN	Hierarchical
Number of clusters	20	2
Silhouette score	0.59	0.58
Calinski-Harabazs score	820.37	730.08
Davies-Bouldin score	1.79	1.70

Using DBSCAN, a large number of clusters is obtained. In addition, the majority of clusters contain fewer than one hundred patients. In comparison to the hierarchical clustering results, the Silhouette score and the Calinski-Harabazs are greater for the DBSCAN

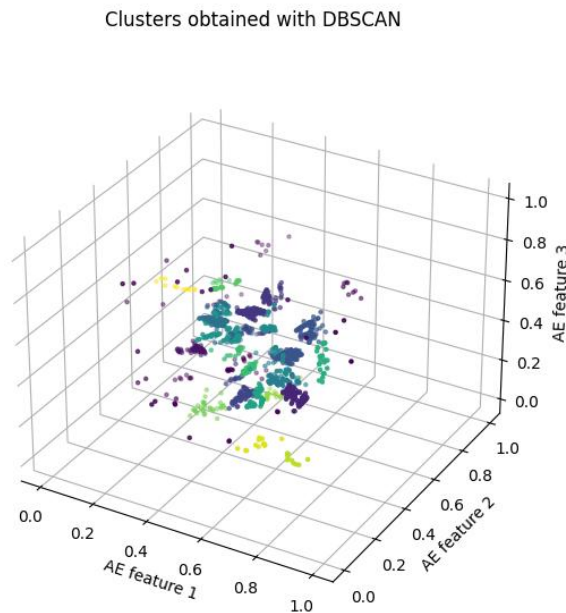


Figure 4.5: Projection of the data using the autoencoder results showing the clusters obtained with DBSCAN

clusters, indicating superior cluster separation. Nevertheless, the Davies-Bouldin score is lower for the hierarchical clustering, indicating the clusters created are less similar to one another. It is possible to realize that different scoring methods will consider different methods as the best method to apply. Furthermore, the number of clusters obtained using DBSCAN is too great to enable good comparisons. For that reason, the hierarchical clusters were selected for further evaluation.

Differences between the two clusters were searched for. It was observed that all patients who were not receiving cytotoxic treatment were placed in cluster 2, along with a few who were receiving it. Additionally, the majority of patients (1145 out of 1232) taking a proteasome inhibitor were assigned to this cluster. The differences in the clinical variables between the two clusters were not so clear. This again demonstrated that the treatment fields were more relevant in the construction of the new encoded variables. This was to be anticipated, as these features had no missing values, whereas the others had up to more than 90%.

In accordance with these findings, we decided to construct new data encodings that separately incorporate clinical and treatment data.

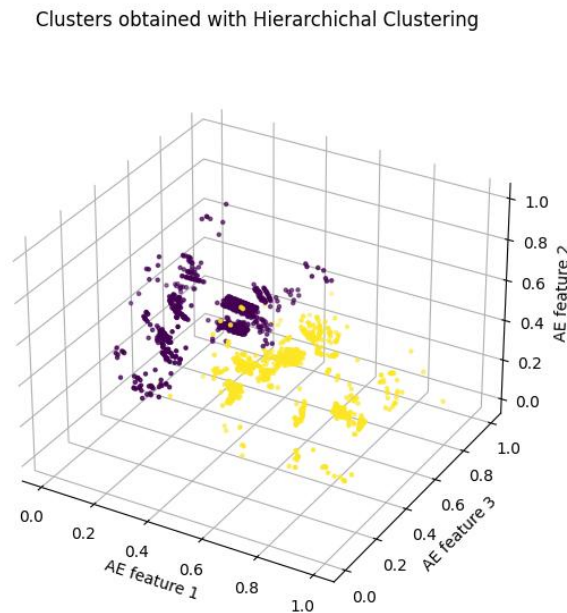


Figure 4.6: Projection of the data using the autoencoder results showing the clusters obtained with Hierarchical Clustering

4.2 Defining Sub-phenotypes of Patients

Two newly-tuned autoencoders were trained separately in order to be used to encode both the clinical and treatment data. The objectives included trying to find sub-phenotypes based only on the clinical data and determining if a relationship could be drawn between the clinical and treatment aspects.

The same methodology shown in the previous chapter was used to encode the data and find the best clusters. The results are shown in Figure 4.7. Two clusters were formed, cluster 1 with 1039 patients (yellow) and cluster 2 with 1358 patients (purple). The Silhouette score was 0.54, the Calinski-Harabasz score was 1794.14 and the Davies-Bouldin score was 1.35.

In order to understand these two clusters, a comparison was made regarding the clinical variables, the treatment variables and the outcome variables. The continuous variables were compared in both clusters using the Mann-Whitney U test and the categorical ones were compared using the Chi-Squared test. The results are shown in table 4.2 and 4.2. It is possible to see that there is a statistically significant difference between the stages in both clusters, both by the Durie-Salmon and the ISS staging systems. Also, there is a significant difference between the presence of the deletion 13q and translocation 4-14. There is no significant difference in the remaining clinical variables, as well as in the treatment variables, except for the flag for the first line of chemotherapy.

Clusters obtained with Hierarchical Clustering

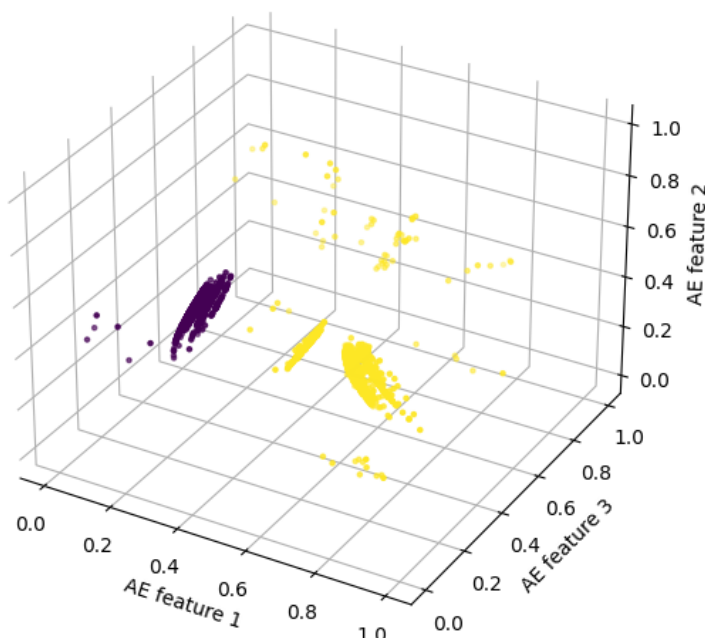


Figure 4.7: Results of hierarchical clustering on the features obtained using the autoencoder on the original clinical variables. Cluster 1 is shown in purple and cluster 2 is shown in yellow.

Table 4.1: Comparison of the continuous variables for the two sub-phenotypes encountered using the Mann Whitney U Test.

Variable	p-value
Age at Diagnosis	0.38
LDH value	0.45

These differences were further evaluated. All of the patients in cluster 1 had a Durie-Salmon stage III, as is shown in Figure 4.8 (yellow). Regarding the ISS stage, both cluster have patients from the three stages. Nevertheless, in cluster 1, the proportion of patients in stage III is higher than in cluster 2 and the proportion of patients in stage I is lower in cluster 1 than in cluster 2.

Regarding the mutations, all the patients with the mutation del13q and translocation 4-14 are assigned to cluster 2. Figure 4.9 illustrates the presence of these mutations (yellow), the absence (blue) and the missing values (purple).

Finally, differences in the response to treatment and survival were evaluated. First, the Qui-squared test was used. It showed a statistically significant difference in survival (p-value: 0.0) and the absence of a difference regarding the response to treatment (p-value: 0.14). The Kaplan-Meier curve is shown in Figure 4.10. The log rank test showed a significant difference in survival in both groups (p-value: 0.0).

Table 4.2: Comparison of the categorical variables for the two sub-phenotypes encountered using the Chi-squared test.

Variable	p-value
DS stage	0.0
ISS stage	0.0
Del13q mutation	0.0
Del17p mutation	0.55
T(4,14) mutation	0.0
Corticosteroids	0.35
Cytostatic	0.99
Immunomodulatory agents	0.54
Proteasome inhibitor	0.72
Auto-transplant	0.82
First-line chemotherapy	0.02

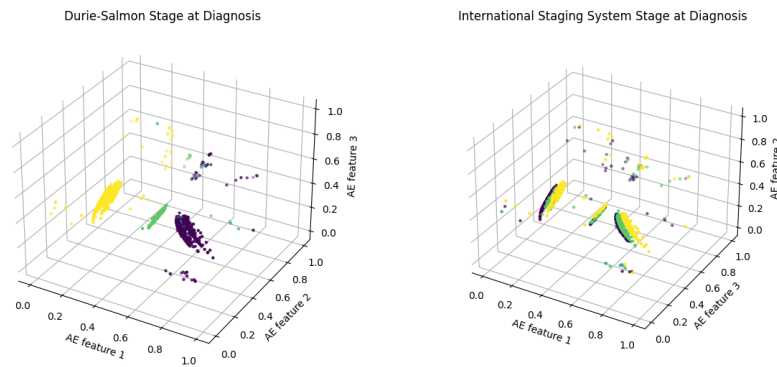


Figure 4.8: Stages shown in the autoencoder representation of the data. Stage I-yellow, Stage II-green, Stage III-purple.

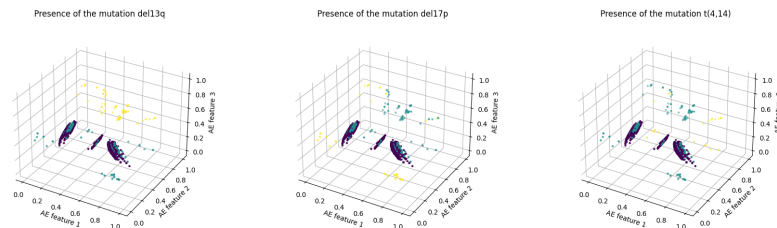


Figure 4.9: Mutations shown in the autoencoder representation of the data. Presence-yellow, Absence-green.

Due to the inability to calculate the time to best response due to lack of data, this comparison was not made.

4.2.1 Relationship Between Sub-phenotypes and the Treatment

For this second objective, the clinical and treatment data sets were encoded and clustered separately, as is shown in Figure 4.11. Then, the clusters obtained were compared.

The Chi-Square test was performed and showed no correlation between the clinical and the treatment clusters. (p-value= 0.52). There is insufficient evidence to conclude that there is a significant association between clusters obtained from clinical data and those derived from treatment data. One hypothesis is that the clinical variables available in this

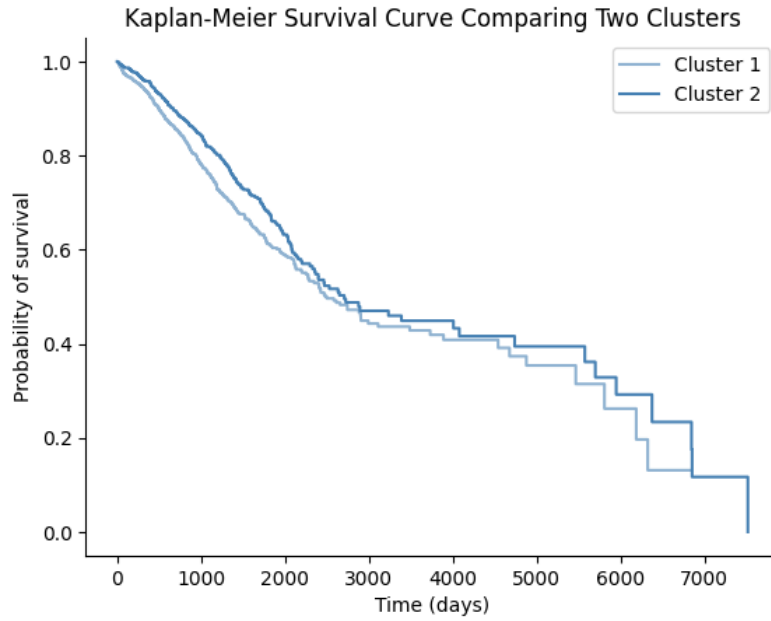


Figure 4.10: kaplan meier in ae

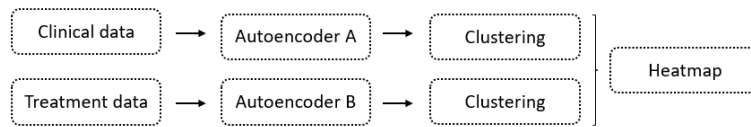


Figure 4.11: AE clinical vs treatment schema

data set did not contain all the information the physicians had at the time they selected the treatment, and that they therefore made informed decisions based on information we lack. On the other hand, this may have occurred due to the fact that this data was collected from a variety of hospitals and countries, and the guidelines, availability of treatments, and preferences of some hospitals or physicians may have caused this difference.

The most important message from this analysis is the fact that there is crucial clinical information missing in this data that was probably available to the physicians, influencing the treatment choices.

4.3 Classification

To complete the classification assignment, it was also necessary to deal with the challenge of the missing values. The imputation proved to be ineffective, as well as removing the most important features. The concept of using an autoencoder to obtain new representations of the features appeared promising, but several significant limitations were encountered. The autoencoder operates as a "black box", and the features obtained cannot be easily interpreted.

In the context of medical data, interpretation of a classifier is crucial. For that reason, the autoencoder was not used to produce representations of the data for the classification task. It was necessary to employ a different methodology: the patterns of missing values was created and will be explained further below.

This methodology was chosen so that it was possible to train a number of classifiers and give examples of the results obtained, although the preferred approach would have been to train a classifier that would work for all the patients and features. For that reason, the methodology used will be described so it can be replicated in a data set that does not have the missing values constraints.

4.3.1 Patterns of Missing Values

The patterns of missing values were defined as the recurring arrangements of missing values for each patient. For this, the clinical and treatment variables were taken into account and the total count of patients associated with each pattern was determined. Table D in the appendix presents the obtained results.

The identified patterns were utilized to establish patient groups that shared common fields of missing values. These fields could then be excluded. As previously stated, the age at diagnosis and the treatment variables did not present any missing values. Thus, the focus was on combinations of the remaining clinical variables.

