Data Mining Project Report

Ana Mangarovska, Anastazija Kovachevikj, Guilherme Soares, Tiago Oliveira

May 10, 2023

1 Introduction and Objectives

Diffuse large-B-cell lymphoma is the most prevalent type of lymphoma in adults; nevertheless, the administration of anthracycline-based chemotherapy on patients that exhibit this disease yields a very low cure rate. The study of microarray features has led to the identification of two subgroups of lymphomas, each one associated with different outcomes after chemotherapy. Moreover, these gene-expression profiles have also been used in survival analysis not only to further refine the previous distinction of lymphomas, but also to formulate a molecular predictor of the success of chemotherapy. Hence, gene-expression micro-array data has shown to be a promising tool in prognostics that deserves full attention.

In this project, we intend to analyse and apply some analysis techniques to a real dataset resulting from a study on this subject. Our objectives include a preliminary analysis of the data, data preprocessing, a check for missing values and impute them if exist, Outlier Detection, Principal Component Analysis and testing different clustering methods.

2 Missing Values Analysis

In the beginning of our study, we analysed the order of magnitude of the missing values present in our dataset. As we can see in Fig. 1, there are genes (variables) that have a significant number of missing values (some more than 100, that is, data related to those genes is missing in 100 or more samples, which is relevant, considering that our dataset only has 240 samples). To reduce the impact of the missing values in our posterior analysis, and because we have more than 7000 variables, we decided to remove the variables that presented more than 5% of missing values, so we kept a total of 4035 genes. To deal with the missing values of those 4035 genes, we proceeded with some value imputation. To do that, we used the K Nearest Neighbors method [5]. In our case, after testing several values of K (1, 2, 3, 5, 8, 10) and choosing the Euclidean distance to give greater influence to closer neighbors and less influence to those which are further away in the prediction of the values, we decided to use K = 5. Although the KNN method can be sensitive to outliers, it can be more accurate than the average, median or most frequent imputation methods.

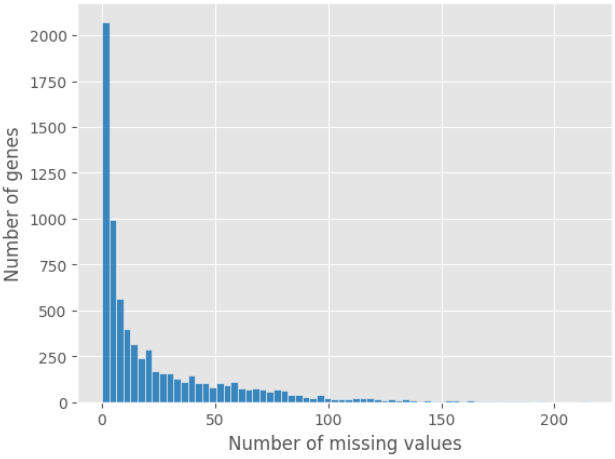

Figure 1: Number of missing values vs Number of genes

3 PCA

3.1 PCA on Signatures

Given the five signatures and its expression in all individuals we decided to apply Principal Component Analysis (PCA) [2], in order to understand which signatures are more relevant to the outcome (dead or alive) for the individuals. We plotted the contribution of each signature for the first two PCs in Fig.2. Then, we added the individuals projected in the same space - see Fig.3 - and coloured each dot according to each individual's outcome.

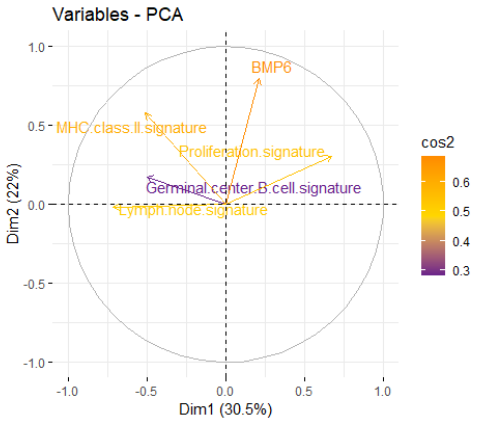

Figure 2: Contribution of each signature to the first two PCs. The five signatures were projected in two dimensions. The square of the cosine is captured by the coloured label.

