Introduction to Machine Learning and Data Mining

Biomarkers of Breast Cancer

Jacob Sonne

October 24, 2018

This report is submitted as partial fulfillment of the requirements for the course Introduction to Machine Learning and Data Mining (02450) at the Technical University of Denmark.

Contents

Summary

In this report the feasibility of screening for breast cancer using a blood sample where nine attributes are measured is asserted. The data sample contains 64 patients with breast cancer and 52 healthy controls.

The best classification model is found to be a decision tree with an the estimated generalization error of 0.22, which can be used to estimate the model performance for unseen data. The best regression model is found to be logistic regression with an estimated generalization error of 0.28. The relatively small sample size gives substantial variations in the estimated generalization errors between successive runs.

Data clustering analyses were, at large, futile when comparing the predicted clusters to the class labels in the data sample.

From density scoring methods potential outliers were identified, but due to lack of knowledge about the uncertainty in the blood analysis method and the variations in human physiology the classification, regression and clustering analyses were not repeated with a pruned data set.

Association mining revealed that high "Glucose" levels in combination with high "HOMA" levels (or high "Insulin" levels) are found for the patients with a confidence of just below 80 %.

In most countries the breast cancer incidence rate is about 100 out of 100,000 women. In such populations our models will have a false-positive rate of about 24,000 while the false-negative rate will be about 30. These numbers should be carefully considered before implementing a screening program using blood samples.

1 Introduction

In this report the feasibility of screening for breast cancer using a cheap blood sample where nine attributes are measured will be asserted. Below follows an excerpt from the description provided at the data source [7].

> ***Abstract:*** *Clinical features were observed or measured for 64 patients with breast cancer and 52 healthy controls [...]*
>
> ***Data Set Information:*** *There are 10 predictors, all quantitative, and a binary dependent variable, indicating the presence or absence of breast cancer. The predictors are anthropometric data and parameters which can be gathered in routine blood analysis. Prediction models based on these predictors, if accurate, can potentially be used as a biomarker of breast cancer.*
>
> ***Attribute Information:*** *Age (years), BMI (kg/m^2), Glucose (mg/dL), Insulin (μU/mL), HOMA, Leptin (ng/mL), Adiponectin (μg/mL), Resistin (ng/mL), MCP-1(pg/dL).*
>
> *Each individual in the dataset belong either to the class "Healthy controls" or the class "Patients".*

Patrício et al. [6] analyzed the data using logistic regression, random forests and support vector machines. Due to time constraints their analysis will not be discussed further in this report.

This report is structured as follows. First I describe the prerequisites for running the code before describing the data sample in terms of summary statistics, principal components and correlations. Next I describe my efforts into supervised learning where I fit both classification and regression models that are then able to classify data objects as belonging either to the "Healthy control" or to the "Patient" class. Then I model the data using linear regression where one continuous attribute is predicted from the other continuous attributes. In my example, which is of little practical use, I will predict the value of the attribute "HOMA". After the regression section I describe two unsupervised methods for clustering the data namely hierarchical clustering and Gaussian mixture models and compare their clusters with the true classes. Arguably somewhat late in the report I then turn to outlier detection by means of density scoring methods before discussion the problem at hand using association mining. The discussion then puts the results in a real life context i.e. try to asses the performance of the models in a more realistic population, which is very imbalanced due to a (thankfully) low incidence rate. The discussion is followed by the summary and appendices.

2 Prerequisites

For this report I have used Anaconda Python 2.7.15. The required packages are defined in the Anaconda environment file and the source code used for this report is available online [1]. The package `mltools` is based on the folder `02450Toolbox_PythonTools/Tools` in the course material, which I made into a Python wheel using `setup.py` [1]. For association mining I used the Apriori algorithm from http://www.borgelt.net/apriori.html.

The data used for the report can be downloaded from the UCI machine leaning repository [7].

I have used `scikit-learn`, `scipy` and some functions provided in the course toolbox. The primary source of the theoretical background is the course textbook [3].

3 Feature extraction and visualization

3.1 Data description

The data objects are characterized using nine features: Age (years), BMI (kg/m^2), Glucose (mg/dL), Insulin (μU/mL), HOMA, Leptin (ng/mL), Adiponectin (μg/mL), Resistin (ng/mL), MCP-1(pg/dL) and are each labeled as either "Healthy control" or "Patient". In some of the figures the attribute names have been abbreviated to the first seven letters to increase clarity.

The attributes have the following properties.

- **Discrete**. Class label (Healthy control or Patient).
- **Continuous**. BMI, Glucose, Insulin, HOMA, Leptin, Adiponectin, Resistin, MCP-1. Although Age is reported as integers (discrete) it will be treated as being continuous in the regression analyses.

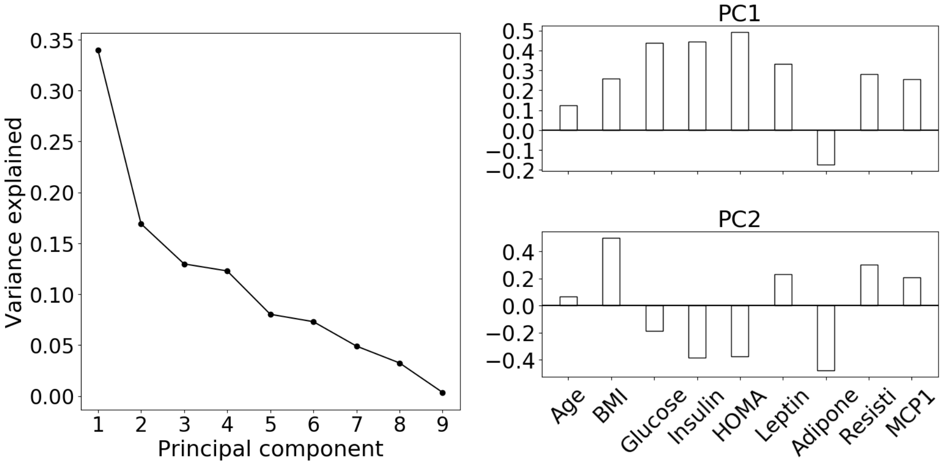
The attributes belong to the following types.

- **Nominal**. Class label (Healthy control or Patient).
- **Ordinal**. None.
- **Interval**. None.
- **Ratio**. BMI, Glucose, Insulin, HOMA, Leptin, Adiponectin, Resistin, MCP-1. Again we will treat Age as being of type "Ratio" since it is used directly as input in e.g. the linear regression.

There appears to be no missing values or corrupted data in the data sample.

3.2 Principal component analysis

The data was standardized to zero mean and unit variance before principal component analysis was carried out. This prevents the units of the measured attributes to influence the analysis. Figure 1a shows the variance explained by each principal component and shows that about 50 percent of the variance is explained by the first two principal components. The coordinates of the two first principal components, in the standardized attribute space, are illustrated in Fig. 1b.



(a)

(b)

Figure 1: Principal component analysis. (a) The fraction of the total variance of the standardized data explained by each of the principal components. The line is inserted to guide the eye. (b) Coordinates of the two first principal components in the standardized attribute space.

Figure 2 shows the standardize data projected on the two first principal components i.e. the two components that have the largest contribution to the variation in the standardized data. In the subspace spanned by the two first principal components there is a significant part of the two classes that overlap, but the patient group may have a tendency to have higher values in the PC1 direction and lower (negative) values in the PC2 direction.

3.3 Descriptive statistics and correlation

Table 1 summarizes the data studied in this report. Similarly, Fig. 3 shows boxplots of the corresponding standardized data, but no obvious differences distinguish the "Healthy control" and the "Patient" classes in the boxplot. The standardization in Fig. 3 increases clarity by putting all attributes to the same scale. There are numerous positive "fliers" for all attributes except

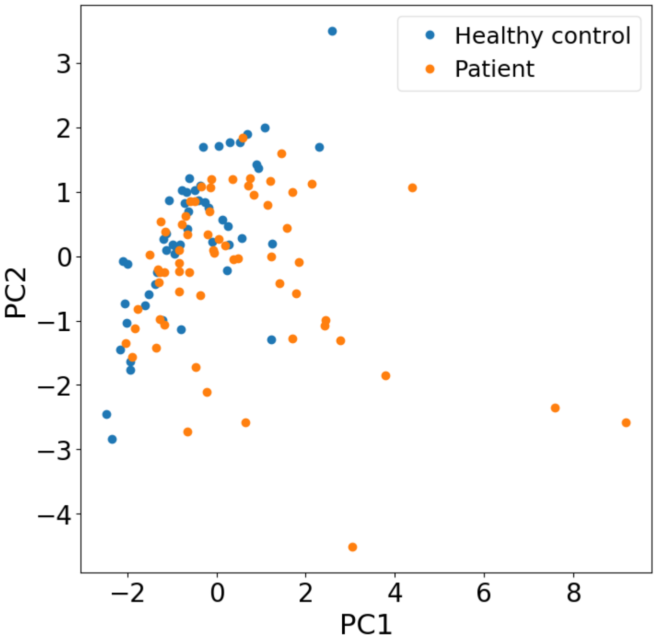


Figure 2: Standardized data projected on the principal components. PC1 and PC2 are the principal components corresponding to largest and second largest eigenvalue, respectively.

Table 1: Summary statistics for each attribute.

	Age	BMI	Glucose	Insulin	HOMA	Leptin	Adiponectin	Resistin	MCP1
count	116.00	116.00	116.00	116.00	116.00	116.00	116.00	116.00	116.00
mean	57.30	27.58	97.79	10.01	2.69	26.62	10.18	14.73	534.65
std	16.11	5.02	22.53	10.07	3.64	19.18	6.84	12.39	345.91
min	24.00	18.37	60.00	2.43	0.47	4.31	1.66	3.21	45.84
25%	45.00	22.97	85.75	4.36	0.92	12.31	5.47	6.88	269.98
50%	56.00	27.66	92.00	5.92	1.38	20.27	8.35	10.83	471.32
75%	71.00	31.24	102.00	11.19	2.86	37.38	11.82	17.76	700.08
max	89.00	38.58	201.00	58.46	25.05	90.28	38.04	82.10	1698.44

"Age" and "BMI" i.e. values falling outside the whiskers in the boxplot. This can be an indication that there are outliers in the data. The histograms in Fig. 4 reveal that many of the distributions are quite asymmetric and have thick positive tails. The attributes "Glucose", "Insulin", "HOMA" and "Resistin" all appear to have positive outliers. In the remainder of this report, these putative outliers are included in the analyses since I do not have sufficient domain knowledge (blood analysis methods and human physiology) to conclusively label them as outliers. In Sec. 6.3 outliers will addressed in greater detail from a statistical perspective.