Emphasis was placed on the most prevalent patterns, as being those that are more frequent. To allow the study of patients with genetic mutations, as these were split into patterns with small numbers of patients, a new and simplified pattern was established, that involved joining patients who possessed the information on these mutations into one group. This approach was adopted due to the substantial proportion of missing values associated with these variables, which would not yield sufficiently large groups to be analyzed independently.

To make the most of this methodology and ensure larger groups of patients, we added all patients who could be included in a pattern, by disregarding additional fields they may have possessed. This process is outlined in Figure 4.12.

The final groups, along with their composition, are displayed in Table 4.3.1. Besides the features in this table, the variables include the age at diagnosis and the treatment variables all the patterns.

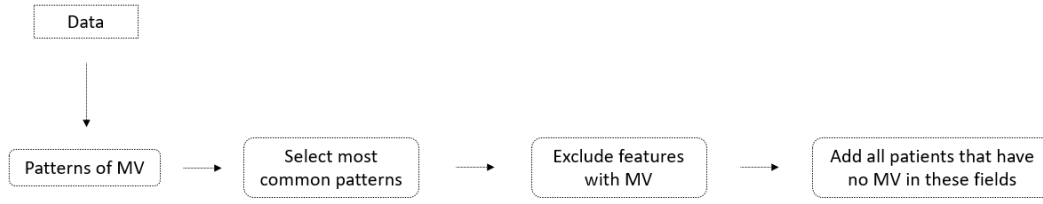


Figure 4.12: Selection of Missing Values Patterns

Table 4.3: Final Patterns of Missing Values. The age at diagnosis and the treatment variables are present for all groups.

Group	ldh_serum_num	del13q_cat	del17p_cat	t_4_14_cat	durie_salmon_cat	iss_cat	Count	Total Features
0	0	0	0	0	1	0	1436	8
1	0	0	0	0	1	1	469	9
2	1	0	0	0	1	0	465	9
3	0	0	0	0	0	1	854	8
4	1	0	0	0	1	1	191	10
5	0	0	0	0	0	0	1987	7
6	1	0	0	0	0	1	361	9
7	0	1	1	1	0	0	258	10

The groups were analyzed separately. Both the global and the localized approach were used, as is shown in Figure 4.13. The global approach included using all the patients in one group for the classification directly. For the localized approach, subgroups of patients were selected using clustering techniques, followed by the training of a classifier for each clusters. For analysis of the classifiers obtained, the results were compared with the baseline results. Furthermore, for model interpretation, feature ranking was obtained by using the Shap value and feature permutation, among others.

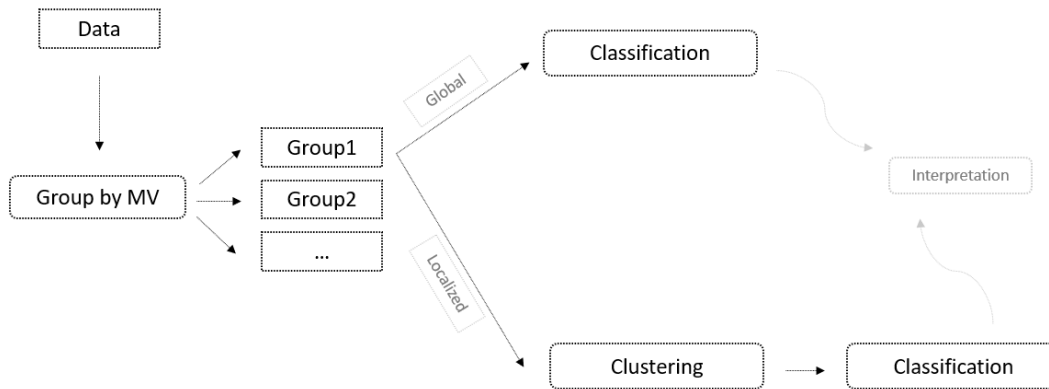


Figure 4.13: Methodology by group

4.3.1.1 Limitations of the Patterns of Missing Values

By creating these groups with patterns of missing values, it was possible to create groups without missing values. Nevertheless, the amount of clinical variables in each group

decreased substantially. It is important to mention that most of our variables are categorical, and most are even binary, guaranteeing a limited amount of possible different feature values combinations. Also, it is important to notice that we are using the treatment classes, instead of the treatments themselves, which also contributes to more patients having the same pattern of input variables.

For that reason, the presence of duplicates and incoherent instances was investigated. Patients were counted as duplicates if there was at least one other patient that presented exactly the same clinical and treatment features. Incoherent instances was defined as duplicate patients for which there is at least another duplicate with a different label for the response to treatment. Table 4.4 shows the number of duplicates and incoherent examples for each group.

Table 4.4: Duplicates and incoherent instances for each subgroup

Group	Size	Duplicates	Incoherent
0	1436	617	362
1	469	54	48
2	465	0	0
3	854	261	194
4	191	0	0
5	1987	1321	479
6	361	0	0
7	258	52	33

It is possible to notice that there are no duplicates or incoherent patients in groups 2, 4 and 6. This is understandable, as these 3 groups include the only continuous variable, which decimal points, the value of LDH.

This constituted a limitation as it is not possible that the algorithms used learn correctly from these examples. Also, this proved that there is crucial information missing. It is possible to conclude that there are features missing that would help distinguish the reason why a patients responds or not to the treatment.

4.3.2 Classification

One goal of this project was to determine the feasibility of creating a classifier that would predict treatment response, while ensuring its clinical utility. For this, the most useful metric had to be discussed.

The objective was to predict the response to treatment in patients already diagnosed with multiple myeloma that were certainly undergoing treatment. Therefore, the major challenge is to ensure that the given treatment is effective. Identifying treatments that are ineffective and understanding the reason of the failure in response holds less significance in this context, as the clinician would still have to chose an alternative treatment. Concluding, the main purpose of this classifier is to minimize the probability of administering a treatment to a patient that does not yield the desired response. For that reason, the metric chosen to evaluate and compare our models was the precision on the "responders to treatment" class.

A nested cross-validation approach was implemented to train the classification algorithms, as is shown in Figure 4.14. This methodology allowed exploration of multiple algorithms with varying hyper-parameters and different subsets of our data, in order to evaluate how the different algorithms behave in data such as ours. This approach was repeated 3 times for each algorithm, guaranteeing the creation of 15 trained models.

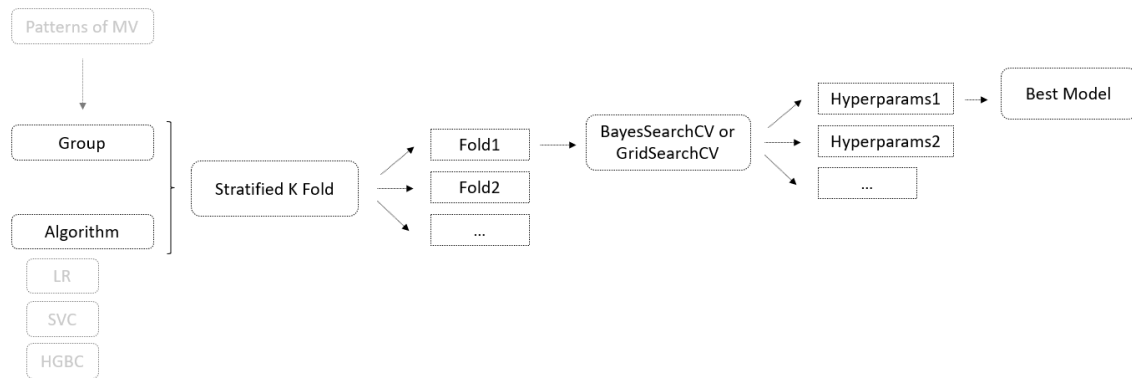


Figure 4.14: Nested Cross Validation Schema

In summary, sub-sets were created within our data, for which the best parameters of each algorithm were found, and, then, after training each model, their performance was compared.

Stratified K-fold was applied to guarantee a similar proportion of examples of each class in each fold and the training data. The data was shuffled and a total of 5 folds were selected, in order to obtain a validation set with 20% the size of the data in each fold. Random seeds were used.

In order to determine the best hyper-parameters for our models in each fold, two methods were utilized, depending on the classification algorithm chosen. In the context of SVC, the decision was made to employ the BayesSearchCV algorithm. In contrast to GridSearchCV, wherein all potential parameter values are exhaustively explored, this methodology involved sampling a fixed number of parameter settings from the specified distributions. For the remaining algorithms, the GridSearchCV was employed. A 5-fold stratified cross validation was applied in both methods, and the parameter chosen to optimize was yet again the precision on the positive response.

For the algorithms that provided this feature, the `class_weight` was set to "balanced" to ensure that the algorithm would address the imbalance issue by reducing bias towards the majority class and enhancing overall performance.

For each trained model, the precision was calculated. The model with the best precision for each algorithm was chosen to be evaluated regarding feature importance.

First, the global approach was applied, where each classifier was trained with the data from all the patients from each group. Then, the results were compared with the localized

approach. The results in the next sections are all obtained in group 7, the group of patients that underwent genetic testing.

4.3.3 Global Approach

Several machine learning algorithms were compared in terms of overall performance and interpretability. Initially, more transparent models were used, including logistic regression and SVM with a linear kernel. It was not included in this demonstration, but trying the SVM with other kernels might improve the performance of the models. Hist-GradientBoostingClassifier was then implemented to exemplify the usage of an ensemble method.

These models were trained 3 times using StratifiedCV with five folds, yielding 15 models for each algorithm. The precision score was retrieved for all the models and the boxplot comparing these values for group 7 is shown in Figure 4.15.

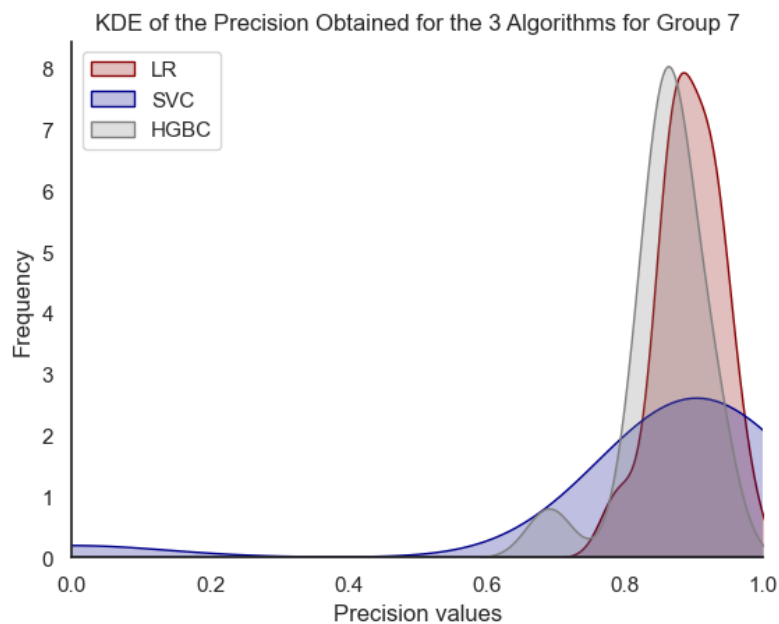


Figure 4.15: Precision KDE

The model with the best precision for each algorithm was selected to assess the general results and feature ranking. Since the model was selected based on the error on the validation set and we did not secure a test set, it was not possible to calculate an unbiased estimate of the error of our models. This was decided as we consider more important to describe the methodology than demonstrating final error results, as the limitations in these data make it impossible to retrieve useful results.

For each algorithm, there was one model that obtained a precision score of 1. These were selected for further analysis.

The confusion matrix for the logistic regression model is shown in Table 4.5. Although, as expected since the precision obtained was 1, there are no false positives, it is possible

to notice the large number of false negatives encountered in these confusion matrices.

Table 4.5: Confusion matrix for the best logistic regression model. PN-predicted negatives, PP-predicted positives, RN-real negatives, RP-real positives

	PN	PP
RN	8	0
RP	23	21

Due to this classification results, with a large number of false negatives, alternative metrics were studied. It is clear that minimizing the number of false positives is our major goal, but we must also bear in mind the importance of a useful classification that does not have too much false negatives, that would lead to changing the treatment to too much patients, possibly to more aggressive treatments that would have a higher burden in the patients health.

For this reason, the F0.5 score was chosen. The same process as the one described previously were repeated and the results have shown that, with a small increase in the number of false positives, the number of false negatives could be highly decreased. Table 4.6 shows these results for the best model representing each algorithm.

Table 4.6: Confusion matrix for the best logistic regression model. PN-predicted negatives, PP-predicted positives, RN-real negatives, RP-real positives

	PN	PP
RN	5	2
RP	11	33

Nevertheless, we may observe that we shifted from a model that could correctly identify the combination of clinical and treatment characteristics that would lead to a positive result to a model where it is possible to find false positives. We consider there is a significant difference between the error regarding the false positives and false negatives. Concluding, for this case, we believe it is worth it to have and increase in false negative so that we may have a precision of 1.

4.3.4 Localized Approach

The localized may be useful to separate patients in subgroups in which a local classifier could have better performance than a global one. The initial idea of this project was to use the sub-phenotypes obtained using the clustering techniques in the first part of this project to later train a classifier for each of the subgroups. Nevertheless, as was explained previously, using the encoded data would lead to the creation of models that were not possible to interpret. Using the clusters obtained in the first part with all the data directly would bias our validation set. For that reason, a different methodology was used: clustering the patients in each k-fold, separating the training from the validation set. The results were similar to those obtained by the global algorithms. Nevertheless, we believe it would be interesting to make a thorough comparison once another more complete data

set is provided. There, the unbiased test error could be calculated and compared between the global and the localized approach.

4.4 Model Interpretation and Feature Importance

Explainability is crucial in the medical field to provide us insight on several aspects of a model. Multiple methodologies were applied to guarantee thorough analysis of feature importance in the models in each group. For the linear models, the first approach included interpretation of the coefficients attributed by the model to each feature, which are easily accessible for these algorithms. Their interpretation can already give insights on how the model is behaving regarding each feature.