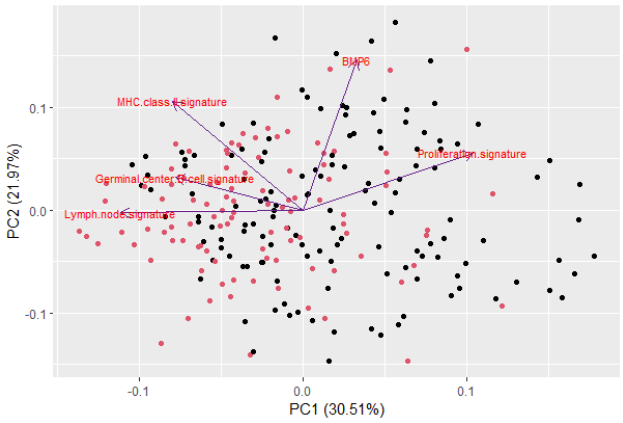

Figure 3: Outcome of all individuals projected in the plane obtained using PC1 and PC2. The colour of each dot indicates if the outcome is alive (red) or dead (black).

By taking a closer look to these two plots, we see that the majority of dead people (black dots) are on the right side of the graph of Fig.3, while the red dots (favorable outcome) are mostly on the left side. This agrees with the conclusions presented on the paper [4], as it states that the signatures *MHC class*, *Germinal center B-cell* and *Lymph node* yield a good outcome. Notice that the projections of these signatures point to the left as well. On the other hand, signatures correlated with a poor outcome in the paper - *Proliferation* and others - are pointing to the right side, just like the black dots. Nevertheless, it is also worth noticing that not all the signatures are well represented by the PCA projections. Particularly, the *Germinal B-cell* signature has a very poor representation, since its cosine squared is quite close to zero.

3.2 PCA on Genes

3.2.1 Classical PCA

Regarding the genes provided by the `microarray.csv` file, we decided to apply PCA in order to reduce dimensionality. Initially, our dataset consists of about 4000 genes, which explains the need to reduce the dimensionality, since it is not practical to handle such a big number of variables. Before proceeding with the PCA, we normalized the data. This step is crutial since variables in a larger scale would explain more variability - which would not be correct. We opted to use the min-max normalization [3], which scales and translates each feature individually such that it is in the given range on the training set, e.g. between zero and one. After applying PCA, we chose to retain the number of PCs needed to explain 80% of all the variance. This number is 62, as we can see in Fig.5.

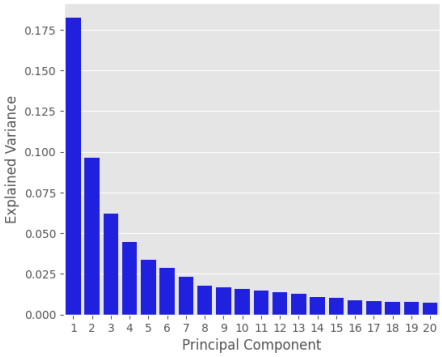

Figure 4: Variance explained by each Principal Component

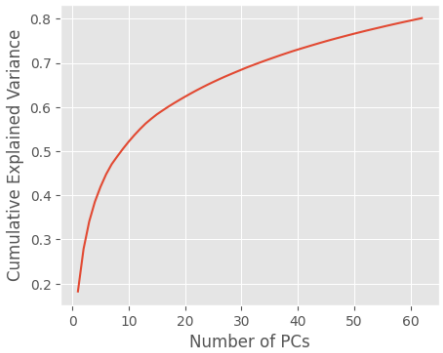

Figure 5: Cumulative variance in function of the number of Principal Components

3.2.2 Robust PCA

First of all, we shall observe that PCA is highly sensitive to outliers. This is due to the fact that PCA will rotate the coordinate axis in order to capture the directions of higher variance. The presence of outliers in our dataset will precisely increase the variance in its direction since they will be far from the majority of the other points, by the definition of *outlier*. Thus, they will influence the directions of PCA in an unwanted but drastic way. Taking this into account it is clear that we need a PCA technique robust to outliers - and that is why we implemented Robust PCA [1]. In a few words, robust PCA will weight less points that are far away from the majority of points. This means they will have much less influence on the direction of the new axis. Similarly to the classic PCA, we kept the first k components that can explain 80% of variability. This time k equals 57.

4 Outlier Detection

Outliers are extreme values that stand out from the overall pattern of values in a dataset. It is important to note that not all outliers are errors or indicative of unusual phenomena. It may simply be an extreme value that is within the range of what is expected given the distribution of the data. So, it is necessary to investigate the cause in order to understand its implications.