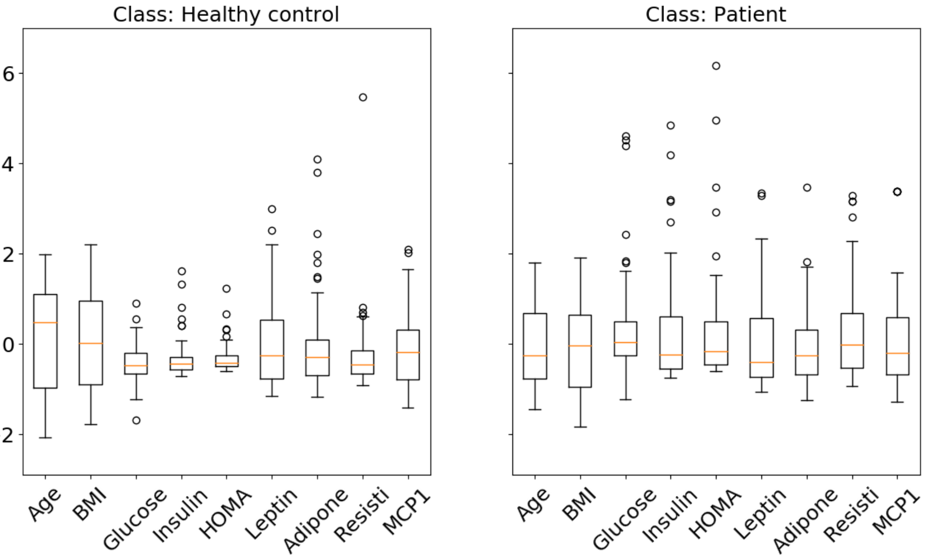


Figure 3: Boxplots of standardized data. In each box the orange line represents the median, the box extends from the 25th (Q1) to the 75th (Q3) percentile of the data, while the upper whisker extend from the box (Q3) to the largest data point less than 1.5 times the inter-quartile range (Q3-Q1). Similarly the lower whisker extend from the box (Q1) to the smallest data point larger than 1.5 times the inter-quartile range (Q3-Q1). Data outside the whiskers (fliers) are marked with open circles.

The correlation coefficient gives an overall indication of whether two attributes are correlated or not. Figure 5 shows the correlation coefficient for attribute pairs for all individuals. The highest positive correlation is found between "HOMA" and "Insulin" while "Adiponectin" and "BMI" have the largest negative correlation. Interestingly, the "Adiponectin"-"BMI" correlation is strong for the "Healthy control" class while being almost absent for the "Patient" class. Similarly, "HOMA" and "Glucose" correlate more strongly for the "Patient" class compared to the "Healthy control".

To examine correlations between attributes in greater detail and to look for clusters Fig. 6 shows scatter plots of the data projected onto the directions defined by the original attributes. Most noticeably there is a strong and approximately linear relationship between the attributes "HOMA" and "Insulin", which is consistent with the high correlation coefficient between these attributes in Fig. 5. This correlations is interesting for regression purposes, but since the data points for the two classes overlap it is not relevant for classification problems. The Z-score matrix in Appendix A offers little additional insight. At this stage, there seems to be no simple way to distinguish the "Healthy control" class from the "Patient" class.

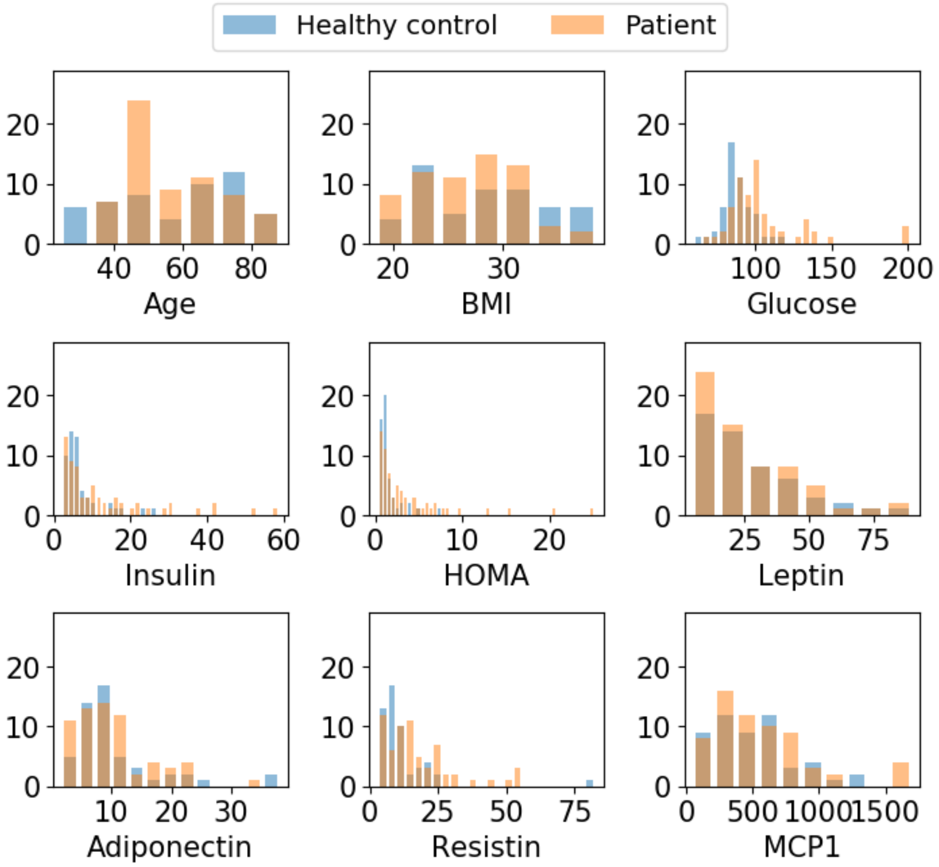


Figure 4: Histogram for each attribute and for each class. The units on the abscissa axes are given in Sec. 1 (Introduction). The ordinate axes show the bin count.

As previously mentioned the regression task of predicting the "HOMA" values from the eight other attributes seems quite feasible due to the high correlation with "Insulin". The classification and clustering tasks, however, are somewhat uncertain at this stage.

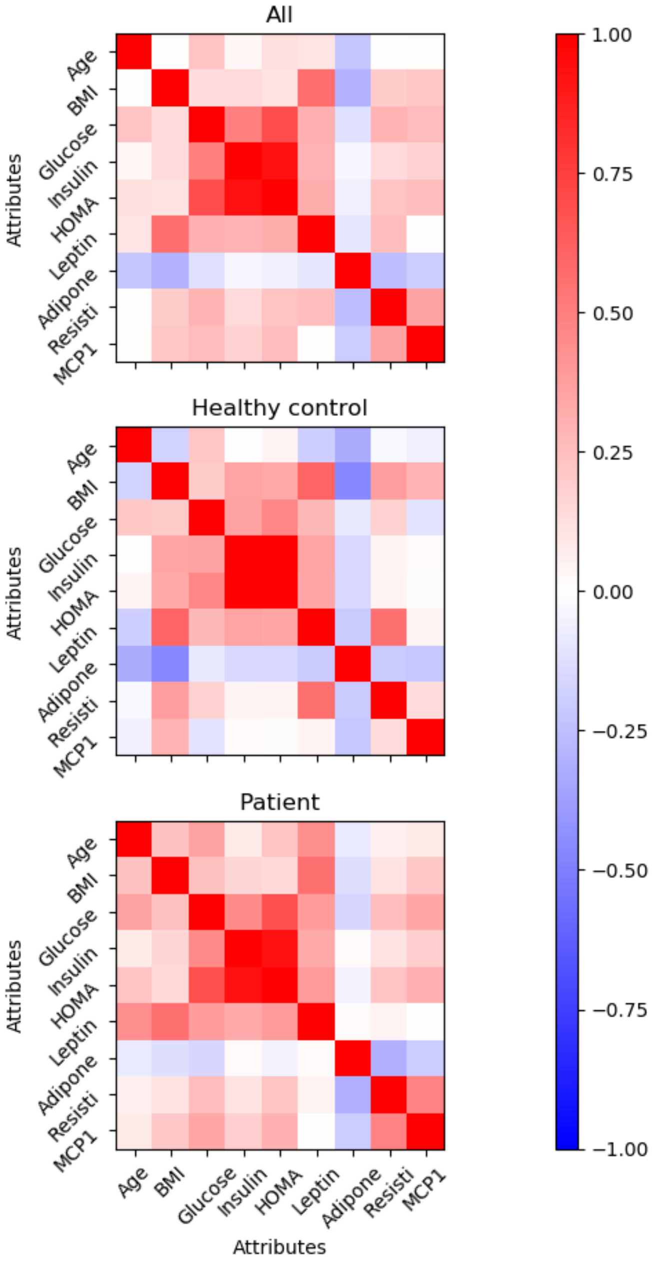


Figure 5: Correlation coefficient matrices for all attributes. The top pane shows correlations for all data, the middle pane shows correlations for the "Healthy control" class while the bottom pane shows correlations the "Patient" class.