Furthermore, there are multiple ways of getting interpretations of these models, as was explained previously. For feature ranking, the permutation importance analysis, the Shap values and the partial dependence plots were applied to the training set. As an example, the plots obtained for these three different methodologies will be displayed for one model trained on group 7.

4.4.1 Feature Ranking by Feature Permutation

First, the feature importance was evaluated using feature permutation, meaning, the influence of permutation of the values in a single feature while the other features do not vary was evaluated. The model performance metric used was the precision. The results are shown in Figure 4.16. This plot shows the feature importance ranking. The most important features include treatment with auto-transplant, the age at diagnosis, the treatment with chemotherapy and the presence of the mutation del13q. We may conclude that the random permutation of values in these variables would lead to worst precision. Moreover, this process would not affect the model performance if the permutation was performed on the remaining features.

4.4.2 The Shap Values

The Shap values evaluation summary plot results are shown in Figure 4.17. It is possible to notice the feature ranking given by the order the features appear on the plot. The feature ranking is the same as that obtained with the feature permutation. Treatment with auto-transplant is the most important feature as it takes the higher mean Shap value. Also, it is noticeable in this case that most features have low importance.

Finally, it is even possible to understand how different values of some features influence the way the model is classifying the points. For example, for the treatment with auto-transplant, we may notice that positive values will lead the model to classify the instance as a positive responder. Also, the lower and higher values of age at diagnosis seem to lead the model to classify the instance as non-responder, although this evaluation is limited due to the subjective evaluation of this plot. This method has the advantage of providing an

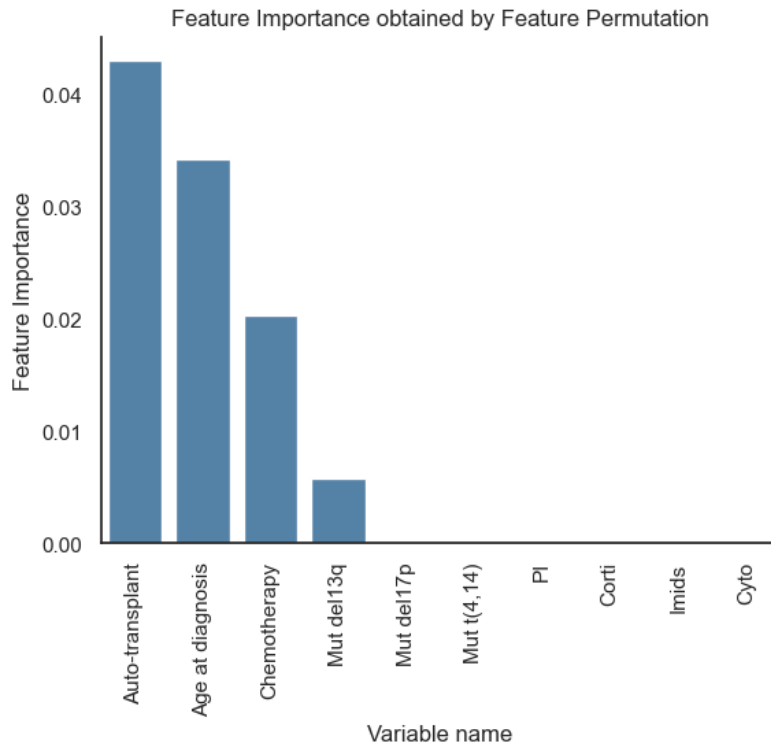


Figure 4.16: Feature Ranking with Feature Permutation for Group 7

explanation on how the model is working for each training instance, represented by one point in the plot.

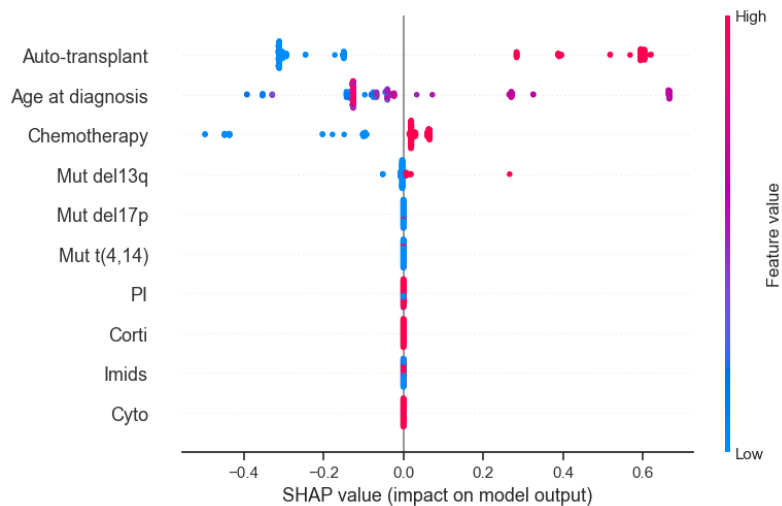


Figure 4.17: Feature Ranking with Shap Values for Group 7

Figure 4.18 shows the Shap values for one particular instance. This serves only as an example, but this evaluation might be useful in further analysis, as it will help understand the cause of the correct or incorrect classification in particular examples.

It is also possible to obtain a plot that shows the Shap value for a given variable for each variable value. This might add important information to the summary plot, as this is

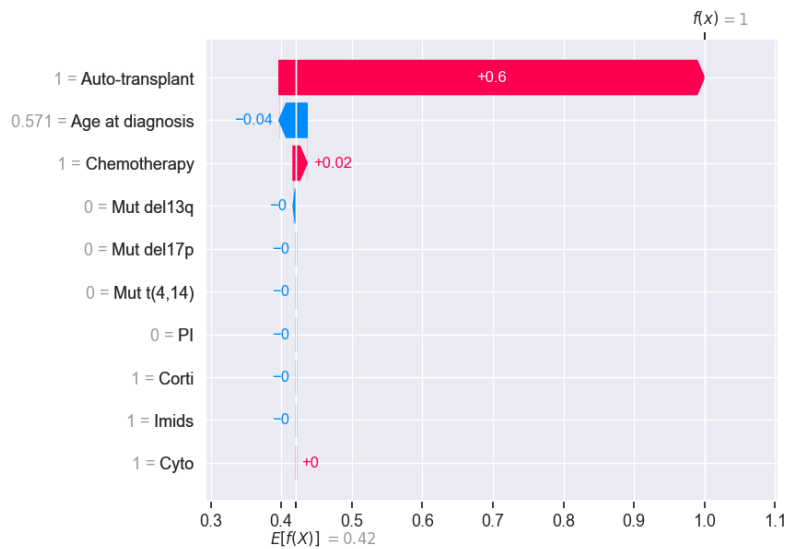


Figure 4.18: Shap Values for one particular instance

only represented by the color of each point. Two examples are shown in Figure 4.19 and 4.20. In Figure 4.19, it is possible to observe that intermediate values of age at diagnosis result in higher and positive Shap values, while lower and higher values of age lead to negative Shap values.

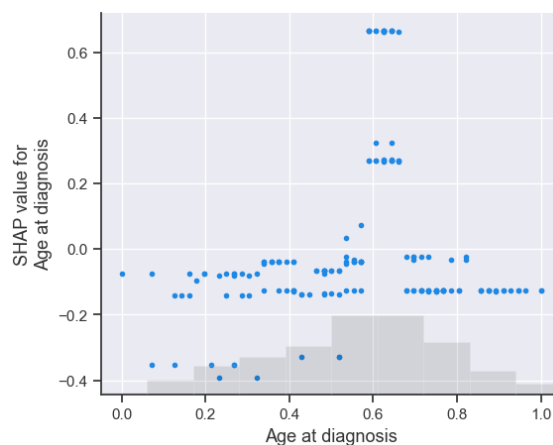


Figure 4.19: Shap value per normalized value of age at diagnosis

Figure 4.20 shows the absence of a transplant is associated with lower Shap values, while the presence of transplant is associated with lower Shap values. Nevertheless, this example shows a relation that was already clear in the summary plot in Figure 4.17.

The Shap interaction plot is shown in Figure 4.21. It is possible to notice the most important interactions are between the features with the highest rankings. Nevertheless, surprisingly, the variables "Imids"(treatment with immunomodulatory agents) shows interaction with itself and the treatment with chemotherapy and the auto-transplant. This particular example shows how some interactions may limit our analysis of feature importance and further enhances the need of using multiple methodologies to make better

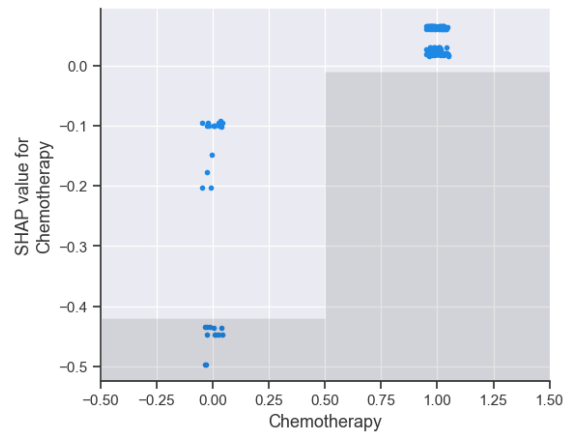


Figure 4.20: Shap value per normalized value of auto-transplant

conclusions on the behavior of classification models.

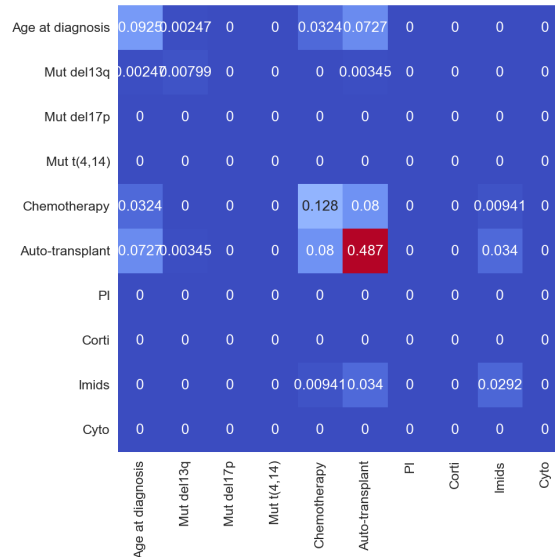


Figure 4.21: Shap Interaction Plot

4.4.3 Partial Dependence Plots and Individual Conditional Expectation

The partial dependence plots for the same model are shown in Figure 4.22. With this plot, it is easier to understand how the model classification varies with the continuous values of age at diagnosis. Again, the influence of chemotherapy and auto-transplant are visible in the plots. Again, the influence of the immunomodulatory agents is uncovered by these plots, although it was not visible with most the previous methods.

Finally, figure 4.23 shows the partial dependence plot and the individual conditional expectation for the age at diagnosis for the same model, elucidating the behaviour of the model in an instance level.

Using multiple methods may facilitate the discovery of nonlinear relationships between variables, thereby revealing important features that would otherwise be deemed less

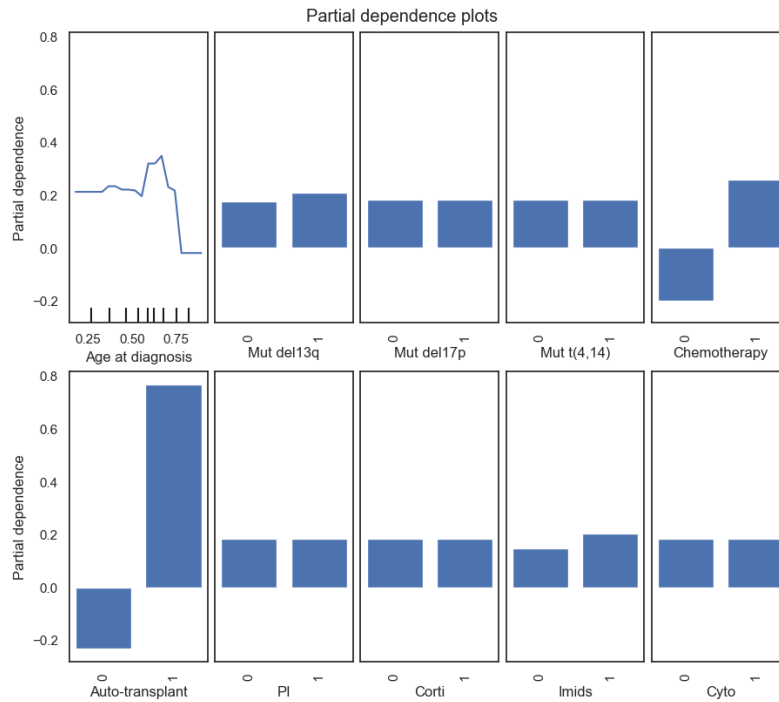


Figure 4.22: Feature Ranking with Partial Dependence Plots

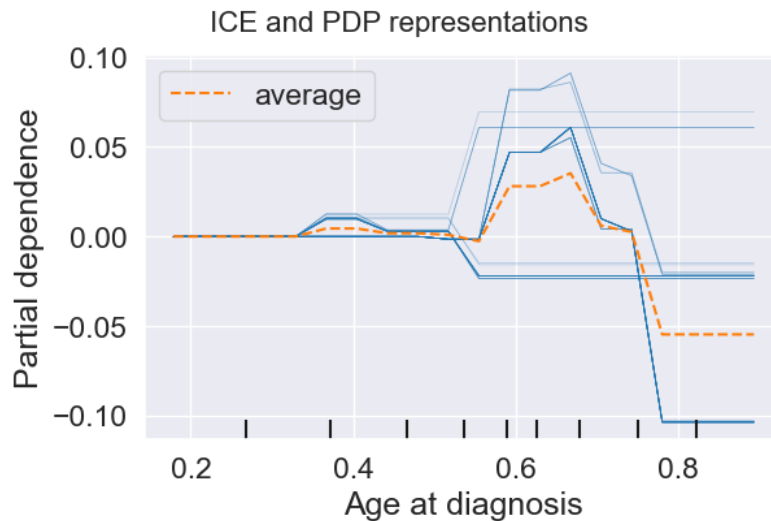


Figure 4.23: Partial Dependence Plot and Individual Conditional Expectation for age at diagnosis.

important. Also, these methods provide different perspectives on the feature importance.