4.1 Using Mahalanobis distance

In Mahalanobis distance (MD), where $\boldsymbol{\mu}$ is the mean vector of the distribution, $\boldsymbol{\Sigma}$ the covariance matrix and τ is the threshold value, the following rule is used to define if an observation x is an outlier:

$$\text{If } (\mathbf{x}-\boldsymbol{\mu})^{\intercal}\,\boldsymbol{\Sigma}^{-1}\,(\mathbf{x}-\boldsymbol{\mu}) \geq \tau \Rightarrow \mathbf{x} \text{ is an Outlier}$$
$$\text{Otherwise} \Rightarrow \mathbf{x} \text{ is Regular}$$

In classical MD, it is assumed that the distribution is normal and that the sample data used to estimate the parameters of the distribution (mean and covariance) are free of outliers. Thus their presence can significantly alter the estimated mean and covariance. On the other hand, the robust MD uses the Minimum Covariance Determinant (MCD) to estimate the mean and the covariance of a distribution, which is less sensitive to the presence of outliers in the data. The MCD estimate is obtained by finding the subset of the data that has the smallest possible determinant of the covariance matrix, while still containing a certain percentage of the data. This subset is then used to estimate the mean and covariance of the distribution. The classical MD detected 42 outliers and the robust MD detected 84.

4.2 Using PCA

Outliers can be detected using PCA by estimating the first PCs and projecting the data into the main directions. Then, compute the orthogonal and score distances of each projected point. Finally, we use a threshold to determine which points are outliers, as it can be seen in figure 6 and figure 7. For this, we tested grid and robust PCA. Grid PCA is designed to handle large-scale data sets that do not fit in memory. It works by dividing the data into a grid of smaller subsets, and performing PCA on each subset separately. The resulting principal components from each subset are then combined to form the final set of principal components for the entire data set. The Grid PCA detected 15 outliers and the robust PCA detected 86.

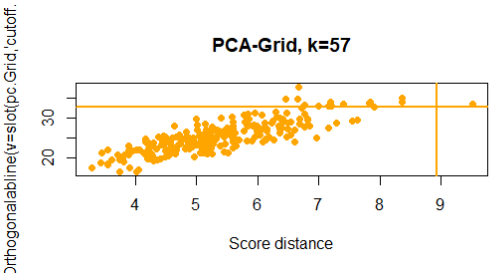

Figure 6: Score and Orthogonal distance plot for Grid PCA with 57 PC's

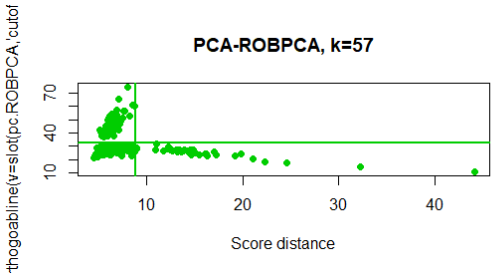

Figure 7: Score and Orthogonal distance plot for robust PCA with 57 PC's

5 Clustering

Our next step was to compute clusters in an attempt to recover some results of the paper. More specifically, we computed clusters using patients as objects in order to obtain the subgroups mentioned in the paper: GCB, Type III and ABC.

5.1 Hierarchical methods

Here we present the results of clustering provided by the WARD and Complete linkage methods. These were the two hierarchical methods that gave us more plausible results, when compared to the others mentioned. We say that these ones were plausible because, by inspecting the dendograms in Figs. 8 and 9, it is clear that we can divide the population in 3 different groups easily - which did not happen with the Average and Single linkage.