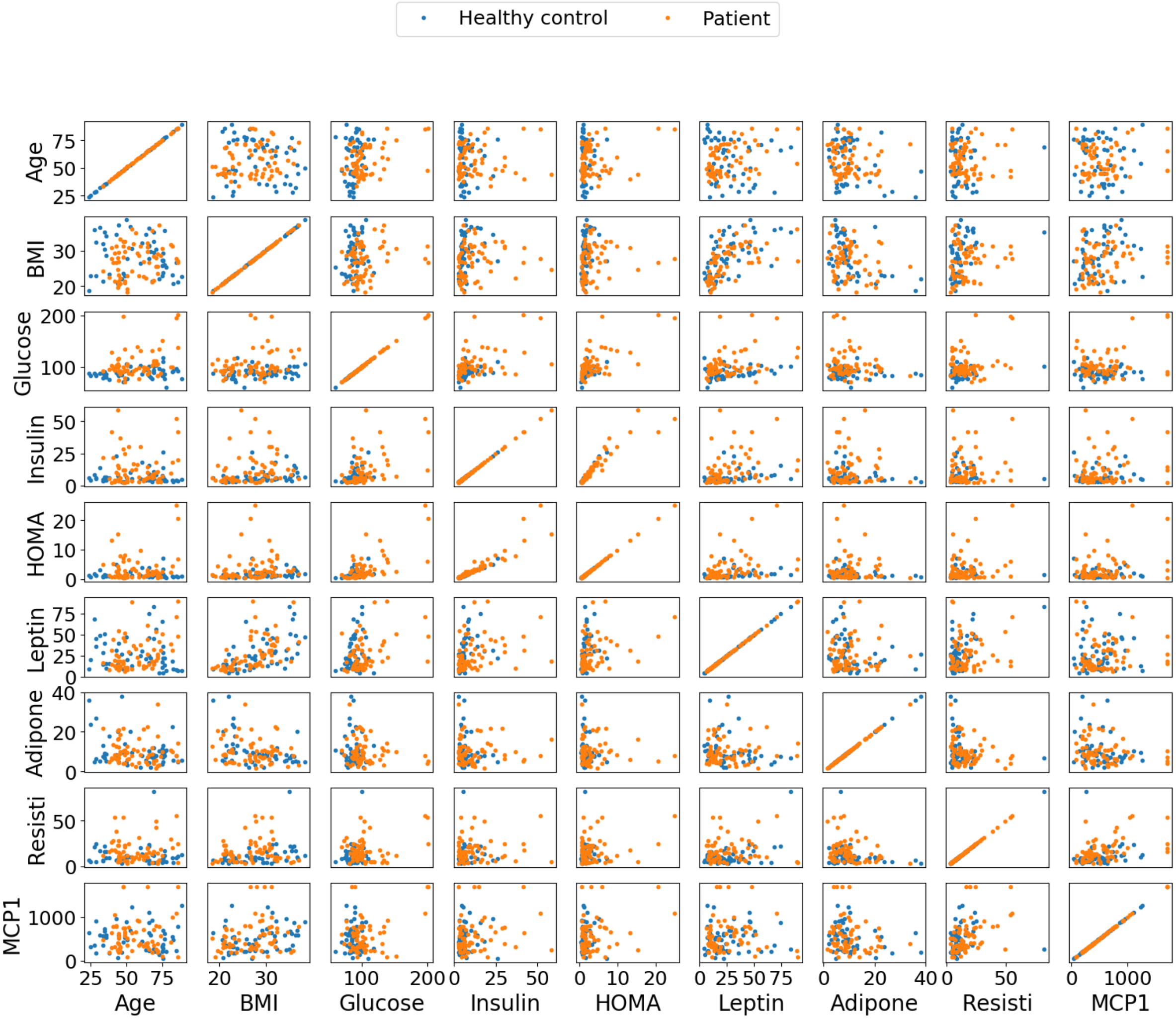


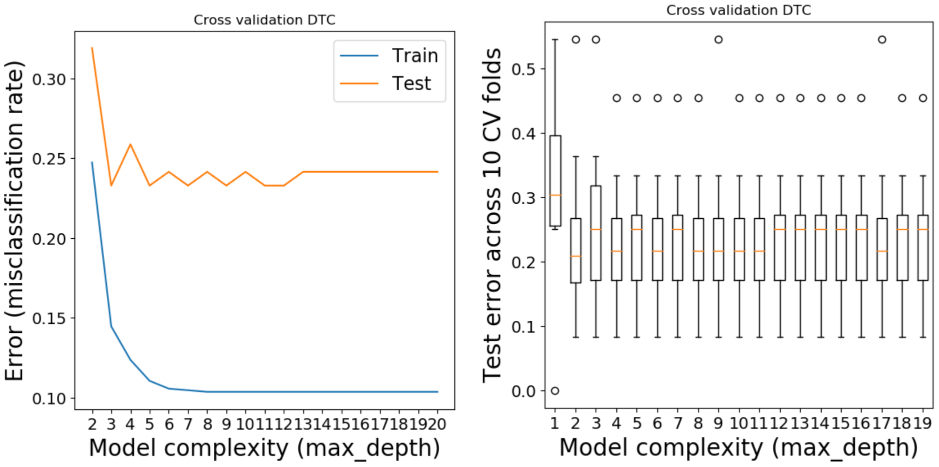
Figure 6: Scatter plot matrix for all attributes. Each plot contains two series corresponding to the "Healthy control" class and the "Patient" class.

4 Classification

4.1 Decision tree with optimal pruning

In an attempt, from a blood sample, to predict whether an individual is healthy or might have breast cancer a decision tree model was trained on the data sample. To determine the pruning level that minimizes the generalization error 10-fold cross-validation was carried out using the maximal tree depth as the pruning parameter. The results of the cross-validation is summarized in Fig. 7. The optimal tree depth is found to be 3.00e+00. The corresponding generalization error of 2.33e-01 cannot be used to estimate the model performance for unseen data.

The optimal tree depth and the estimated generalization error varies between repeated cross-validation runs. These variations are probably due to the data sample being relatively small. Detailed cross-validation data is available from Appendix B along with an illustration of the tree. The attributes included in the optimally pruned tree are "Glucose", "Age", "Resistin", "Insulin" and "BMI".



(a)

(b)

Figure 7: Summary of the decision tree cross-validation. (a) Generalization error for different tree depths. (b) Boxplots for the test error determined in the 10-fold cross-validation. For an explanation of the boxplot see Fig 3.

To see if there is any trend in the misclassifications Fig. 8 shows the predictions for the whole data set using the optimal model determined from Fig. 7. The confusion matrix in Fig. 8 is included to stress the point that the error rate implied in the left pane of Fig. 8 does not approximate the generalization error. In any case, no obvious trend in the misclassifications can be deduced. The same is true when the misclassified data is plotted in the original feature space (see Fig. B.3).

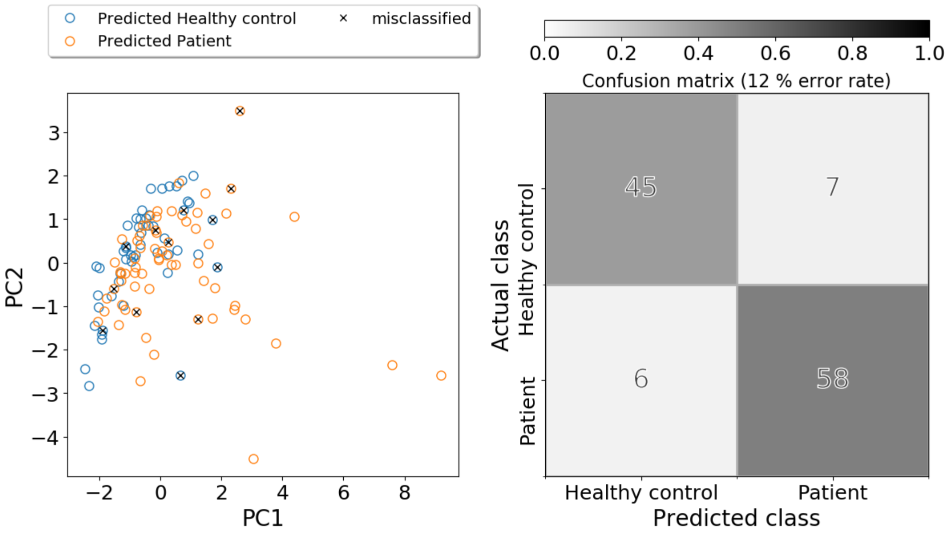


Figure 8: Classification plot for all data using a decision tree with optimal depth. In the left pane the data is projected on the two first principal components. The right pane shows the confusion matrix. Note that the indicated error rate is not representative of the generalization error since much of the data used to predict the error rate in this figure was also used for the training.

4.2 Two-level cross-validation of multiple classifiers

To determine if another classification model may have greater predictive power for this problem the performance of decision trees, K-nearest-neighbors (using the Mahalanobis distance) and Naive Bayes will be compared in this section. Further, the predictive powers of these models will be compared to a model where all outputs are predicted to be identical to the majority class in the training data. To this end an additional model namely `DummyClassifier` with `strategy="most_frequent"` from `sklearn.dummy` was included in the analysis. In an inner cross-validation loop, an optimal complexity will be determined for each model. The target complexity parameter is the tree depth for the decision tree, the number of neighbors for the K-nearest-neighbors and the smoothing parameter for the Naive Bayes. The results from the outer validation loop are summarized in Table 2 and the corresponding summary statistics are presented in Table 3 and in Fig. 9.

The minimal estimated generalization error is 2.24e-01 obtained as a weighted fold-average for the DecisionTree. The best model complexity is `max_depth` = 8 obtained from fold number 9 with test error 9.09e-02. The estimated

Table 2: Test error and value of the model complexity parameter for each tested model and for each fold (columns with headers 0-9). For the DummyClassifier the `most_frequent` strategy was used in all folds. Note that the fold numbers are zero-based in the table.

		0	1	2	3	4	5	6	7	8	9
Model	Property										
DecisionTree	max_depth	8.00	9.00	5.00	3.00	4.00	6.00	4.00	4.00	8.00	7.00
	Test error	0.33	0.17	0.17	0.17	0.17	0.17	0.27	0.36	0.09	0.36
KNeighbors	n_neighbors	3.00	4.00	1.00	1.00	4.00	4.00	1.00	3.00	3.00	4.00
	Test error	0.33	0.33	0.25	0.08	0.42	0.25	0.27	0.27	0.36	0.45
MultinomialNB	alpha	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
	Test error	0.50	0.25	0.75	0.50	0.67	0.50	0.45	0.45	0.36	0.27
Dummy	Test error	0.50	0.42	0.58	0.50	0.33	0.17	0.73	0.36	0.55	0.36

Table 3: Summary statistics over folds for cross-validation for model selection and model complexity adjustment. The column "count" is the number of folds used in the outer cross-validation loop.

		count	mean	std	min	25%	50%	75%	max
Model	Property								
DecisionTree	max_depth	10.00	5.80	2.10	3.00	4.00	5.50	7.75	9.00
	Test error	10.00	0.23	0.10	0.09	0.17	0.17	0.32	0.36
KNeighbors	n_neighbors	10.00	2.80	1.32	1.00	1.50	3.00	4.00	4.00
	Test error	10.00	0.30	0.10	0.08	0.26	0.30	0.36	0.46
MultinomialNB	alpha	10.00	0.01	0.00	0.01	0.01	0.01	0.01	0.01
	Test error	10.00	0.47	0.16	0.25	0.39	0.48	0.50	0.75
Dummy	Test error	10.00	0.45	0.16	0.17	0.36	0.46	0.53	0.73

generalization error 2.24e-01 can be used to estimate the model performance for unseen data.