4.4.4 Global and Localized Learning

Regarding the local approach, an interesting evaluation would be to compare the feature ranking obtained in each sub-phenotype, as well as comparing them with the results obtained in the global approach. It is expected that the feature ranking, specially regarding the clinical features, is different in sub-phenotypes that differently express the disease.

The Figures [F.1](#) and [F.2](#) in the Appendix show the feature ranking obtained by the Shap values for two models trained in two clusters from group 7.

4.5 Limitations

Regarding the challenges found during this project, the most important one was that of missing information. This is a common limitation when using medical data. Medical data collection is costly, as it needs physicians annotations; it is also sparse, as it might be difficult to gather information from different hospitals and even countries and some information is still not in digital format; it is difficult to manage due to ethical constraints, attributable to the indispensable anonymization. For example, the variable containing the sex of the patients in our dataset was randomized due to questions raised regarding anonymization.

Firstly, due to the large amount of missing values, which is a problem that was addressed and proved to be very hard to deal with. Also, clinical knowledge led us to believe there are other clinical information missing that is very important in the context of managing patients with multiple myeloma. Examples include the presence of the most common clinical manifestations of the disease, such as the presence of anemia or renal failure. This suspicion was confirmed by the lack of correlation between the clinical variables and the treatment variables.

One other limitation encountered in this study was the presence of incoherent instances, which refers to patients who share identical clinical and treatment variables but exhibit different labels for their response to treatment. For example, in group 0, 25.2% of all patients were found to be incoherent. This inconsistency could be attributed to the considerable amount of missing clinical data, resulting from both the limited original clinical variables and the presence of missing values within the available variables.

Additionally, in order to ensure an adequate number of patients for meaningful comparisons, the study employed treatment classes rather than individual treatments. This guarantees the presence of patients with similar treatment variables patterns, although it further contributes to an increased occurrence of identical input variable patterns among patients and the presence of incoherent instances.

These limitations have implications for the learning capabilities of the algorithms used in the study. The presence of incoherent instances makes it challenging for the algorithms to accurately learn from these examples, potentially affecting the overall performance and reliability of the results.

Regarding the selected methodologies, specific obstacles were identified. Using an autoencoder to avoid imputation appeared to be a promising method, but new challenges arose. The creation of uninterpretable features hindered their use in classification tasks as it was considered that interpretability is a key aspect for classifiers in the medical field.

Lastly, due to the limited amount of data, it was impossible to conduct an analysis that incorporated the time-dependent progression of patients. Due to patients dying or

dropping out of the study, the data for the remaining lines of treatment is even sparser than for the first.

Due to these obstacles, it was not possible to draw practical conclusions that would have contributed to the management of multiple myeloma patients. Nonetheless, useful information can be derived from the methodologies employed and utilized in subsequent research. In light of this, the following chapter provides a summary of the procedures and recommendations for dealing with data sets with characteristics similar to the ones found here.

RECOMMENDED PROCEDURES

The provided data set had several limitations, including a large number of missing values and the absence of crucial clinical characteristics. Nevertheless, this thesis contains valuable information on procedures that may be applied to the data. In light of this, the purpose of this chapter is not to draw conclusions from the supplied data set, but rather to provide a set of procedures that may be useful when dealing with similar data that is more comprehensive.

5.1 Preprocessing

First, data cleansing and preprocessing is necessary.

For this, several steps may be necessary. Figure 5.1 shows the first steps of this process.

In a data set containing a large number of variables, the first and most crucial stage is to determine which are irrelevant to the objectives of the study. Variables with only missing values are obviously useless, but variables with a single value for all patients also lack interest for clustering and classification. Also, variables that are directly derived from others, such as discretized variables derived from continuous ones, may be discarded as they contain repeated information.

Then, it is essential to differentiate between continuous and categorical variables. For continuous variables, it is crucial to comprehend their distributions. This can be accomplished with simple calculations, such as the mean or median, but visualizations, such as histograms and KDE plots, are also very useful. It is essential to evaluate the presence and nature of outliers, and decide whether they should be removed.

Before addressing categorical variables, it is necessary to determine their nature. The division of categorical variables into ordinal and nominal variables will facilitate their management.

First, it is reasonable to assign each class to an integer using ordinal encoding for ordinal variables. However, it is essential to recognize that if the intervals between each category are not uniform, this method may not be suitable.

In contrast, nominal variables may be encoded with one-hot encoding. For each value

in the original variable, a new binary variable is created that contains the value 1 if and only if the corresponding value is present in the original variable. In the specific case where the values of a nominal variable correspond to a list representing the combination of a limited number of unique values, the formation of variables for each of these values helps reduce the number of new variables produced by one-hot encoding.

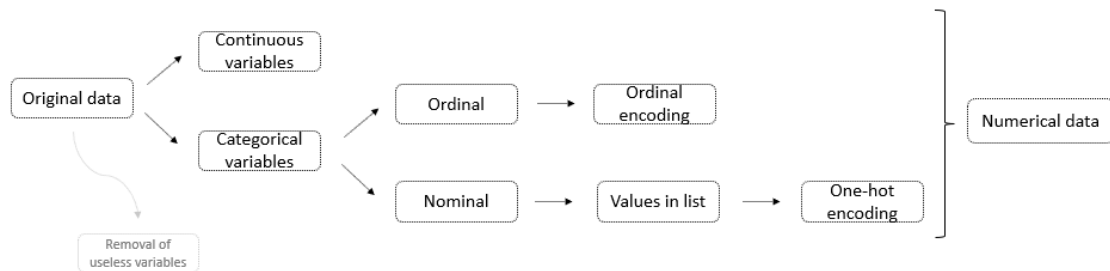


Figure 5.1: Recommendations for data preprocessing: how to encode the original variables

One other interesting aspect this data set included was the fact that there were measurements made over time. There are variables that represent measurements made at the time of diagnosis and at each line of treatment. Nevertheless, clinical knowledge may provide important information on how these features might vary or not in time.

The presence of genetic mutations is one example in our data of something that ought to remain the same over time. Additionally, it is reasonable for patients to undergo only one genetic test. The data confirms this as there are no inconsistencies between the results of genetic tests conducted at different times on the same patient.

Consequently, this information could be used to generate a single variable that contains this unique information for the patient, regardless of whether the test was performed at the time of diagnosis or during one of the treatment lines, as is shown in Figure 5.2.

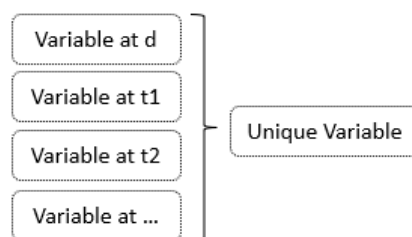


Figure 5.2: Recommendations for data preprocessing: how to handle information that is constant over time for a patient but is dispersed across multiple variables and observations | d-time of diagnosis; t1-time of the first line of treatment; t2-time of the second line of treatment.

This method is useful for data sets that present data in time, as it prevents the loss of information, reduces the number of missing values for important information and helps reducing the number of original features.

Furthermore, when the number of missing values is elevated several limitations might occur. One is the inability to study the patients over time, as not only they have a lot of missing values right in the first line of treatment, but also, with the number of patients diminishing in each treatment line, the amount of information for subsequent lines of treatment is limited. For that reason, it was decided to work only with the first line of treatment.

Knowing that we have information regarding the diagnosis and the first line of treatment, it is important to find correlations between variables that pertain to the same information at these different times, as the time interval may not be very large and the values registered may pertain to the same value.

One example is that of the blood tests, such as was proven with the LDH value, for which the measurement made at diagnosis were used also to fill the values for the first line of treatment, as probably new blood tests had not been done yet.

First, the correlation between these pairs of variables can be calculated for this purpose. If the correlation shows to be high, to clarify the nature of the correlation, scatter plots for continuous variables and heatmaps for categorical variables can be created. In the case where the present variables show to have the same values for the observations in which both values are recorded, we might assume values from the same measurement where used to complete both variables. For that reason, two important steps might be taken. First, the values obtained in the first line of treatment for patients that had missing values at diagnosis might be used to fill in these missing values. Then, it is possible to proceed to the elimination of the variable corresponding to the first line of treatment. This helps with dimensionality reduction but also with diminishing the number of missing values. This procedure is shown in the schema present in Figure 5.3

After cleansing the variables, the treatment of their values is crucial. Normalization is essential for improved algorithmic performance, and selecting the optimal normalization technique is essential. The min-max normalization offers a number of benefits for datasets containing predominantly categorical or binary variables.

5.2 Autoencoder to solve the missing values limitation

The presence of a large number of missing values is a common but very limiting challenge. We propose a method of dealing with the missing values that does not use imputation, but rather uses an autoencoder that uses a matrix indicating the missing values as a mask for the loss function. The lower-dimension representation of the data obtained by the encoder part of the autoencoder might be used for further analysis. As shown before, Figure 4.1 provides a schema of this process.

The architecture and parameters of the autoencoder might be selected using a hyperparameter optimization process, both manually or using an already implemented tuner.

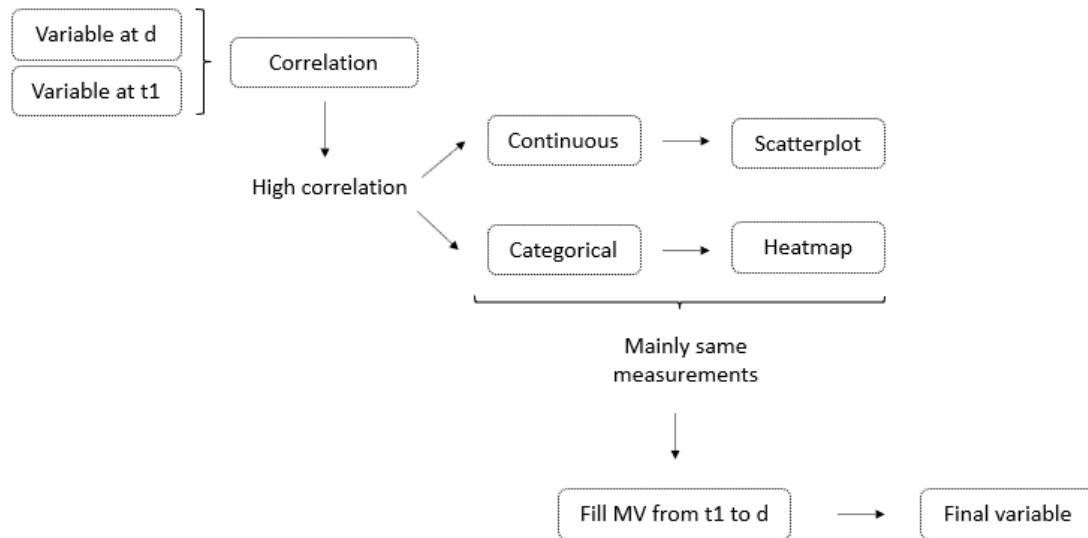


Figure 5.3: Recommendations for data preprocessing: how to handle information that is constant over time for a patient but is dispersed across multiple variables and observations | d-time of diagnosis; t1-time of the first line of treatment; MV-missing values.

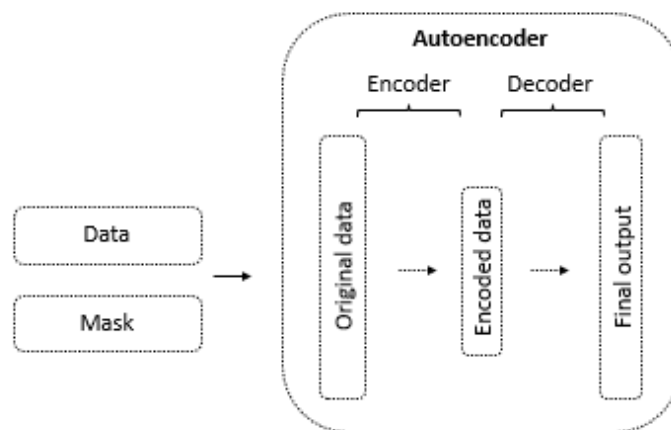


Figure 5.4: Autoencoder schema

We recommend that the number of encoded features is either 2 or 3, for easy graphical representation.

5.3 Sub-phenotypes

As shown in Figure 5.5, the autoencoder can be used to obtain a representation based solely on the clinical characteristics. The encoded data can then be clustered, and the resulting clusters can be studied to determine the characteristics of various clinical phenotypes. Also, distinctions in treatment, response to treatment, and survival can be evaluated.

Various cluster analysis algorithms can be utilized. Data visualization may aid in

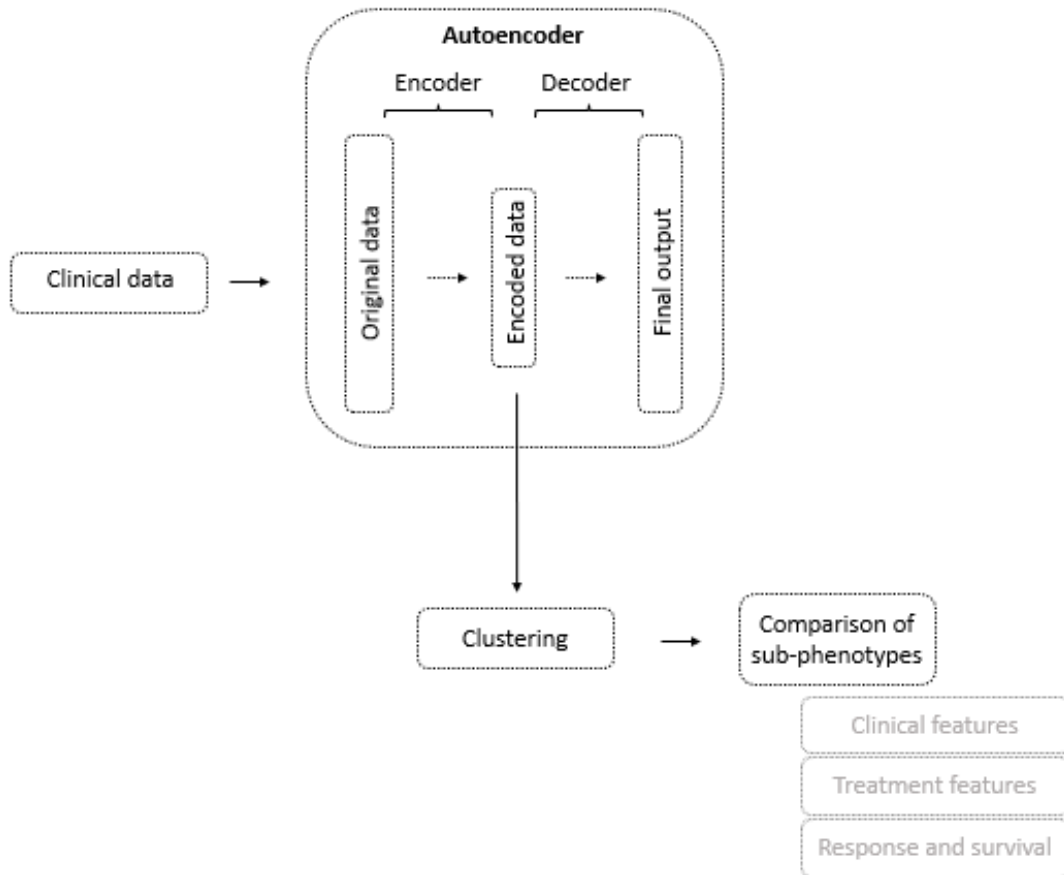


Figure 5.5: Autoencoder schema

selecting the most effective methods. For instance, K-means was not used with our data because the clusters did not have a spherical shape. Hierarchical clustering appeared to perform better with our data. DBSCAN may also be beneficial in the presence of outliers and densely populated areas, although finding a reasonable number of clusters was not possible with our data.