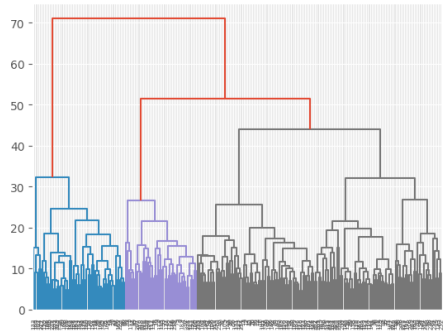

Figure 8: Dendogram with WARD's method

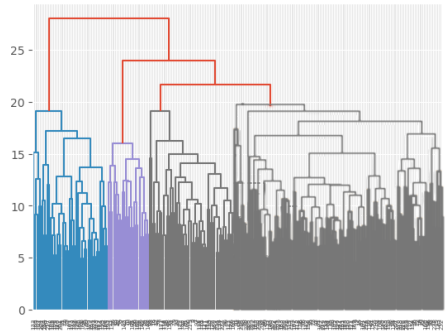

Figure 9: Dendogram with Complete Linkage

Now, our aim is to compare our results with the ones in the paper. Fig. 10 has the patients projected in the first 2 principal components obtained before, and each group is represented by a different colour and a numbered label. Namely, **0** is the ABC group, **1** is the GBC and **2** is the Type III. It is worth to mention that Fig.10 is simply a 2D projection of our data with the unique purpose of data visualisation. It does not mean that we are finding clusters in a 2D space. Instead, clusters were computed with 57 variables as mentioned above. In an attempt to validate our results, we did the same thing to our clusters.

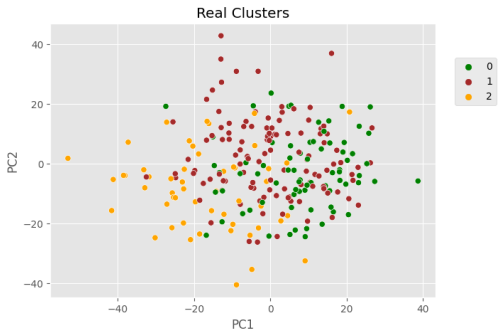

Figure 10: Clusters according to the paper

Figure 11: Clusters using WARD's method

Figure 12: Clusters using Complete linkage

The results were plotted in Figs. 11 and 12 for WARD and Complete linkage, respectively. In this case, we cannot establish a straight forward connection between each colour/label in the graph and each subgroup,

because unfortunately there is not a clear similarity between any of these graphs and the one on Fig. 10. For both methods we computed the Silhouette graphs in order to validate its quality. The Silhouette is a graphical display for partitioning techniques in which each cluster is represented by a silhouette, based on its tightness and separation. The Silhouette Coefficient is a metric used to calculate how good a clustering technique is. Its value ranges from -1 to 1. Taking into account that we know *a priori* that there are 3 subgroups (GBC, ABC and Type III) we concluded that the number of clusters must be 3 as well. According to the graph plotted on Fig.13, we observe that the silhouete score decreases from 2 to 3 clusters. However, we must keep 3 as the number of clusters because of the reasons presented above. By inspection of the average Silhouette score in both Figs. 14 and 15 it is clear that WARD method performs better. However, none of these seem to have a good Silhouette score. In fact, there might be a good reason for the fact that our results are not matching with the paper ones. In the paper, they state that the 3 subgroups were obtained based on 100 genes, while in our case clusters were computed after PCA which left us with 57 features. Also, these 57 dimensions are not genes, but linear combinations of genes.

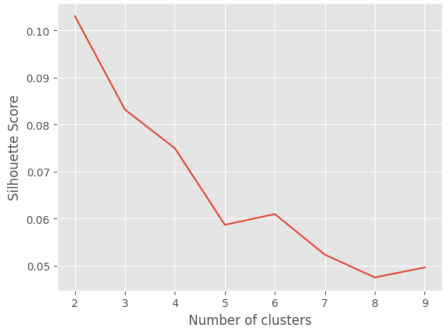

Figure 13: Silhouette score as a function of the number of clusters using WARD

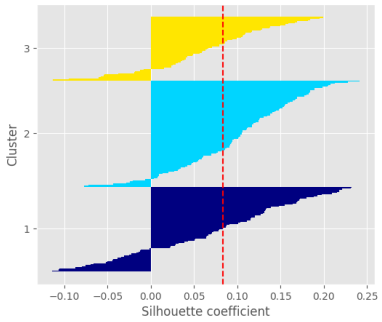

Figure 14: Silhouette Analysis with WARD's method

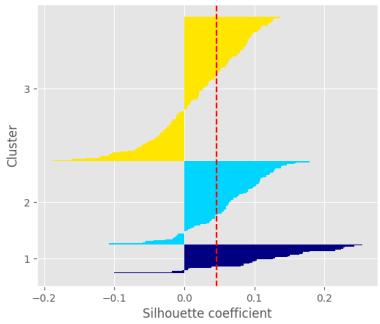

Figure 15: Silhouette Analysis with Complete Linkage method

5.2 Partitioning method

K-means is a clustering algorithm based on the computation of the centroids of each partition (cluster) at each iteration. Partitions are computed based on the minimum distance to the centroids. On Fig.16 was plotted a graph analogous to the ones above (see Fig.10, for instance), for 3 clusters. By comparing the average Silhouette score in Figure 17 with the ones in Figures 14 and 15, we can observe that K-means performs slightly better for this data. Even so, its performance is not very good either, probably for the reasons already given for the other methods.