To evaluate whether the model performances are significantly different Table 4 shows the p-values from from paired t-tests of the model test errors (outer cross-validation loop) for all model pairs. For the particular run used in this report Table 4 shows that on a five percent level the performance of the decision tree is significantly better than the Naive Bayes and the reference model (Dummy). The same is true for the K-nearest-neighbors model although the conclusion is weaker. I have used `scipy.stats.ttest_rel` and not `scipy.stats.ttest_ind` as suggested in Exercise 6.

Note that running the two-level cross-validation multiple times result is quite different performance results. The variation is probably due to the data sample being relatively small.

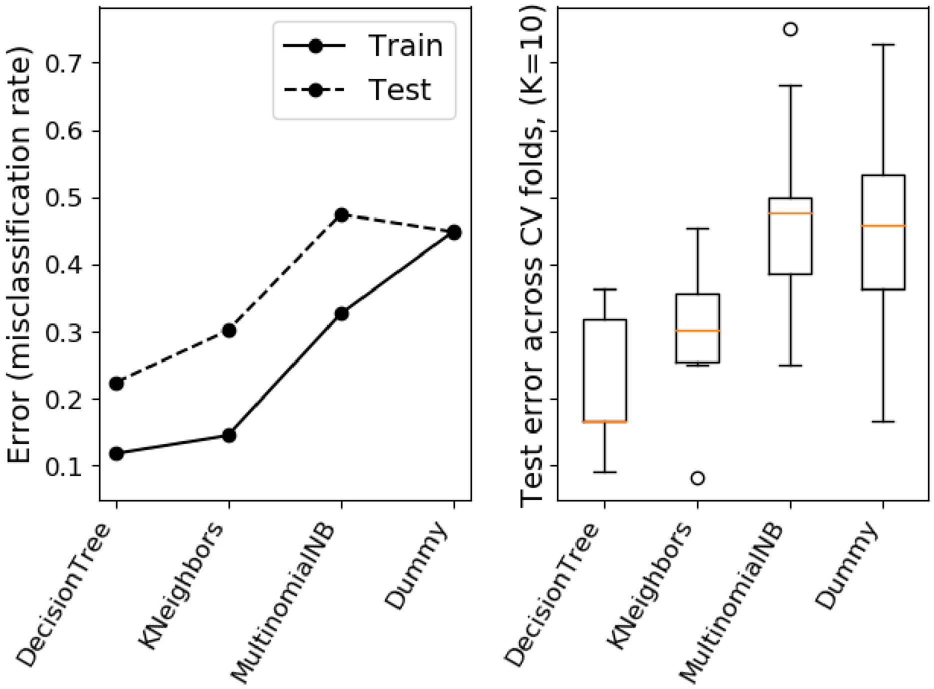


Figure 9: Results from the two-level cross-validation. The left pane shows the generalization error for the model types. The line is inserted to guide the eye. The right pane summarizes the (outer loop) test errors for the classification models as a boxplot. For an explanation of the boxplot see Fig 3.

Table 4: p-values from paired t-tests of the test errors for all model pairs. If the p-value is smaller than α the models are significantly different on level α.

	DecisionTree	KNeighbors	MultinomialNB	Dummy
DecisionTree	nan	0.08	0.00	0.00
KNeighbors	0.08	nan	0.03	0.05
MultinomialNB	0.00	0.03	nan	0.76
Dummy	0.00	0.05	0.76	nan

5 Regression

5.1 Baseline

To establish a baseline for further discussion of the ability to predict "HOMA" from all the remaining attributes linear regression was carried out on the standardized with no further transformations. The standardization allows for interpreting the coefficients directly in terms of sensitivities on "HOMA". As we have seen previously "HOMA" correlate strongly with "Insulin" and this pair turns out to illustrate an interesting point in Section 7: Association mining.

The results from the linear regression are shown in Fig. 10. As expected from the correlation shown in Fig. 5 the largest coefficients are found for the attributes "Insulin" (0.78) and "Glucose" (0.27). The left pane in Fig. 10 shows that the residuals are not independent of the "HOMA" value, which indicates that some transformation may improve the regression fit. Figures 6 and C.4 suggest that by using the squared "Glucose" instead of simply "Glucose" would improve the fit. The results from such a regression are shown in Fig. 11.

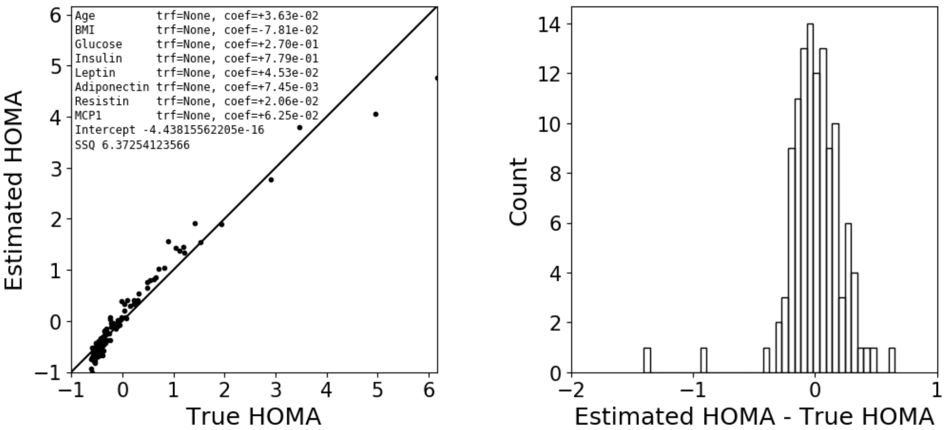
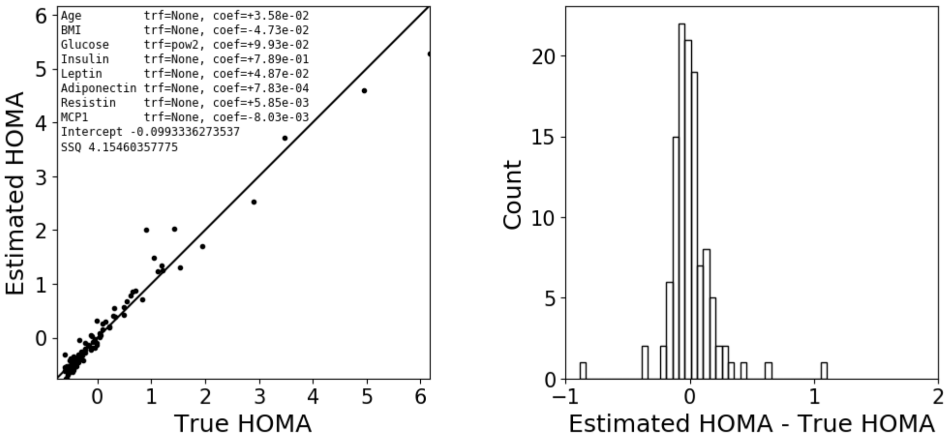


Figure 10: Linear regression of standardized data, with no further transformations. `trf` is an abbreviation for "transformation" and indicates the applied transformation. `coef` is an abbreviation for "coefficient" and is the regression coefficient.

With the transformation of "Glucose", the sum of squared residuals decreases, the residuals are more symmetrically distributed around the mean value and there is no longer an obvious trend in the residuals. These improvements are also obvious when comparing Figs. C.4 and C.5 in Appendix C although a small upward trend may be argued for "Insulin". Thus, it is quite possible that additional transformations or attribute combinations would improve the regression even more, but for this report the regression approach in Fig. 11 is accepted as being sufficiently good.

Note that predicting the "HOMA" value for a new person requires transforming the new data object using the standardization applied above or repeating the regression using the non-standardized data.



```
Age         trf=None, coef=+3.58e-02
BMI         trf=None, coef=-4.73e-02
Glucose     trf=pow2, coef=+9.93e-02
Insulin     trf=None, coef=+7.89e-01
Leptin      trf=None, coef=+4.87e-02
Adiponectin trf=None, coef=+7.83e-04
Resistin    trf=None, coef=+5.85e-03
MCP1        trf=None, coef=-8.03e-03
Intercept -0.0993336273537
SSQ 4.15460357775
```

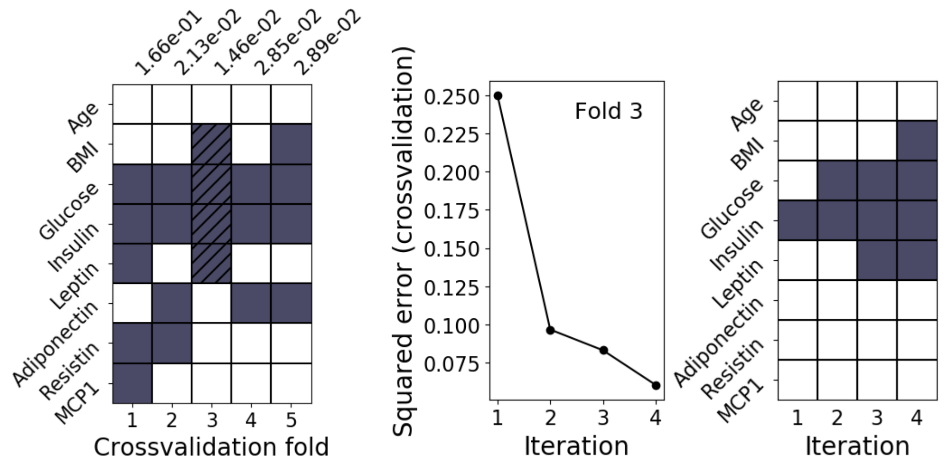
Figure 11: Linear regression of standardized data and using the square of "Glucose" instead of simply "Glucose". `trf` is an abbreviation for "transformation" and indicates the applied transformation. `coef` is an abbreviation for "coefficient" and is the regression coefficient.

5.2 Linear regression with forward selection

Using the square "Glucose" as the only transformation linear regression with forward selection on the standardized data is carried out. Five folds are used in the outer cross-validation loop and 10 folds are used in the inner validation loop. The results are summarized in Fig. 12.