5.4 Prediction of response to treatment

The initial challenge that must be evaluated is the presence of class imbalance, as it may limit the classifier's performance. We propose a method that eliminates the need for oversampling or undersampling. First, when constructing folds for cross validation, stratified K-fold must be used to ensure that the proportion of instances from each class is the same in both the training and validation sets. An additional crucial step is to determine if the algorithms already incorporate a "balanced" mode, which must be activated so that the algorithm gives instances from imbalanced classes different weight.

Regarding classification algorithms, we propose beginning with simple, linear algorithms that are easier to interpret. Then, more complicated algorithms could be implemented. Specifically, ensemble methods may aid in performance enhancement.

Choosing which metric should be maximized is an essential aspect of classification. Precision is the ideal metric to maximize when attempting to predict whether a combination of clinical and treatment variables will result in a positive response.

5.5 Model interpretation and feature importance

It is essential to interpret the models and provide evidence that they employ a credible relationship between the variables and the response. We believe that multiple methods are required for a comprehensive interpretation. Feature permutation, the Shap values and the Partial Dependence plots and the Individual Conditional Expectation are methods that provide insights on feature importance.

We suggest a first feature ranking using feature permutation, which gives a general idea on the importance of each feature. This values can latter be confirmed with the following methods. Adding the Shap values provides several advantages: it is possible to evaluate feature importance at an instance level, it is even possible to understand how the values of some features influence the way the model is classifying the points; and Shap interaction plots may elucidate relations between variables. Finally, the partial dependence plots give a easy to understand visualization of how the model behaviour changes with the varying value of each variable, while the individual conditional expectation provides instance level explanations.

CONCLUSIONS

The provided data presented a number of challenges, including lack of essential information. This was mainly due to the large number of missing values in the available features and the absence of other important features. Lack of information is a common limitation in data science, and specifically in the medical field. Medical data collection is costly, sparse and difficult to manage. For these reasons, the problems encountered with this data set are not uncommon and must be dealt with.

Some insights on multiple myeloma patients subphenotypes are outlined. Three of the genetic abnormalities associated with multiple myeloma were studied. Although one of the high risk mutations, deletion 17q, in fact determined a worse survival outcome in our cohort, this was not true with the translocation (4;14). Also, the intermediate risk deletion 13p did not determine a worse survival outcome in our patients.

This dissertation contains valuable information and discussion on procedures that can be applied to other data sets with comparable limitations. Using an autoencoder, a procedure is defined for dealing with a large number of missing values without resorting to imputation. This new representation of the data may be used for clustering in the future in an attempt to identify patient subphenotypes. Due to the requirement of creating interpretable classifiers, it is not possible to train classifiers using these encoded features. Therefore, a missing value pattern aggregation strategy is recommended. In addition, the importance and implementation of methods that ensure model interpretability and transparency are discussed.

We hope that this thesis is not only beneficial for providing methodologies for additional data processing, but also for raising awareness of the importance of collecting more and better data in the medical field, in order to guarantee the creation of reliable and transparent models.

BIBLIOGRAPHY

- [1] N. Abiri et al. “Establishing strong imputation performance of a denoising autoencoder in a wide range of missing data problems”. In: (2020-04). DOI: [10.1016/j.neucom.2019.07.065](https://doi.org/10.1016/j.neucom.2019.07.065). URL: <http://arxiv.org/abs/2004.02584><http://dx.doi.org/10.1016/j.neucom.2019.07.065> (cit. on pp. 6, 12).
- [2] O. Y. Al-Jarrah et al. “Efficient Machine Learning for Big Data: A Review”. In: *Big Data Research* 2 (3 2015), pp. 87–93. ISSN: 22145796. DOI: [10.1016/j.bdr.2015.04.001](https://doi.org/10.1016/j.bdr.2015.04.001) (cit. on p. 15).
- [3] A. Allegra et al. “Myeloma Diagnosis , Prognosis , and Treatment Selection”. In: (2022), pp. 1–16 (cit. on pp. 1, 3, 6–8, 30, 39, 40).
- [4] E. Alpaydin. *Introduction to Machine Learning Second Edition*. Vol. 1107. 2014, pp. 105–28. ISBN: 9780262012430. DOI: [10.1007/978-1-62703-748-8_7](https://doi.org/10.1007/978-1-62703-748-8_7). URL: <http://www.ncbi.nlm.nih.gov/pubmed/24272434> (cit. on pp. 20, 21, 23, 24, 27, 28, 30–34).
- [5] A. A. Alyousef et al. “Nearest Consensus Clustering Classification to Identify Subclasses and Predict Disease”. In: *Journal of Healthcare Informatics Research* 2 (4 2018), pp. 402–422. ISSN: 2509498X. DOI: [10.1007/s41666-018-0029-6](https://doi.org/10.1007/s41666-018-0029-6) (cit. on pp. 18, 30).
- [6] A. Y. A. Amer. “Localized Least Squares Support Vector Machines with Application to Weather Forecasting”. In: (August 2016 2016). DOI: [10.13140/RG.2.2.21331.48167](https://doi.org/10.13140/RG.2.2.21331.48167) (cit. on pp. 15–18, 31–33, 35).
- [7] A. Y. A. Amer et al. “A Localized Learning Approach Applied to Human Activity Recognition”. In: *IEEE Intelligent Systems* 36 (3 2021), pp. 58–71. ISSN: 19411294. DOI: [10.1109/MIS.2020.2964738](https://doi.org/10.1109/MIS.2020.2964738) (cit. on pp. 15, 19).
- [8] R. Andrinandrasana et al. “A Review of Activation Function for Artificial Neural Network”. In: *SAMI 2020 • IEEE 18th World Symposium on Applied Machine Intelligence and Informatics* (2020) (cit. on pp. 25, 26).

-
- [9] F. Anowar, S. Sadaoui, and B. Selim. "Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE)". In: *Computer Science Review* 40 (2021), p. 100378. ISSN: 15740137. DOI: [10.1016/j.cosrev.2021.100378](https://doi.org/10.1016/j.cosrev.2021.100378). URL: <https://doi.org/10.1016/j.cosrev.2021.100378> (cit. on pp. 20, 23).
 - [10] D. Bank, N. Koenigstein, and R. Giryes. "Autoencoders". In: (2020-03). URL: <http://arxiv.org/abs/2003.05991> (cit. on pp. 24, 25).
 - [11] V. Belle and I. Papantonis. *Principles and Practice of Explainable Machine Learning*. 2021-07. DOI: [10.3389/fdata.2021.688969](https://doi.org/10.3389/fdata.2021.688969) (cit. on p. 37).
 - [12] I. Benjamin et al. *Cecil Essentials of Medicine*. 9th (cit. on p. 2).
 - [13] B. Bischl, J. Schiffner, and C. Weihs. "Benchmarking local classification methods". In: *Computational Statistics* 28 (6 2013), pp. 2599–2619. ISSN: 09434062. DOI: [10.1007/s00180-013-0420-y](https://doi.org/10.1007/s00180-013-0420-y) (cit. on pp. 15–18).
 - [14] C. M. Bishop. *Pattern Recognition and Machine Learning*. 2006. ISBN: 9781119130536 (cit. on pp. 27, 30, 32).
 - [15] L. Bottou and V. Vapnik. "Local Learning Algorithms". In: *Neural Computation* 4 (6 1992), pp. 888–900. ISSN: 0899-7667. DOI: [10.1162/neco.1992.4.6.888](https://doi.org/10.1162/neco.1992.4.6.888) (cit. on pp. 15–17).
 - [16] H. Cheng, P. N. Tan, and R. Jin. "Localized support vector machine and its efficient algorithm". In: *Proceedings of the 7th SIAM International Conference on Data Mining* (2007), pp. 461–466. DOI: [10.1137/1.9781611972771.45](https://doi.org/10.1137/1.9781611972771.45) (cit. on pp. 16, 18, 19).
 - [17] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machine and Other Kernel-based Learning Methods*. Ed. by C. U. Press. Press, Cambridge University, 2000. ISBN: 9780521780193 (cit. on p. 35).
 - [18] L. Deng and D. Yu. *Deep learning: Methods and applications*. 2013. DOI: [10.1561/2000000039](https://doi.org/10.1561/2000000039) (cit. on pp. 24, 25).
 - [19] K. Dietz et al. *Statistics for Biology and Health*. 2000, p. 549. ISBN: 9780387874579 (cit. on pp. 13, 14).
 - [20] M. A. Dimopoulos et al. "Multiple Myeloma: EHA-ESMO Clinical Practice Guidelines for Diagnosis, Treatment and Follow-up". In: 2 (December 2020 2021) (cit. on pp. 1, 2).
 - [21] M. Dong and J. Wu. "Localized support vector machines for classification". In: *IEEE International Conference on Neural Networks - Conference Proceedings* (2006), pp. 799–805. ISSN: 10987576. DOI: [10.1109/ijcnn.2006.246766](https://doi.org/10.1109/ijcnn.2006.246766) (cit. on p. 19).

- [22] A. B. Downey. *Think Stats Exploratory Data Analysis in Python*. Ed. by G. T. Press. Version 2. 2014. ISBN: 9781449307110. URL: <http://books.google.com/books?hl=en&lr=&id=TCfZ7d6skT4C&oi=fnd&pg=PR5&dq=Think+Stats&ots=LxYK9ntvRZ&sig=UchH878Uf04hPLGPmOwnNHBGmdk> (cit. on pp. 9, 10).
- [23] B. G. Durie and S. E. Salmon. "A clinical staging system for multiple myeloma correlation of measured myeloma cell mass with presenting clinical features, response to treatment, and survival". In: *Cancer* 36 (3 1975), pp. 842–854. ISSN: 10970142. DOI: [10.1002/1097-0142\(197509\)36:3<842::AID-CNCR2820360303>3.0.CO;2-U](https://doi.org/10.1002/1097-0142(197509)36:3<842::AID-CNCR2820360303>3.0.CO;2-U) (cit. on p. 2).
- [24] N. Eisemann, A. Waldmann, and A. Katalinic. "Imputation of missing values of tumour stage in population-based cancer registration". In: *BMC Medical Research Methodology* 11 (2011). ISSN: 14712288. DOI: [10.1186/1471-2288-11-129](https://doi.org/10.1186/1471-2288-11-129) (cit. on pp. 7, 11).
- [25] A. Farswan et al. "AI-supported modified risk staging for multiple myeloma cancer useful in real-world scenario". In: *Translational Oncology* 14 (9 2021), p. 101157. ISSN: 19365233. DOI: [10.1016/j.tranon.2021.101157](https://doi.org/10.1016/j.tranon.2021.101157). URL: <https://doi.org/10.1016/j.tranon.2021.101157> (cit. on pp. 3, 7, 9, 11).
- [26] A. B. Fassas and G. Tricot. "Chromosome 13 deletion/hypodiploidy and prognosis in multiple myeloma patients". In: *Leukemia and Lymphoma* 45 (6 2004-06), pp. 1083–1091. ISSN: 10428194. DOI: [10.1080/10428190310001642710](https://doi.org/10.1080/10428190310001642710) (cit. on p. 2).
- [27] I. M. Foundation. *International Myeloma Foundation - Durie Salmon Staging*. 2023-02 (cit. on p. 2).
- [28] J. H. Friedman. 999 REITZ LECTURE GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE 1. 2001 (cit. on pp. 37, 38).
- [29] A. Goldstein et al. "Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation". In: (2013-09). URL: <http://arxiv.org/abs/1309.6392> (cit. on p. 39).
- [30] C. Goswami et al. "Staging system to predict the risk of relapse in multiple myeloma patients undergoing autologous stem cell transplantation". In: *Frontiers in Oncology* 9 (JULY 2019), pp. 1–9. ISSN: 2234943X. DOI: [10.3389/fonc.2019.00633](https://doi.org/10.3389/fonc.2019.00633) (cit. on pp. 3, 6–8, 20, 27).
- [31] R. W. Grant et al. "Use of Latent Class Analysis and k-Means Clustering to Identify Complex Patient Profiles". In: *JAMA Network Open* 3 (12 2020), pp. 1–13. ISSN: 25743805. DOI: [10.1001/jamanetworkopen.2020.29068](https://doi.org/10.1001/jamanetworkopen.2020.29068) (cit. on pp. 39, 40).
- [32] P. R. Greipp et al. "International staging system for multiple myeloma". In: *Journal of Clinical Oncology* 23 (15 2005), pp. 3412–3420. ISSN: 0732183X. DOI: [10.1200/JCO.2005.04.242](https://doi.org/10.1200/JCO.2005.04.242) (cit. on p. 2).