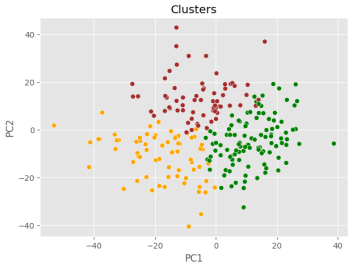

Figure 16: Clusters using K-means

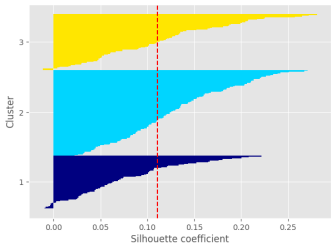

Figure 17: Silhouette Analysis with K-means

5.3 Discussion

Clustering Method	Accuracy (%)		
	Classical PCA	Robust PCA	Without PCA
Ward	55.41	59.58	50.00
Complete Linkage	62.91	47.08	49.16
KMeans	57.91	59.17	59.58

Table 1: Clustering method's accuracy

Here, we compare the accuracy of all methods we discussed. Accuracy was calculated based on the percentage of patients that were classified in the correct subgroup. By taking a first look, we see that having implemented PCA often improves the model's accuracy due to the fact that it yields a model with much less variables, when compared to the model without PCA. Then, when comparing Classical vs Robust PCA, it is clear that Robust PCA yields better results, with the only exception of complete linkage method. Finally, if we consider all methods implemented, we get our best accuracy with Complete linkage using classical PCA.

6 Conclusion

Throughout this study, various techniques of multivariate analysis learned in class were applied. Regarding the Missing Values Analysis and Imputation, we noticed that there was a considerable number of variables that had a percentage of missing values above the desirable and that, when discarded from our dataset, ended up affecting the entire study subsequently carried out. The imputation techniques used in variables that remained despite having missing values also affected the results. It would be interesting to test other methods of Data Imputation, use the entire imputed dataset or change the percentage of missing values allowed per variable. Regarding the PCA, we realized that MHC class, Germinal center B-cell and Lymph node are good predictors of a favorable outcome as referred in the paper. It also allowed us to reduce the dimensionality of the data and thus to apply other techniques more efficiently and without a computational burden that would make our study difficult. As the classic PCA is highly sensitive to outliers, we applied the Robust PCA to our dataset. Regarding Outlier Detection, we used four different strategies: Classical and Robust Mahalanobis distance and Grid and Robust PCA. From the results we obtained, we could see that the robust methods of both techniques detected significantly more outliers than the other methods, although we chose not to remove any data from the dataset. In the future, it would be interesting to remove outliers in order to understand their effect on the results, use other robust methods for their detection and evaluate one by one the samples that were classified as outliers in order to understand if there was any common point in these data. In terms of Clustering, several methods were tested, both hierarchical and partitioning methods. A Silhouette analysis was also performed for all of them. When analyzing the results we noticed that the accuracies were below the desired for all methods when combined with the Classic PCA, Robust PCA and in the dataset without PCA, not existing one that stood out from the rest. In the future, it would be interesting to test other clustering methods as well as to find out which genes, out of all 7000, are specifically used in patient classification.

References

[1] Mia Hubert, Peter J Rousseeuw, and Karlien Vanden Branden. Robpca: A new approach to robust principal component analysis. *Technometrics*, 47(1):64–79, 2005.

[2] Richard Arnold Johnson and Dean W. Wichern. *Applied multivariate statistical analysis*. Prentice Hall, Upper Saddle River, NJ, 5. ed edition, 2002.

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[4] Andreas Rosenwald, George Wright, Wing C Chan, Joseph M Connors, Elias Campo, Richard I Fisher, Randy D Gascoyne, H Konrad Muller-Hermelink, Erlend B Smeland, Jena M Giltnane, et al. The use of molecular profiling to predict survival after chemotherapy for diffuse large-b-cell lymphoma. *New England Journal of Medicine*, 346(25):1937–1947, 2002.

[5] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.