Figure 12a shows the features that give rise to the lowest test error are "BMI", "Glucose", "Insulin" and "Leptin". Again, note that "Glucose" is actually "Glucose" squared. This combination is found in fold 3 with a test error of 1.46e-02. The selection of first "Insulin" and then "Glucose" as the first two attributes is not surprising since these attributes have the numerically largest coefficients in Fig. 11 where all features are included in the regression. When running the feature selection multiple times "Insulin" and "Glucose" are always selected, while the remaining attributes being selected (if any) vary. The same tendency is also observed when looking at the cross-validation folds in Fig. 12a.

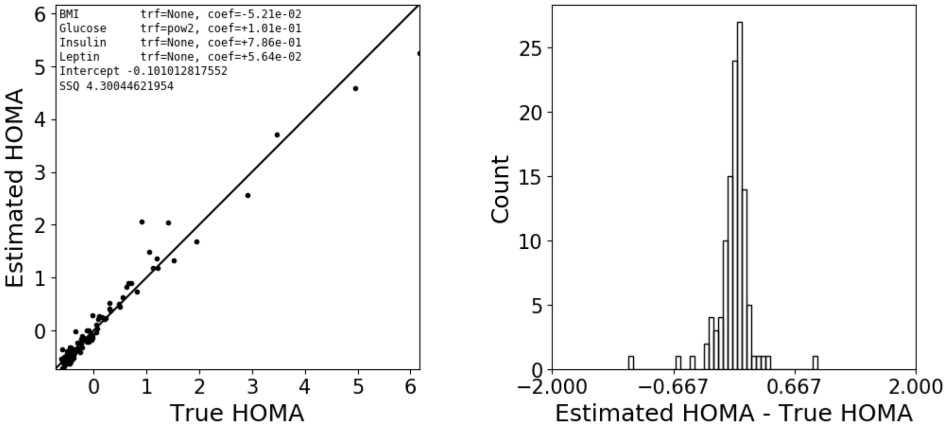
The prediction for the model with features "BMI", "Glucose", "Insulin" and "Leptin" is illustrated in Fig. 13. Note that the coefficients for "Insulin" and square "Glucose" are essentially unchanged when comparing the corresponding coefficients in Fig. 11 while the sum of square residuals have only increased minutely when reducing the feature space from nine to four attributes.
The curious reader can see the residuals from the linear regression with feature selection in Fig. D.6 in Appendix D.



(a)

(b)

Figure 12: Linear regression with forward feature selection. (a) Selected features in each fold shown with dark squares. The hatched squares mark the feature combination that gives rise to the lowest test error. Test error for each fold is shown above the plot. (b) Example of the cross-validation error in the inner loop iteration and the corresponding progression in the forward feature selection for the fold with the lowest test error. Note that the attribute "Glucose" is actually the square of "Glucose".



```
BMI          trf=None, coef=-5.21e-02
Glucose      trf=pow2, coef=+1.01e-01
Insulin      trf=None, coef=+7.86e-01
Leptin       trf=None, coef=+5.64e-02
Intercept -0.101012817552
SSQ 4.30044621954
```

Figure 13: Linear regression of standardized data and using the square of "Glucose" instead of simply "Glucose" and using the features selected by forward selection. `trf` is an abbreviation for "transformation" and indicates the applied transformation. `coef` is an abbreviation for "coefficient" and is the regression coefficient.

5.3 Logistic regression and artificial neural network

This section presents a comparison of the classification performance obtained by logistic regression with the classification performance obtained by a multilayer perceptron classifier. First, however, to give an impression of the logistic regression model, Fig. 14 (left) shows the results from training a logistic regression model on the whole data sample and then predicting the probability of being healthy using that model on the whole data sample. The right pane shows the nine coefficients. The coefficient for each attribute is the logarithm of the odds-ratio for that attribute on the likelihood of being a patient.

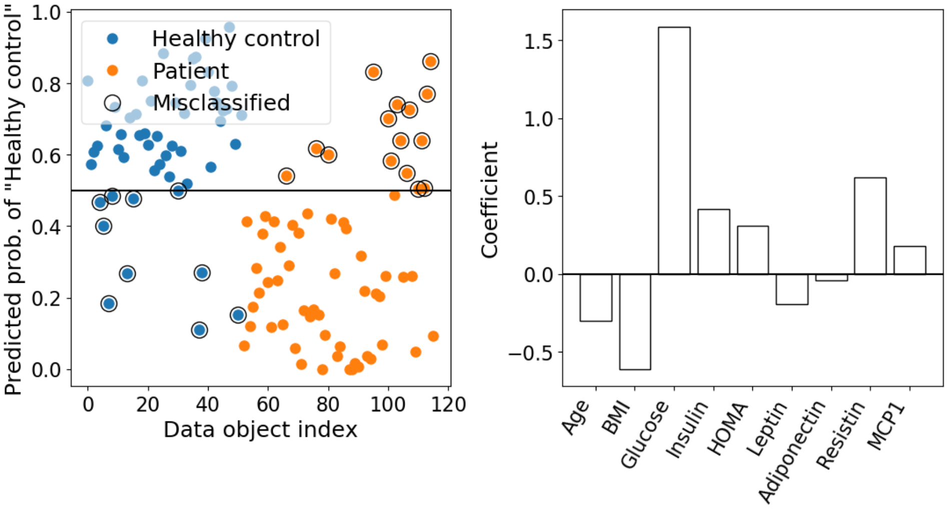


Figure 14: Logistic regression of standardized data. Left: predicted probabilities of being healthy for each data object in the data sample. Right: Logistic regression coefficients.

Fig. 14 (right) shows that high values of "Glucose", "Insulin", "HOMA" and "Resistin" increase the likelihood of belonging to the "Patient" class. Conversely, high values of "BMI" and "Age" increase the likelihood of belonging to the "Healthy control" class. Note that we have not included any feature transformations or feature combinations, except for scaling, in the regression.

As for the classification models the predictive powers of the two regression models will be compared to a model where all outputs are predicted to be identical to the majority class in the training data. To this end an additional model namely `DummyClassifier` with `strategy="most_frequent"` from `sklearn.dummy` will be included in the analysis. Further, in an inner cross-validation loop, the optimal complexity will be determined for each model. The target complexity parameter is the inverse of regularization strength for the logistic regression, and the number of hidden units for the multilayer perceptron classifier. The results from the outer validation loop are summarized in Table 5 and the corresponding summary statistics are presented in Table 6 and in Fig. 15.

Table 5: Test error and value of the model complexity parameter for each tested model and for each fold (columns with headers 0-9). For the DummyClassifier the `most_frequent` strategy was used in all folds (not shown).

		0	1	2	3	4	5	6	7	8	9
Model	Property										
LogisticRegression	C	23.36	3.36	2.07	0.48	0.48	5.46	14.38	0.01	100.00	8.86
	Test error	0.33	0.25	0.17	0.33	0.33	0.33	0.36	0.45	0.09	0.18
MLPClassifier	hidden_layer_sizes	5.00	8.00	5.00	5.00	8.00	5.00	5.00	5.00	9.00	5.00
	Test error	0.25	0.25	0.50	0.33	0.33	0.33	0.27	0.27	0.36	0.18
Dummy	Test error	0.50	0.25	0.33	0.67	0.42	0.42	0.45	0.55	0.45	0.45

Table 6: Summary statistics over folds for cross-validation for model selection and model complexity adjustment. The column "count" is the number of folds used in the outer cross-validation loop.

		count	mean	std	min	25%	50%	75%	max
Model	Property								
LogisticRegression	C	10.00	15.85	30.49	0.01	0.88	4.41	13.00	100.00
	Test error	10.00	0.28	0.11	0.09	0.20	0.33	0.33	0.46
MLPClassifier	hidden_layer_sizes	10.00	6.00	1.63	5.00	5.00	5.00	7.25	9.00
	Test error	10.00	0.31	0.09	0.18	0.26	0.30	0.33	0.50
Dummy	Test error	10.00	0.45	0.11	0.25	0.42	0.46	0.49	0.67

To evaluate whether the model performances are significantly different Table 7 shows the p-values from from paired t-tests of the model test errors (outer cross-validation loop) for all model pairs.

Table 7: p-values from paired t-tests of the test errors for all model pairs. If the p-value is smaller than α the models are significantly different on level α.

	LogisticRegression	MLPClassifier	Dummy
LogisticRegression	nan	0.63	0.00
MLPClassifier	0.63	nan	0.02
Dummy	0.00	0.02	nan

Running the two-level cross-validation multiple times result is quite different performance results. This variation is probably due to the data set being

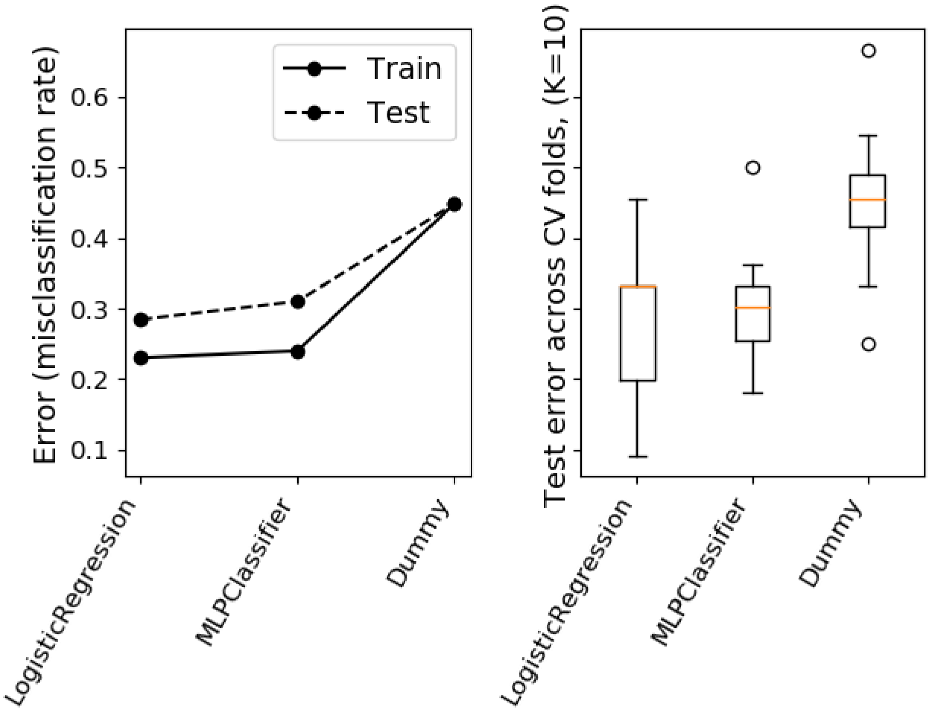


Figure 15: Results from the two-level cross-validation. The left pane shows the generalization error for the model types. The line is inserted to guide the eye. The right pane summarizes the (outer loop) test errors for the classification models as a boxplot. For an explanation of the boxplot see Fig 3.

relatively small. For the particular run used in this report Table 7 shows that on a five percent level the performance of the logistic regression and the multilayer perceptron are not significantly different and that the generalization error is just below 30 percent. Both models are significantly better than the reference (Dummy) model.