- [33] M. Gönen and E. Alpaydin. "Localized algorithms for multiple kernel learning". In: *Pattern Recognition* 46 (3 2013), pp. 795–807. ISSN: 00313203. DOI: [10.1016/j.patcog.2012.09.002](https://doi.org/10.1016/j.patcog.2012.09.002) (cit. on p. 17).
- [34] M. Gönen and E. Alpaydin. "Localized multiple kernel learning". In: *Proceedings of the 25th International Conference on Machine Learning* (2 2008), pp. 352–359. DOI: [10.1145/1390156.1390201](https://doi.org/10.1145/1390156.1390201) (cit. on pp. 15, 19).
- [35] M. A. Hearst. "Support Vector Machines (SVMs)". In: *IEEE Intelligent Systems* 7 (1998), pp. 85–105. ISSN: 21976511. DOI: [10.1007/978-3-319-06938-8_5](https://doi.org/10.1007/978-3-319-06938-8_5) (cit. on p. 35).
- [36] A. Henelius et al. "A peek into the black box: Exploring classifiers by randomization". In: *Data Mining and Knowledge Discovery* 28 (5-6 2014), pp. 1503–1529. ISSN: 13845810. DOI: [10.1007/s10618-014-0368-8](https://doi.org/10.1007/s10618-014-0368-8) (cit. on pp. 37, 38).
- [37] K. Hongyu, V. L. M. Sandanielo, and G. J. de Oliveira Junior. "Análise de Componentes Principais: resumo teórico, aplicação e interpretação Principal Component Analysis: theory, interpretations and applications". In: *ES Engineering and Science* 5 (1 2016), pp. 83–90. DOI: [10.18607/ES20165053](https://doi.org/10.18607/ES20165053) (cit. on p. 21).
- [38] K. Huang et al. *Machine Learning - Modeling Data Locally and Globally*. Zhejiang University Press, Hangzhou and Springer-Verlag GmbH Berlin Heidelberg, 2008. ISBN: 9787308058315 (cit. on p. 15).
- [39] A. J. Izenman. "Introduction to manifold learning". In: *Wiley Interdisciplinary Reviews: Computational Statistics* 4 (5 2012), pp. 439–446. ISSN: 19395108. DOI: [10.1002/wics.1222](https://doi.org/10.1002/wics.1222) (cit. on p. 23).
- [40] J. J. F. AS, and L. D. Loscalzo J. eds. Kasper DL Hauser SL. *Harrison's Principles of Internal Medicine*. 2nd. 2018, pp. 793–802 (cit. on pp. 1–4).
- [41] P. Kalyani. "Approaches to Partition Medical Data using Clustering Algorithms". In: *International Journal of Computer Applications* 49 (23 2012), pp. 7–10. DOI: [10.5120/7941-1102](https://doi.org/10.5120/7941-1102) (cit. on pp. 7, 27).
- [42] I. M. Kapetanovic, S. Rosenfeld, and G. Izmirlian. "Overview of Commonly Used Bioinformatics Methods and Their Applications". In: *New York Academy of Sciences* (2004) (cit. on p. 35).
- [43] J. Kishore, M. Goel, and P. Khanna. "Understanding survival analysis: Kaplan-Meier estimate". In: *International Journal of Ayurveda Research* 1 (4 2010), p. 274. ISSN: 0974-7788. DOI: [10.4103/0974-7788.76794](https://doi.org/10.4103/0974-7788.76794) (cit. on pp. 13, 14).
- [44] D. G. Kleinbaum and M. Klein. *Statistics for Biology and Health Survival Analysis A Self-Learning Text Third Edition*. URL: <http://www.springer.com/series/2848> (cit. on p. 13).
- [45] T Kohonen. *Self-Organization and Associative Memory*. 3rd. Springer-Verlag Berlin, 1989, pp. 119–157. ISBN: 9783540513872 (cit. on pp. 22, 23).

- [46] A. Kraskov, H. Stögbauer, and P. Grassberger. “Estimating mutual information”. In: *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics* 69 (6 2004), p. 16. ISSN: 1063651X. DOI: [10.1103/PhysRevE.69.066138](https://doi.org/10.1103/PhysRevE.69.066138) (cit. on p. 10).
- [47] S. Kumar et al. “International Myeloma Working Group consensus criteria for response and minimal residual disease assessment in multiple myeloma”. In: *The Lancet Oncology* 17 (8 2016-08), e328–e346. ISSN: 14745488. DOI: [10.1016/S1470-2045\(16\)30206-6](https://doi.org/10.1016/S1470-2045(16)30206-6) (cit. on p. 106).
- [48] G. N. Le et al. “Current and future biomarkers for risk-stratification and treatment personalisation in multiple myeloma”. In: *Molecular Omics* 15 (1 2019), pp. 7–20. ISSN: 25154184. DOI: [10.1039/c8mo00193f](https://doi.org/10.1039/c8mo00193f) (cit. on p. 3).
- [49] S. learn. *HistGradBoostClassifier*. URL: <https://scikit-learn.org/stable/modules/ensemble.html#lightgbm> (cit. on p. 36).
- [50] R. D. Ledesma, P. Valero-Mora, and G. Macbeth. “The Scree Test and the Number of Factors: a Dynamic Graphics Approach”. In: *The Spanish journal of psychology* 18 (June 2015), E11. ISSN: 19882904. DOI: [10.1017/sjp.2015.13](https://doi.org/10.1017/sjp.2015.13) (cit. on p. 21).
- [51] J. M. Lourenço. *The NOVAthesis L^AT_EX Template User’s Manual*. NOVA University Lisbon. 2021. URL: <https://github.com/joaomlourenco/novathesis/raw/main/template.pdf> (cit. on p. ii).
- [52] S. M. Lundberg, P. G. Allen, and S.-I. Lee. *A Unified Approach to Interpreting Model Predictions*. URL: <https://github.com/slundberg/shap> (cit. on p. 38).
- [53] S. Marsland. *Machine Learning An Algorithmic Perspective*. 2009 (cit. on pp. 22, 23, 27, 35).
- [54] A. V. Messem. *Support vector machines: A robust prediction method with applications in bioinformatics*. 1st ed. Vol. 43. Elsevier B.V., 2020, pp. 391–466. DOI: [10.1016/bs.host.2019.08.003](https://doi.org/10.1016/bs.host.2019.08.003). URL: <http://dx.doi.org/10.1016/bs.host.2019.08.003> (cit. on p. 35).
- [55] A. K. Mitra et al. “A gene expression signature distinguishes innate response and resistance to proteasome inhibitors in multiple myeloma”. In: *Blood Cancer Journal* 7 (6 2017), pp. 1–10. ISSN: 20445385. DOI: [10.1038/BCJ.2017.56](https://doi.org/10.1038/BCJ.2017.56) (cit. on pp. 2, 4, 6, 8, 30).
- [56] R. Mohammed, J. Rawashdeh, and M. Abdullah. “Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results”. In: Institute of Electrical and Electronics Engineers Inc., 2020-04, pp. 243–248. ISBN: 9781728162270. DOI: [10.1109/ICICS49469.2020.239556](https://doi.org/10.1109/ICICS49469.2020.239556) (cit. on p. 12).
- [57] *Mutual information*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.mutual_info_classif.html#r50b872b699c4-1 (cit. on p. 10).

- [58] S. Niño-Ramírez et al. "Reducing the heterogeneity in hepatocellular carcinoma. A cluster analysis based on clinical variables in patients treated at a quaternary care hospital". In: *Revista de Gastroenterología de México (English Edition)* 86 (4 2021), pp. 356–362. ISSN: 2255534X. DOI: [10.1016/j.rgmxe.2021.08.010](https://doi.org/10.1016/j.rgmxe.2021.08.010) (cit. on pp. 10, 39, 40).
- [59] W. S. Noble. "What is a support vector machine?" In: *Nature Biotechnology* 24 (12 2006), pp. 1565–1567. ISSN: 10870156. DOI: [10.1038/nbt1206-1565](https://doi.org/10.1038/nbt1206-1565) (cit. on p. 35).
- [60] A. M. Orgueira et al. "Unsupervised machine learning improves risk stratification in newly diagnosed multiple myeloma: an analysis of the Spanish Myeloma Group". In: *Blood Cancer Journal* 12 (4 2022). ISSN: 20445385. DOI: [10.1038/s41408-022-00647-z](https://doi.org/10.1038/s41408-022-00647-z) (cit. on pp. 3, 7, 8, 14, 27).
- [61] A. M. Orgueira et al. "Prognostic Stratification of Multiple Myeloma Using Clinico-genomic Models: Validation and Performance Analysis of the IAC-50 Model". In: *HemaSphere* 6 (8 2022), e760. DOI: [10.1097/hs9.0000000000000760](https://doi.org/10.1097/hs9.0000000000000760) (cit. on pp. 3, 8, 30).
- [62] A. Palumbo et al. "Revised international staging system for multiple myeloma: A report from international myeloma working group". In: *Journal of Clinical Oncology* 33 (26 2015), pp. 2863–2869. ISSN: 15277755. DOI: [10.1200/JCO.2015.61.2267](https://doi.org/10.1200/JCO.2015.61.2267) (cit. on p. 2).
- [63] A. Paner, P. Patel, and B. Dhakal. "The evolving role of translocation t(11;14) in the biology, prognosis, and management of multiple myeloma". In: *Blood Reviews* 41 (2020-05). ISSN: 15321681. DOI: [10.1016/j.blre.2019.100643](https://doi.org/10.1016/j.blre.2019.100643) (cit. on p. 2).
- [64] H. Pang, M. Hauser, and S. Minvielle. "Pathway-based identification of SNPs predictive of survival". In: *European Journal of Human Genetics* 19 (6 2011), pp. 704–709. ISSN: 10184813. DOI: [10.1038/ejhg.2011.3](https://doi.org/10.1038/ejhg.2011.3) (cit. on pp. 8, 30).
- [65] P. Petsophonsakul et al. "Augmenting THERapeutic Effectiveness Through Novel Analytics (ATHENA)-A Public and Private Partnership Project Funded by the Flemish Government (VLAIO)". In: *Studies in Health Technology and Informatics* 294 (2022), pp. 829–833. ISSN: 18798365. DOI: [10.3233/SHTI220601](https://doi.org/10.3233/SHTI220601) (cit. on pp. 4, 6).
- [66] S. Pourahmad, S. Pourhashemi, and M. Mohammadianpanah. "Colorectal cancer staging using three clustering methods based on preoperative clinical findings". In: *Asian Pacific Journal of Cancer Prevention* 17 (2 2016), pp. 823–827. ISSN: 2476762X. DOI: [10.7314/APJCP.2016.17.2.823](https://doi.org/10.7314/APJCP.2016.17.2.823) (cit. on p. 27).
- [67] S. Pourahmad et al. "Evaluation of Twenty Genes in Prognosis of Patients with Ovarian Cancer Using Four Different Clustering Methods". In: *Asian Pacific Journal of Cancer Prevention* 22 (6 2021), pp. 1781–1787. ISSN: 2476762X. DOI: [10.31557/APJCP.2021.22.6.1781](https://doi.org/10.31557/APJCP.2021.22.6.1781) (cit. on p. 27).

- [68] L. V. Pova, C. H. C. Ribeiro, and I. T. da Silva. "Machine learning predicts treatment sensitivity in multiple myeloma based on molecular and clinical information coupled with drug response". In: *PLoS ONE* 16 (7 July 2021), pp. 1–17. ISSN: 19326203. DOI: [10.1371/journal.pone.0254596](https://doi.org/10.1371/journal.pone.0254596). URL: <http://dx.doi.org/10.1371/journal.pone.0254596> (cit. on pp. 2, 3, 6, 39, 40).
- [69] S. Ray. "A Quick Review of Machine Learning Algorithms". In: *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019* (2019), pp. 35–39. DOI: [10.1109/COMITCon.2019.8862451](https://doi.org/10.1109/COMITCon.2019.8862451) (cit. on p. 35).
- [70] M Ringner. "What is principal components analysis?" In: *Nature Publishing Group - Nature Biotechnology* 26 (3 2008), pp. 303–304 (cit. on p. 20).
- [71] K.-L. Royle and D. A. Cairns. "The development and validation of prognostic models for overall survival in the presence of missing data in the training dataset: a strategy with a detailed example". In: *Diagnostic and Prognostic Research* 5 (1 2021-12). DOI: [10.1186/s41512-021-00103-9](https://doi.org/10.1186/s41512-021-00103-9) (cit. on pp. 6, 11).
- [72] *Shap values*. URL: https://shap.readthedocs.io/en/latest/example_notebooks/overviews/An%20introduction%20to%20explainable%20AI%20with%20Shapley%20values.html (cit. on p. 38).
- [73] L. S. Shapley. "A Value for n-Person Games". In: (1952) (cit. on p. 38).
- [74] J. D. Shaughnessy et al. "Avalidated gene expression model of high-risk multiple myeloma is defined by deregulated expression of genes mapping to chromosome 1". In: *Blood* 109 (6 2007), pp. 2276–2284. ISSN: 00064971. DOI: [10.1182/blood-2006-07-038430](https://doi.org/10.1182/blood-2006-07-038430). URL: <http://dx.doi.org/10.1182/blood-2006-07-038430> (cit. on pp. 1, 3).
- [75] N. Shukla et al. "Breast cancer data analysis for survivability studies and prediction". In: *Computer Methods and Programs in Biomedicine* 155 (2018-03), pp. 199–208. ISSN: 18727565. DOI: [10.1016/j.cmpb.2017.12.011](https://doi.org/10.1016/j.cmpb.2017.12.011) (cit. on pp. 7, 8).
- [76] C. Y. Soekojo et al. "Genomic characterization of functional high-risk multiple myeloma patients". In: *Blood Cancer Journal* 12 (1 2022), pp. 1–9. ISSN: 20445385. DOI: [10.1038/s41408-021-00576-3](https://doi.org/10.1038/s41408-021-00576-3) (cit. on pp. 3, 6–8, 10, 12, 30).
- [77] J. A. Sterne et al. "Multiple imputation for missing data in epidemiological and clinical research: Potential and pitfalls". In: *BMJ (Online)* 339 (7713 2009-07), pp. 157–160. ISSN: 17561833. DOI: [10.1136/bmj.b2393](https://doi.org/10.1136/bmj.b2393) (cit. on p. 11).
- [78] G. Stiglic et al. "Interpretability of machine learning-based prediction models in healthcare". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 10 (5 2020), pp. 1–13. ISSN: 19424795. DOI: [10.1002/widm.1379](https://doi.org/10.1002/widm.1379) (cit. on pp. 39, 40).
- [79] S. Tihon et al. "DAEMA: Denoising Autoencoder with Mask Attention". In: (2021-06). URL: <http://arxiv.org/abs/2106.16057> (cit. on p. 12).

-
- [80] T. Turki and J. T. Wang. "Clinical intelligence: New machine learning techniques for predicting clinical drug response". In: *Computers in Biology and Medicine* 107 (December 2018 2019), pp. 302–322. ISSN: 18790534. DOI: [10.1016/j.combiomed.2018.12.017](https://doi.org/10.1016/j.combiomed.2018.12.017). URL: <https://doi.org/10.1016/j.combiomed.2018.12.017> (cit. on pp. 6, 8).
 - [81] V Vapnik. "Dependencies Estimation". In: *Technology* 909 (1993), pp. 893–909 (cit. on p. 15).
 - [82] J. Viana et al. "Localized Learning : A Possible Alternative to Current Deep Learning Techniques Localized Learning : A Possible Alternative to Current Deep Learning Techniques". In: (2021) (cit. on p. 15).
 - [83] I. R. White, P. Royston, and A. M. Wood. "Multiple imputation using chained equations: Issues and guidance for practice". In: *Statistics in Medicine* 30 (4 2011-02), pp. 377–399. ISSN: 02776715. DOI: [10.1002/sim.4067](https://doi.org/10.1002/sim.4067) (cit. on p. 11).
 - [84] S. T. Wierzchoń and M. A. Kłopotek. *Studies in Big Data; Modern Algorithms of Cluster Analysis*. 2018, p. 433. ISBN: 978-3-319-69308-8. URL: <http://www.springer.com/series/11970> (cit. on pp. 27, 28).
 - [85] M. Xu et al. "Immunogenic Cell Death-Relevant Damage-Associated Molecular Patterns and Sensing Receptors in Triple-Negative Breast Cancer Molecular Subtypes and Implications for Immunotherapy". In: *Frontiers in Oncology* 12 (April 2022), pp. 1–15. ISSN: 2234943X. DOI: [10.3389/fonc.2022.870914](https://doi.org/10.3389/fonc.2022.870914) (cit. on p. 27).
 - [86] M Zaki and W. M. Jr. *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*. Ed. by C. U. Press. 2020 (cit. on pp. 20, 21, 31).
 - [87] M. Zhao et al. "Machine Learning With K-Means Dimensional Reduction for Predicting Survival Outcomes in Patients With Breast Cancer". In: *Cancer Informatics* 17 (2018). ISSN: 11769351. DOI: [10.1177/1176935118810215](https://doi.org/10.1177/1176935118810215) (cit. on pp. 8, 20).
 - [88] X. Zhao et al. "Combining gene signatures improves prediction of breast cancer survival". In: *PLoS ONE* 6 (3 2011). ISSN: 19326203. DOI: [10.1371/journal.pone.0017845](https://doi.org/10.1371/journal.pone.0017845) (cit. on p. 7).

APPENDIX 1

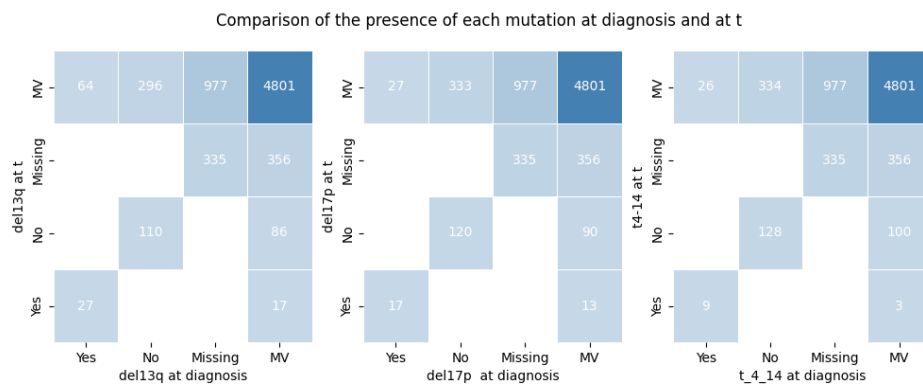


Figure A.1: Comparison of the findings for each mutation at diagnosis and at t (time of treatment line). MV-missing value

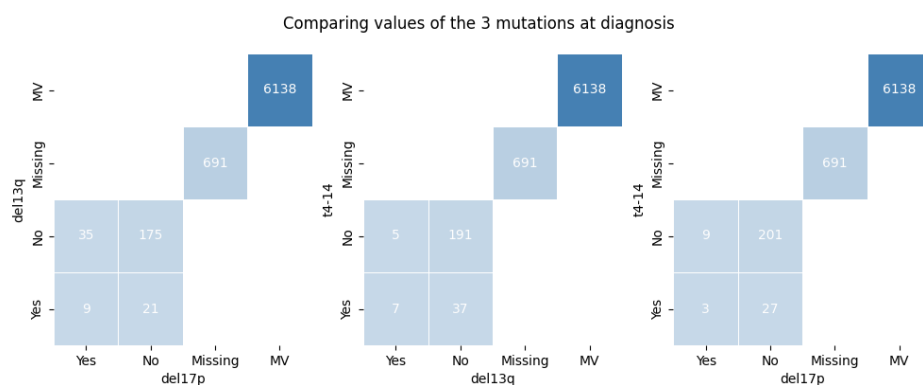


Figure A.2: Comparison of the findings between each mutation at diagnosis. MV-missing value

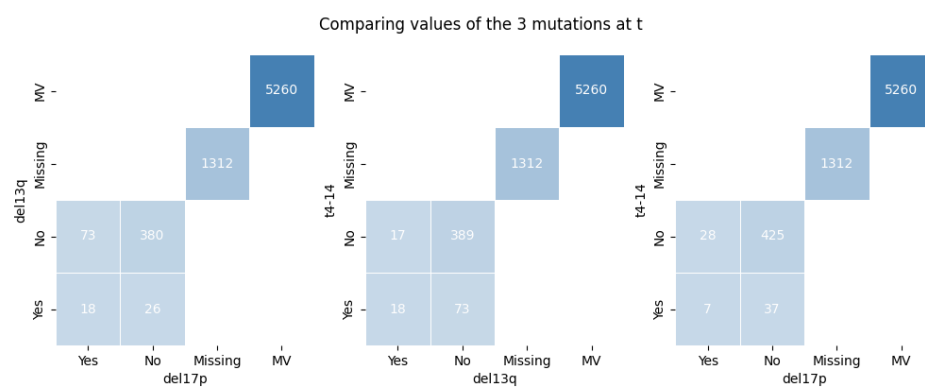


Figure A.3: Comparison of the findings between each mutation at t (time of treatment line). MV-missing value

APPENDIX 2

Table B.1: Treatment Responses [47]

Response	Criteria
CR	Negative immunofixation (serum and urine) + disappearance of plasmacytomas + < 5% plasma cells in bone marrow aspirates
nCR	A subgroup of complete response. < 5% plasma cells in bone marrow aspirates + negative electrophoresis (serum and urine) + positive immunofixation (serum or urine)
VGPR	Serum and urine M-protein detectable by immunofixation but not on electrophoresis OR ≥ 90 reduction in serum M-protein + urine M-protein level <100 mg per 24 h
PR	≥ 50 reduction of serum M-protein + reduction in 24 h urinary M-protein by ≥ 90 or to <200 mg per 24 h; If M-protein is unmeasurable - ≥ 50 decrease in the difference between involved and uninvolved FLC levels is required in place of the M-protein criteria; If also serum-free light assay is unmeasurable, ≥ 50 reduction in plasma cells, provided baseline bone marrow plasma-cell percentage was ≥ 30 . ≥ 50 reduction in the plasmacytomas, if present
MR	≥ 25 but ≤ 90 reduction of serum M-protein + reduction in 24-h urine M-protein by 50-89%. ≥ 50 reduction in the plasmacytomas, if present
SD	Not meeting criteria for CR, nCRm VGPR, PR, MR, PD.
PD	Increase of 25% in: Serum M-protein (absolute increase must be ≥ 0.5 g/dL) OR Serum M-protein increase ≥ 1 g/dL, if the lowest M component was ≥ 5 g/dL OR Urine M-protein (absolute increase must be ≥ 200 mg/24 h) OR If M-protein is unmeasurable, the difference between involved and uninvolved FLC levels (absolute increase must be > 10 mg/dL); (+)If involved FLC levels unmeasurable, bone marrow plasma-cell percentage irrespective of baseline status (absolute increase must be ≥ 10) OR New lesion(s), ≥ 50 increase from nadir in SPD of > 1 lesion OR ≥ 50 increase in the longest diameter of a previous lesion > 1 cm in short axis OR ≥ 50 increase in circulating plasma cells (> 200 cells/ μ L) if this is the only measure of disease

APPENDIX 3

This appendix shows a brief description of the dimensionality reduction methods and results obtained when using the imputed dataset. These methods included principal component analysis, t-SNE and Isomap.

Firstly, principal component analysis was implemented. The feature contribution for each principal component is shown in Figure C.

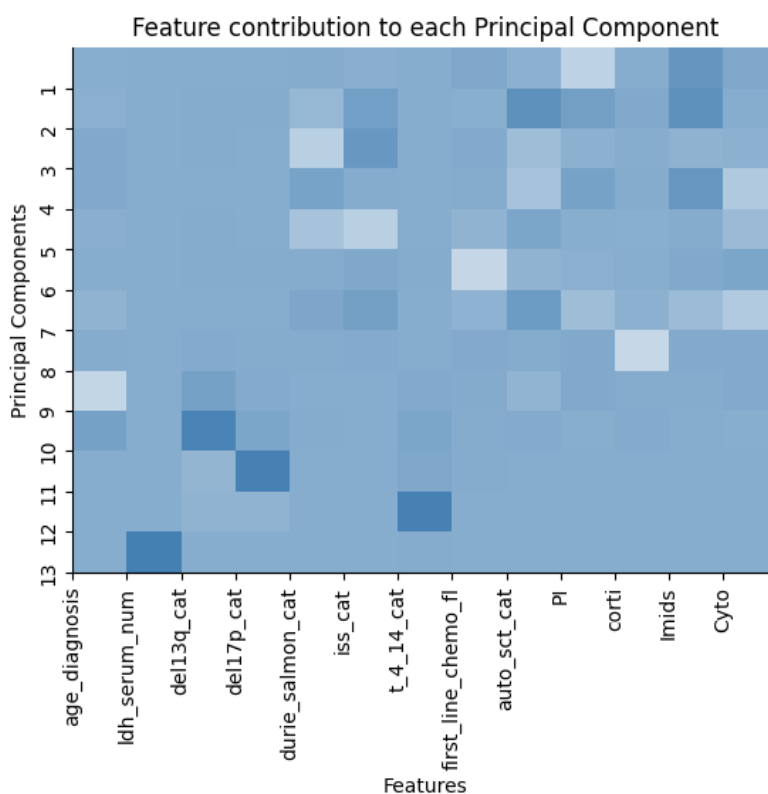


Figure C.1: Feature contribution to each principal component.

To choose the number of principal components to retain, several methods were applied. The Kaiser method showed the first 6 principal components had a higher variance than the mean of the original features. On the other hand, the scree plot method showed an elbow at 5 and 9 components, as is shown in Figure C.2.

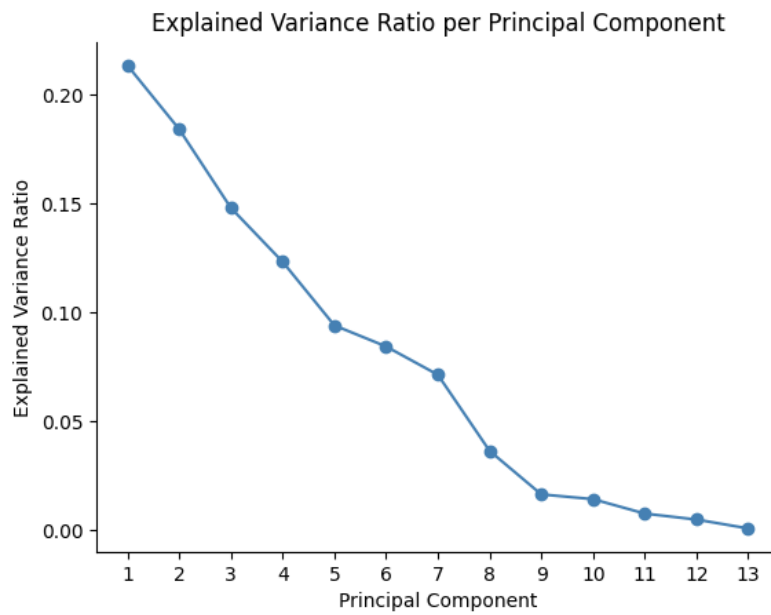


Figure C.2: Principal Component Analysis - Selecting the number of principal components using a scree plot.

The cumulative variance plot and a plot showing the reconstruction error per number of principal components showed no elbows. The reconstruction error plot is shown in Figure C.3.