6 Clustering

6.1 Hierarchical clustering

Since hierarchical clustering is scaling-sensitive I have tried the method using both original and the standardized data. The scaled data performs better and therefore only results from these analyses will be shown. Figure 16 shows the class labels from the data sample along with the cluster labels that result from a hierarchical clustering with maximum two clusters. The clustering is carried out using "complete" linkage and the "correlation" distance as the metric. This combination of linkage and metric was the best combination (maximizing the Jaccard score) found by manual trial and error.

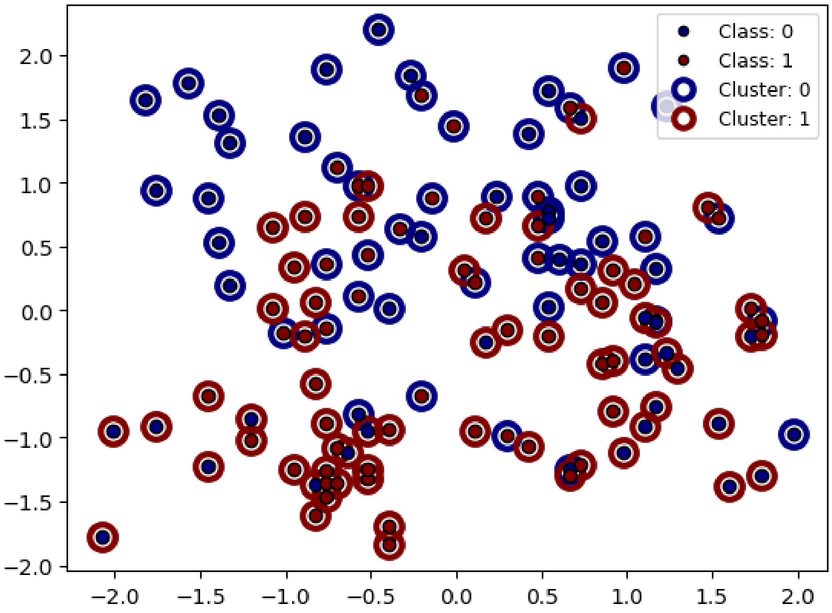


Figure 16: Class and cluster labels. The class labels originate from the data sample while the cluster labels are the result of a hierarchical clustering with maximum two clusters using "complete" linkage and the "correlation" distance as the metric.

The dendrogram belonging to the clustering in Fig. 16 is shown in Fig. 17. Note that the vertical (distance) jumps are relatively small going from 7 to 6 to 5 to 4 to 3 to 2 clusters i.e. there are no late merges that combine clusters far apart. This indicates that there is no natural separation into clusters for the linkage and metric used.

Since class information is available the optimal number of clusters can be determined and Fig. 18 thus quantifies the cluster validity for different values of the maximal number clusters. The adjusted Rand and the Jaccard scores both have a maximum at two clusters. The normalized mutual information score more or less steadily increases for an increasing number of clusters. The adjusted Rand score is larger than zero for all cluster numbers indicating that the clustering is better than random.
To calculate the Jaccard score I have used `jaccard_similarity_score` from `sklearn.metrics` and not the `clusterval` function in the course toolbox since these give different results. The toolbox implementation of the Jaccard similarity does not give a maximum at two clusters.

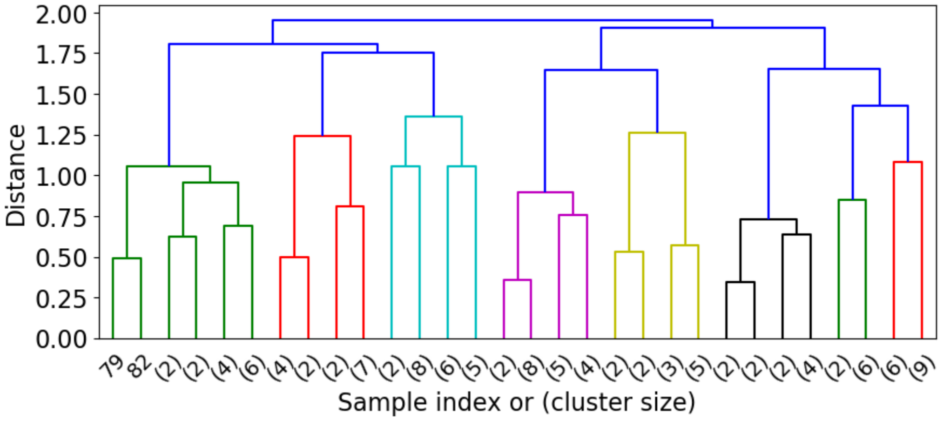


Figure 17: Dendrogram from a hierarchical clustering with maximum two clusters using "complete" linkage and the "correlation" distance as the metric.

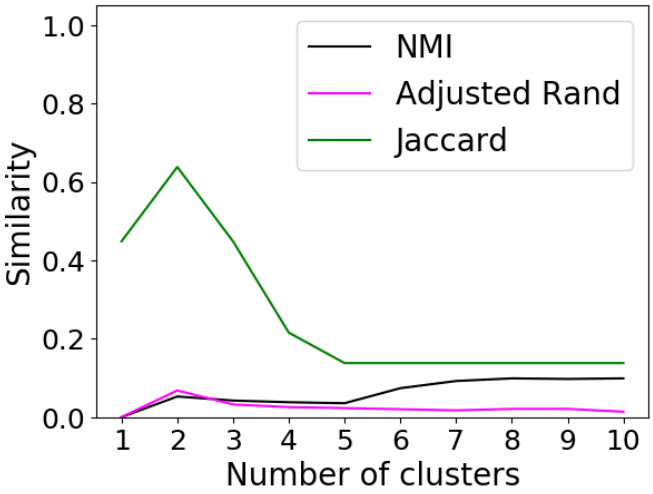


Figure 18: Cluster validity quantified through the Adjusted Rand index, the Jaccard similarity and the normalized mutual information (NMI) when compared to the classes in the data sample.

6.2 Gaussian mixture model

In this section the clustering of the data is analyzed using a Gaussian mixture model (GMM). Unlike the hierarchical clustering the Gaussian mixture model allows for cross-validation, thus determining the best number of clusters with no prior knowledge of the classes. Figure 19 shows that the minimal cross-validation error corresponds to two clusters. In this analysis the Bayesian Information Criteria (BIC) and Akaike's Information Criteria (AIC) are not included in the determination of the optimal number of clusters and therefore that number is two.

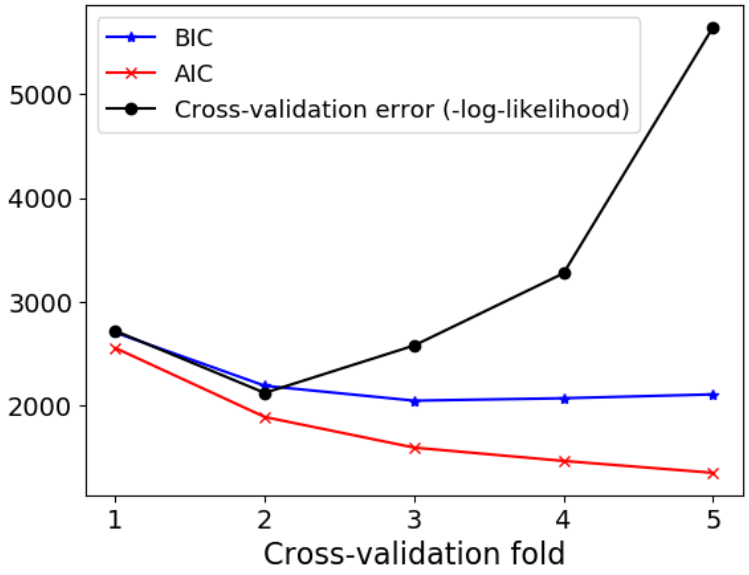
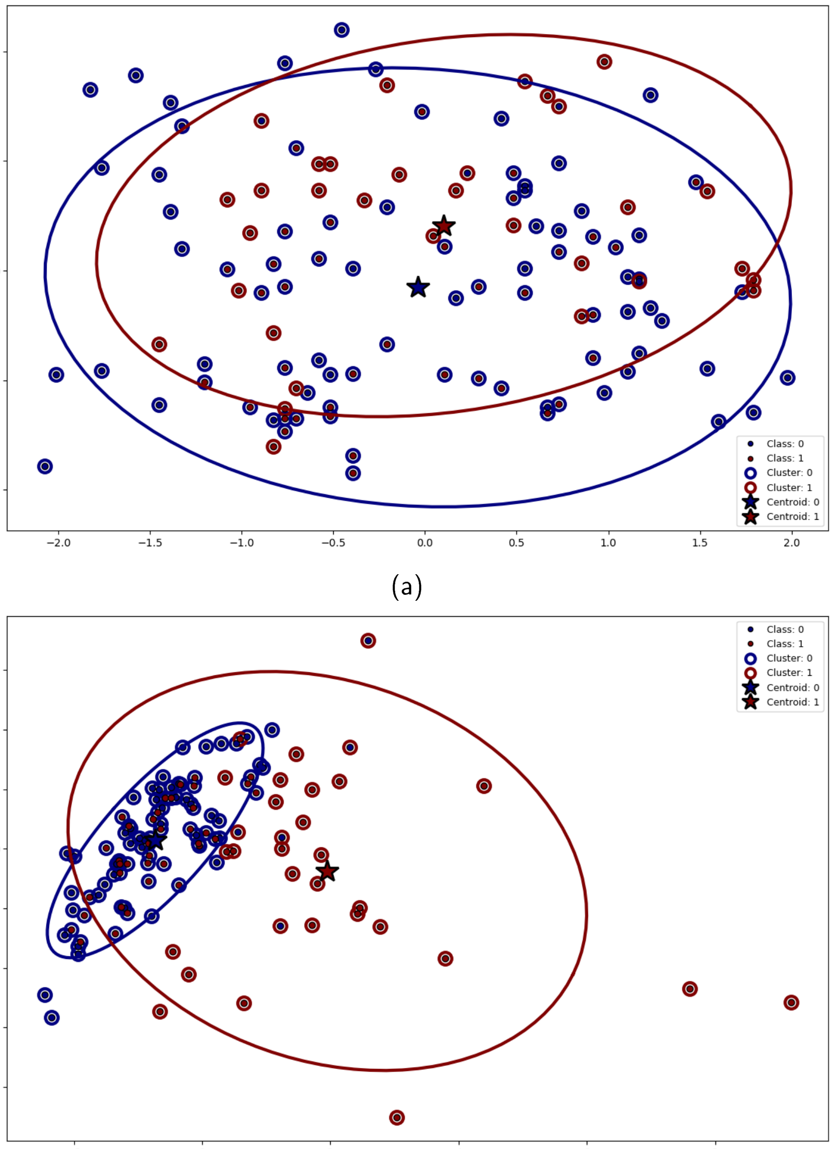


Figure 19: Cross-validation error, Bayesian Information Criteria (BIC) and Akaike's Information Criteria (AIC) obtained from 5-fold cross-validation of a Gaussian mixture model (GMM). Fold 1 corresponds to 1 component in the GMM, fold 2 corresponds to 2 components in the GMM etc..

Figure 20a shows an example of the two Gaussian components represented as ellipses at two standard deviations from the mean in a coordinate system defined by two standardized attributes ("Age", "BMI"). In this projection it is not entirely clear how the two Gaussian components were constructed. Figure. 20b, which shows the same GMM projected on the two first principal components, more clearly illustrates how the Gaussians capture some of the structure in the underlying classes. Using two components the adjusted Rand score is 0.068, the Jaccard score is 0.64 and the NMI is 0.11. These validity scores are approximately the same as previously found by hierarchical clustering.



(a)

(b)

Figure 20: Data classes and clusters as predicted by a two-component Gaussian mixture model. The two Gaussian components are represented as ellipses at two standard deviations from the mean. (a) Projection on the standardized "Age" (abscissa) and standardized "BMI" (ordinate). (b) Projection on the two first principal components.

6.3 Outlier scoring

A Gaussian kernel density estimator may be used to estimate the probability density given data. Such a kernel density estimate may, in turn, be used to identify outliers since data points corresponding to a low density may, according to certain definitions, be labeled as outliers. To illustrate the technique Fig. 21 shows the densities for the 20 data points with the lowest density as predicted by a Gaussian kernel density estimate (KDE) with default width. Figure 22 shows the same data points in a boxplot and the data is also presented Table 8.

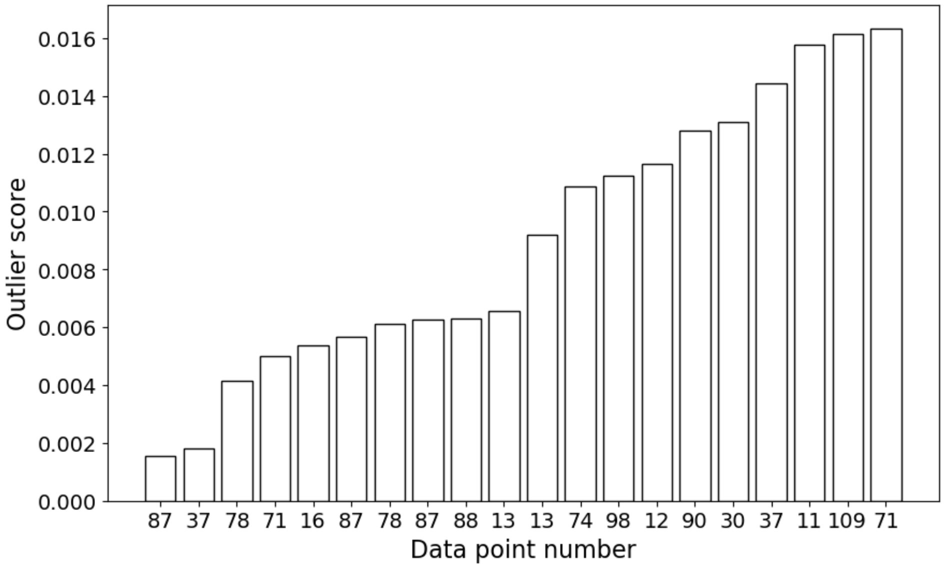


Figure 21: Outlier score for 20 data points corresponding to the 20 lowest densities from a Gaussian kernel density estimate on the data with default width.

To select the optimal width parameter for the kernel the function `gausKernelDensity` from the course toolbox is called with different widths. The optimal width maximizes the log-density for all observations, which is essentially an output of `gausKernelDensity`. The results are summarized in Fig. 23.

To make a robust outlier prediction I will compare the KDE outlier score, the K-nearest neighbor density outlier score and the K-nearest neighbor average relative density outlier score. The K-nearest neighbor density is calculated as the inverse average distance to the $K-1$ nearest neighbors. The K-nearest neighbor average relative density is the inverse average distance to the $K-1$ nearest neighbors relative to the average of the K-nearest neighbor density of the $K-1$ neighbors. The results from the three neighbor-based scores are shown in Figs 23 through 25.

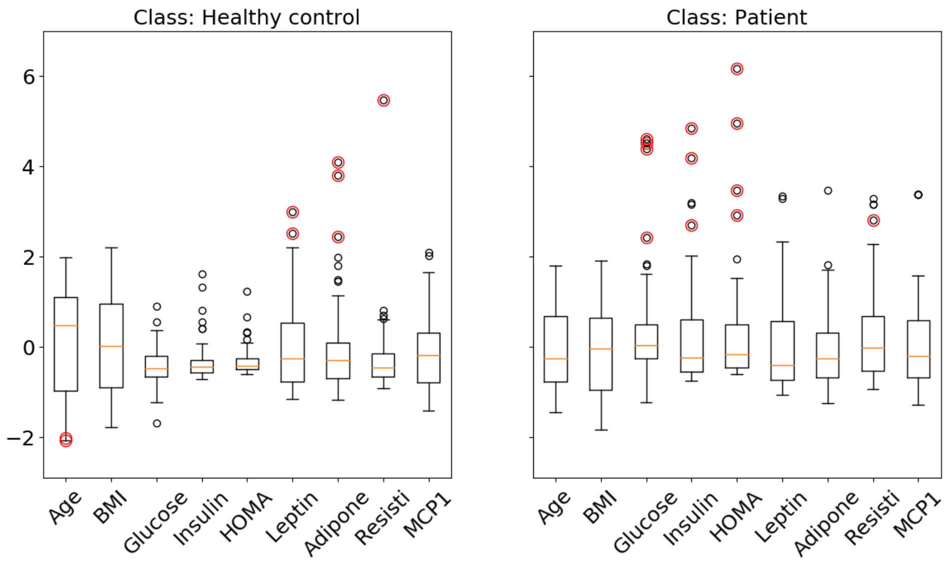


Figure 22: Boxplot of the data as Fig. 3. In addition to the boxplot 20 data points corresponding to the 20 lowest densities from a Gaussian kernel density estimate on the data have been marked by red open circles.

Table 8: Outlier score for 20 data points corresponding to the 20 lowest densities from a Gaussian kernel density estimate on the data with default width. The column "Value" is the value of the "Attribute".

Attribute	Class	Data index	Score	Value
HOMA	Patient	87	0.0016	6.1648
Resistin	Healty	37	0.0018	5.4611
HOMA	Patient	78	0.0042	4.9460
Insulin	Patient	71	0.0050	4.8331
Adiponectin	Healty	16	0.0054	4.0886
Insulin	Patient	87	0.0057	4.1701
Glucose	Patient	78	0.0061	4.6017
Glucose	Patient	87	0.0063	4.3788
Glucose	Patient	88	0.0063	4.5126
Adiponectin	Healty	13	0.0066	3.7981
Age	Healty	13	0.0092	-2.0758
Insulin	Patient	74	0.0109	2.6863
Resistin	Patient	98	0.0112	2.7977
Age	Healty	12	0.0117	-2.0134
HOMA	Patient	90	0.0128	2.9044
Leptin	Healty	30	0.0131	2.5178
Leptin	Healty	37	0.0144	2.9773
Adiponectin	Healty	11	0.0158	2.4273
Glucose	Patient	109	0.0162	2.4169
HOMA	Patient	71	0.0163	3.4719

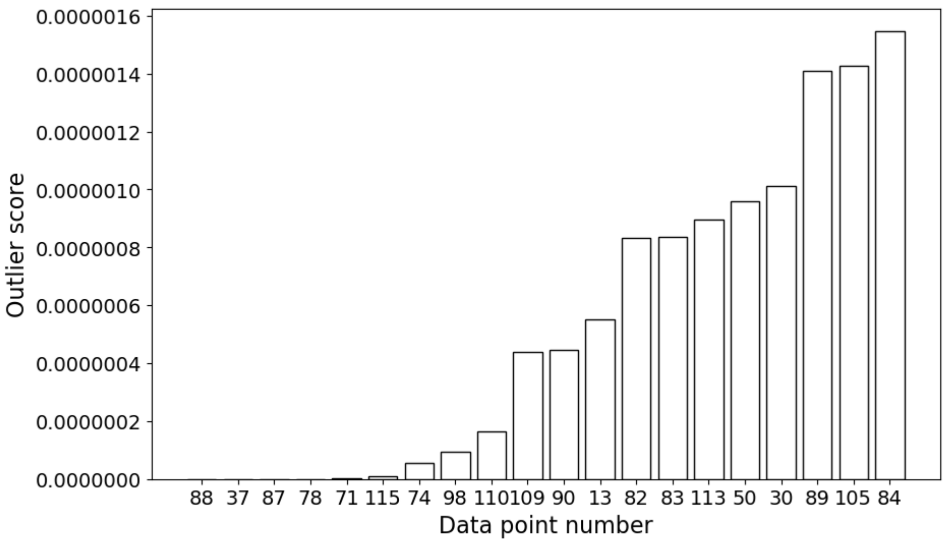


Figure 23: Outlier score for 20 data points corresponding to the 20 lowest densities from a Gaussian kernel density estimate on the data with optimal width 0.5 obtained by maximizing the log-density for all observations.