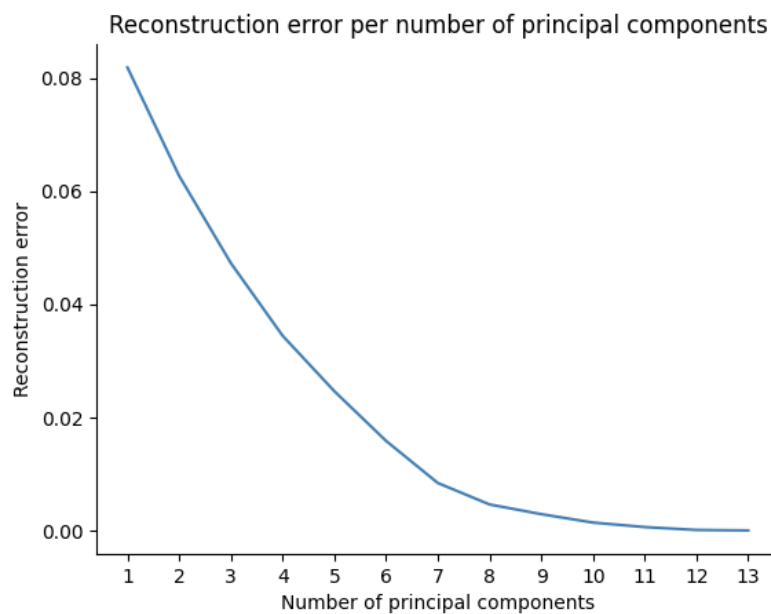


Figure C.3: Principal Component Analysis - Selecting the number of principal components the reconstruction error per number of principal components retained.

It was decided to use the results of the Kaiser method to determine the number of principal components to retain. The first 6 principal components were retained. For visualization purposes, Figure C.4 shows the data obtained by using the principal component

analysis projected on the first 3 principal components.

Projection of the data in the first three principal components

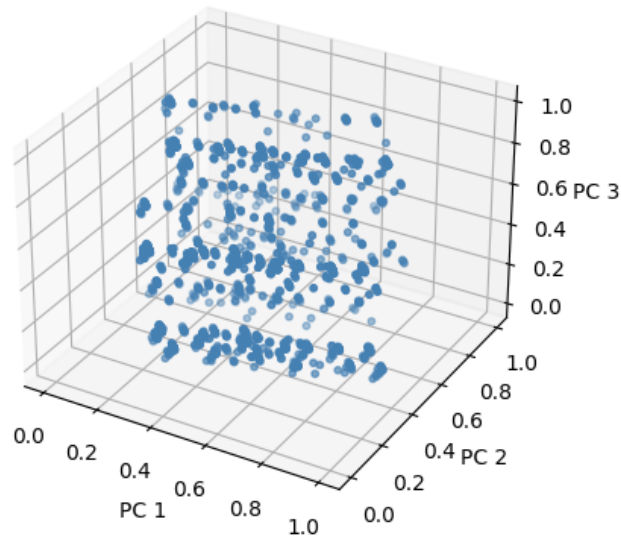


Figure C.4: Principal Component Analysis results projected on the first 3 principal components

Several clustering algorithms were tested and the results showed no useful results. These are not thoroughly explained in this appendix as they are discussed in the remaining part of chapter 4, applied to data that suffered a different preprocessing and dimensionality reduction method.

For visualization, the representations obtained using t-SNE and Isomap are shown in Figures [C.5](#) and [C.6](#)

Again, due to limited results, these analysis will not be more thoroughly explained in this report.

Projection of the data using t_SNE

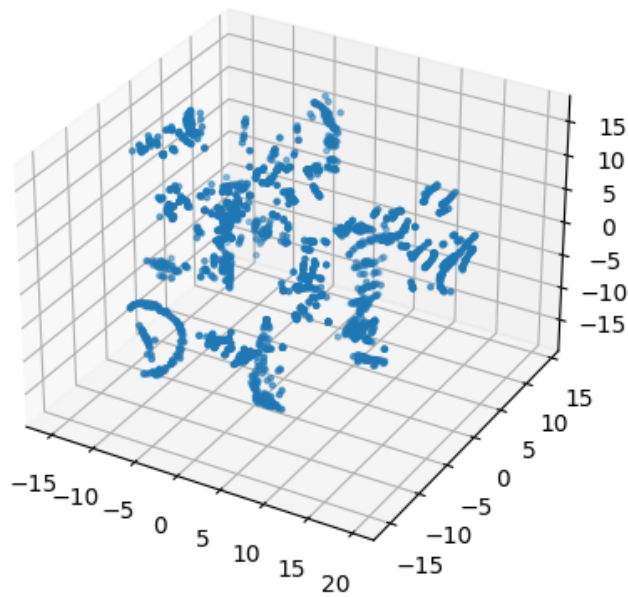


Figure C.5: Representation of the data obtained using t-SNE

Projection of the data with isomap

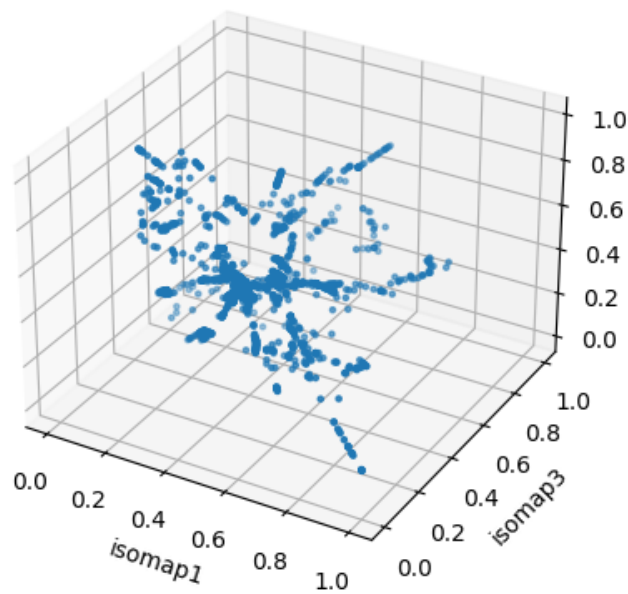


Figure C.6: Representation of the data obtained using Isomap

D

APPENDIX 4

	6 MV (n = 2397)	5 MV (n=2206)	4 MV (n=1015)	3 MV (n=402)	2 MV (n=290)	1 MV (n=190)	0 MV (n=62)
durie_salmon_cat	29.37% (704)	23.25% (513)	21.87% (222)	29.85% (120)	34.83% (101)	30.00% (57)	0%
iss_cat	59.70% (1431)	56.21% (1240)	30.64% (311)	31.34% (126)	36.90% (107)	21.58% (41)	05
ldh_serum_num	76.39% (1831)	74.34% (1640)	47.09% (47.09)	37.81% (152)	45.86% (133)	22.63 (43)	0%
diag_del_13q_cat	87.11% (2088)	85.99% (1897)	69.56% (706)	23.13% (93)	0%	0%	0%
diag_del_17p_cat	87.11% (2088)	85.99% (1897)	69.56% (706)	23.13% (93)	0%	0%	0%
diag_t_4_14_cat	87.11% (2088)	85.99% (1897)	69.56% (706)	23.13% (93)	0%	0%	0%

age_diagnosis	ldh_serum_num	del13q_cat	del17p_cat	durie_salmon_cat	iss_cat	t_4_14_cat	first_line_chemo_fl	auto_sct_cat	status_os	time_to_os	PI	corti	Imids	Cyto	0
1	0	0	0	1	0	0	1	1	1	1	1	1	1	1	667
1	0	0	0	1	1	0	1	1	1	1	1	1	1	1	256
1	1	0	0	1	0	0	1	1	1	1	1	1	1	1	218
1	0	0	0	0	1	0	1	1	1	1	1	1	1	1	198
1	1	0	0	1	1	0	1	1	1	1	1	1	1	1	128
1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	124
1	1	0	0	0	1	0	1	1	1	1	1	1	1	1	111
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	63
1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	59
1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	56
1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	27
1	0	1	1	1	1	0	1	1	1	1	1	1	1	1	26
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	22
1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	17
1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	11
1	0	1	1	1	0	0	1	1	1	1	1	1	1	1	4

APPENDIX 5

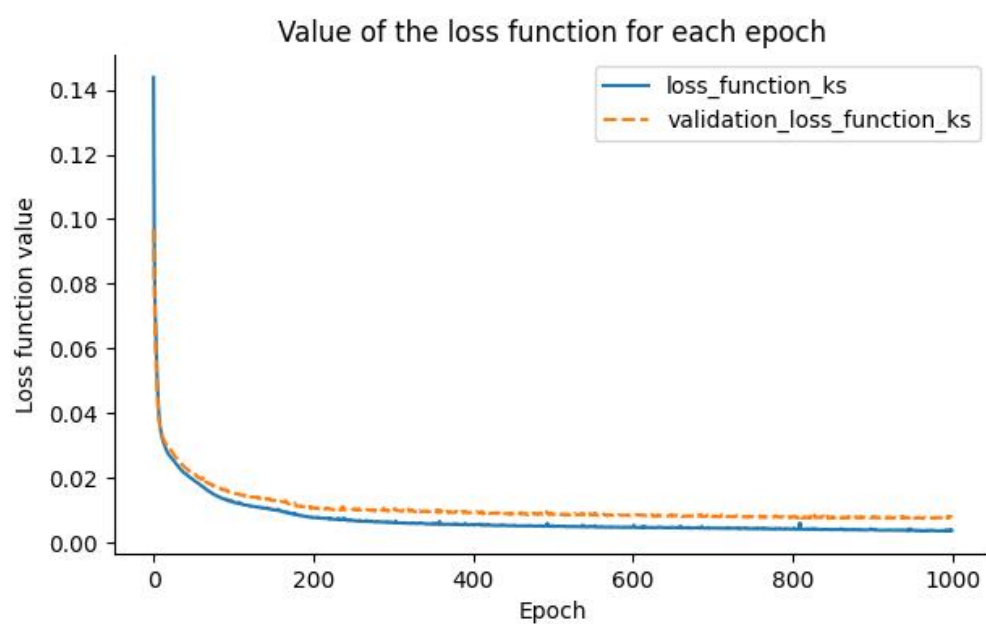


Figure E.1: Evolution of the value of the loss function for each training epoch.

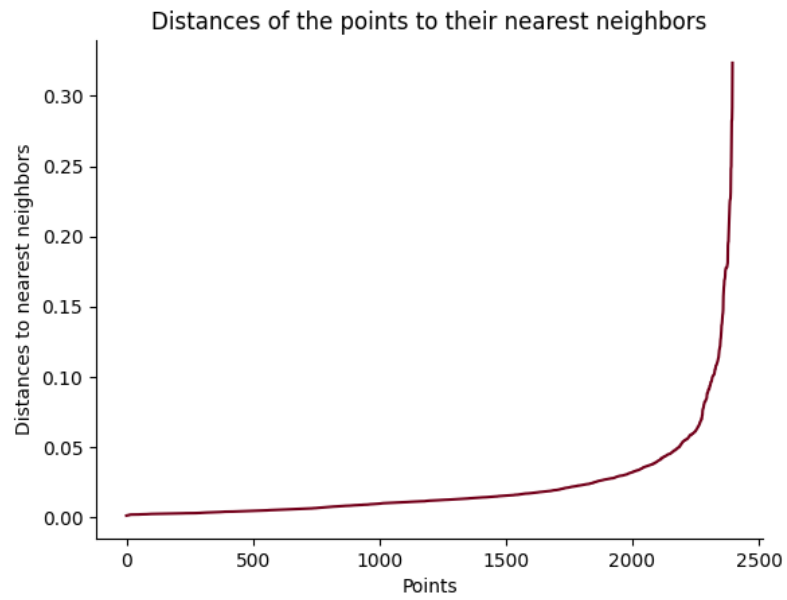


Figure E.2: Nearest neighbors distances to define best epsilon value for DBSCAN

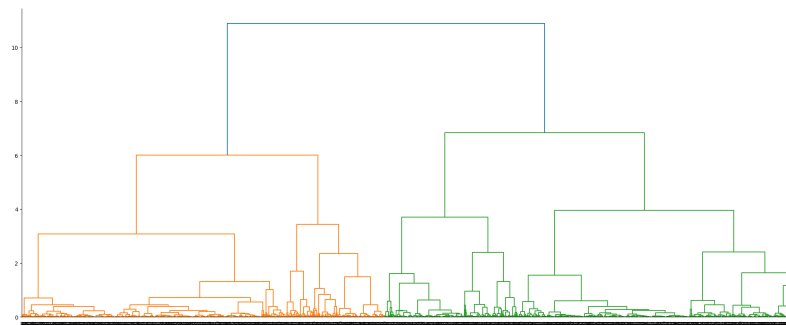


Figure E.3: Dendrogram for Hierarchical Clustering of the data resulting from the autoencoder transformation.

APPENDIX 6

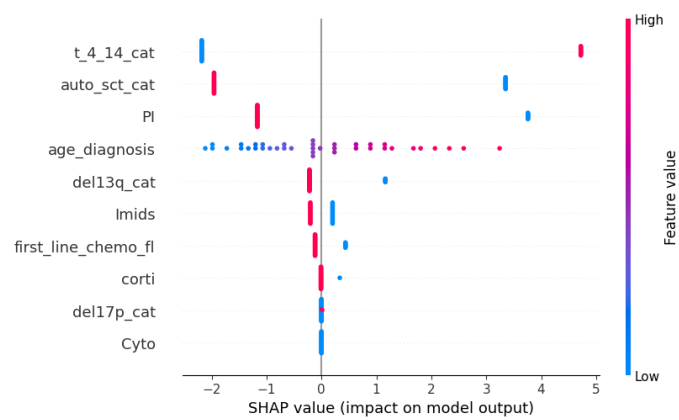


Figure F.1: Shap Values for subphenotype 1

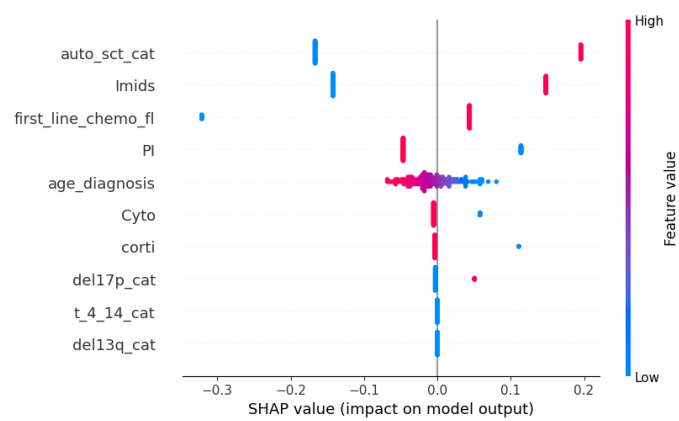


Figure F.2: Shap Values for subphenotype 2



Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology

Mathematical Biology