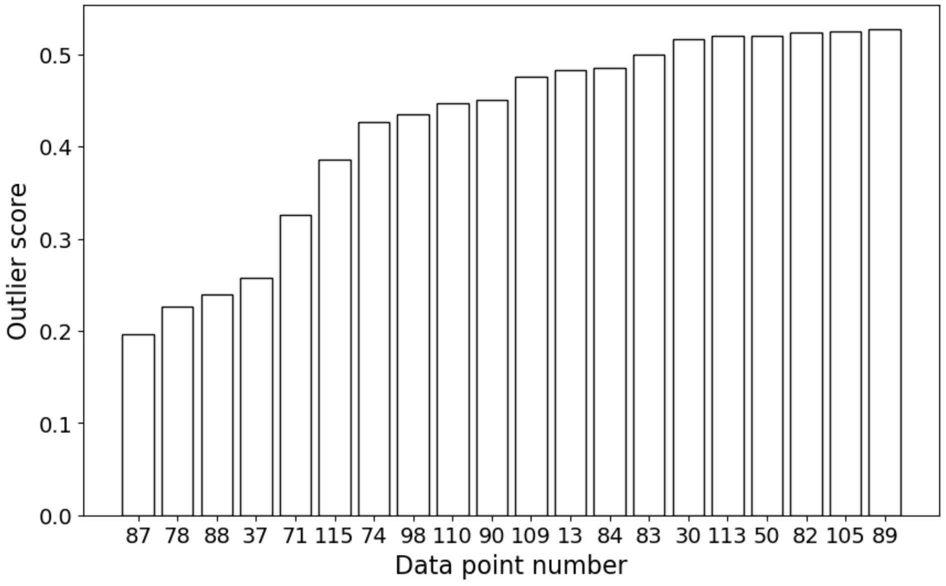


Figure 24: Outlier score for 20 data points corresponding to the 20 lowest 5-nearest neighbor densities (see text for details). Using 5-nearest neighbors is somewhat arbitrary.

The intersection of the top 20 data points with lowest scores from the three outlier scoring methods are 13, 30, 37, 50, 71, 78, 87, 88, 98, 109, 110, 113,

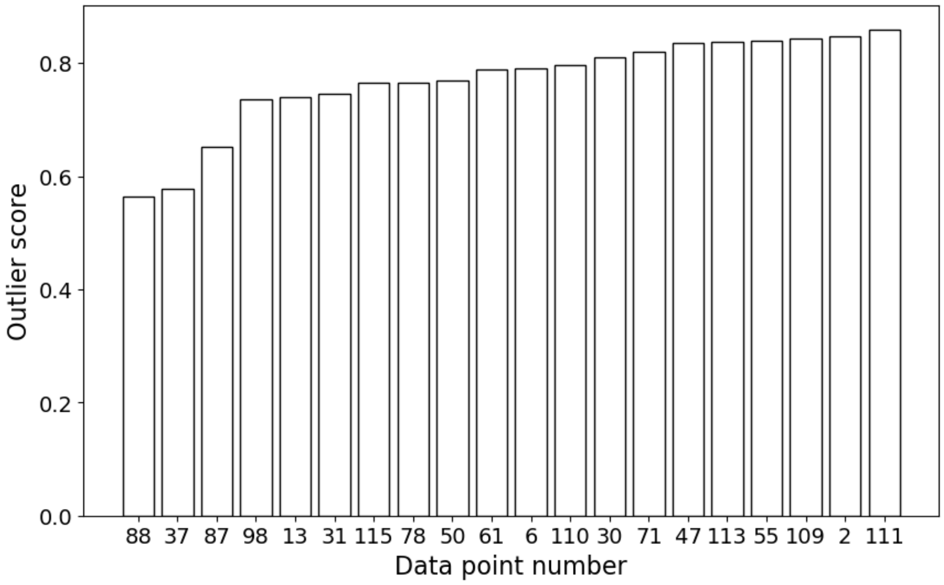


Figure 25: Outlier score for 20 data points corresponding to the 20 lowest 5-nearest neighbor average relative densities (see text for details). Using 5-nearest neighbors is somewhat arbitrary.

115. These data objects will be considered outliers. Figure 26 shows the outliers projected on the first two principal components. Since the putative outliers were included in the pricipal component analysis and the first two principal components account for about 50 percent of the variance in the data this projection should render outliers quite nicely. Even so, some data objects do not look like outliers in this projection. Consider for example the data object with index 98. Table 8 shows that this data object has a standardized "Resistin" value of 2.7977 and belongs to the "Patient" class. With this identification we may locate the data point in Fig. 22 where it qualitatively looks like an outlier.

Before conclusively labeling the above list of data objects as outliers I would investigate common ranges for the studied attributes and look carefully into any differences in accuracy for different blood analysis methods. Having identified outliers I would then run all analyses in this report again i.e. I would start with this outlier analysis. Such investigations are, however, beyond the scope of this report.

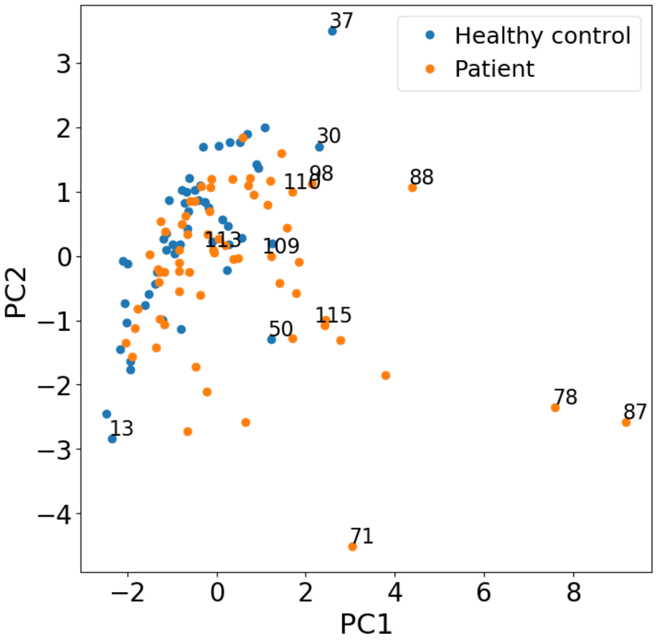


Figure 26: Data shown in a coordinate system defined by the two first principal components. Putative outliers, as jointly identified by the Gaussian kernel density, the 5-nearest neighbor density and by the 5-nearest neighbor average relative densities, are annotated with their index in the data sample.

7 Association mining

In association mining frequent items are identified as well as association rules. Before the data can be mined continuous attributes need be transformed into categorical attributes, which can e.g. be achieved using the function `binarize2` from the course toolbox. This function splits each continuous attribute into two categories at the median, assigns each value to the appropriate category and then transforms each of these binary attribute columns into two columns using the one-out-of-K coding. Further, to include the class label column with 0s corresponding to "Healthy control" and 1s corresponding to "Patient" the class label column is transformed into two columns using the one-out-of-K coding format (`OneHotEncoder` from `sklearn.preprocessing`).

The frequent items found running `apriori.exe` with `minSup` = 35, `minConf` = 65 and `maxRule` = 4 are shown below. There are 28 items eight of which are attribute pairs. As expected when binarizing at the median all the single-attribute frequent items (20 as expected) have a support of about 50 %. Looking at the frequent attribute pairs the results from the correlation analysis and linear regression are confirmed namely that high "HOMA" values are found with high "Insulin" values in 47 % of the data objects (`HOMA Q2+ Insulin Q2+[Sup. 47]`). Likewise for the corresponding low values. More interestingly `Glucose Q2+ Patient[Sup. 37]` indicates that 37 % of the data objects are patients and have high "Glucose" values. With a "Patient" support of 55 % this corresponds to 67 % of the patients (37 %/55 %).

```
Frequent itemsets:
Item: Glucose Q2+[Sup. 49]
Item: BMI Q2+ Leptin Q2+[Sup. 38]
Item: Leptin Q2- BMI Q2-[Sup. 38]
Item: Glucose Q2+ Patient[Sup. 37]
Item: HOMA Q2- Insulin Q2-[Sup. 47]
Item: HOMA Q2+ Insulin Q2+[Sup. 47]
Item: HOMA Q2- Glucose Q2-[Sup. 36]
Item: Patient[Sup. 55]
Item: Resistin Q2+ Patient[Sup. 35]
Item: Glucose Q2+ HOMA Q2+[Sup. 35]
Item: Healthy control[Sup. 45]
Item: Glucose Q2-[Sup. 51]
Item: MCP1 Q2-[Sup. 50]
Item: HOMA Q2+[Sup. 50]
Item: Age Q2-[Sup. 50]
Item: BMI Q2-[Sup. 50]
Item: Insulin Q2-[Sup. 50]
Item: BMI Q2+[Sup. 50]
Item: HOMA Q2-[Sup. 50]
Item: Leptin Q2+[Sup. 50]
Item: Leptin Q2-[Sup. 50]
Item: Age Q2+[Sup. 50]
Item: Insulin Q2+[Sup. 50]
Item: Resistin Q2+[Sup. 50]
Item: Resistin Q2-[Sup. 50]
Item: MCP1 Q2+[Sup. 50]
Item: Adiponectin Q2+[Sup. 50]
Item: Adiponectin Q2-[Sup. 50]
```

The corresponding association rules are shown below.

```
Association rules:
Rule: Glucose Q2+ <- Patient[Conf. 67,Sup. 37]
Rule: Leptin Q2- <- BMI Q2-[Conf. 76,Sup. 38]
Rule: BMI Q2- <- Leptin Q2-[Conf. 76,Sup. 38]
Rule: BMI Q2+ <- Leptin Q2+[Conf. 76,Sup. 38]
Rule: Leptin Q2+ <- BMI Q2+[Conf. 76,Sup. 38]
Rule: HOMA Q2- <- Insulin Q2-[Conf. 95,Sup. 47]
Rule: Insulin Q2- <- HOMA Q2-[Conf. 95,Sup. 47]
Rule: HOMA Q2+ <- Insulin Q2+[Conf. 95,Sup. 47]
Rule: Insulin Q2+ <- HOMA Q2+[Conf. 95,Sup. 47]
Rule: Patient <- Glucose Q2+[Conf. 75,Sup. 37]
Rule: Glucose Q2- <- HOMA Q2-[Conf. 72,Sup. 36]
Rule: HOMA Q2+ <- Glucose Q2+[Conf. 72,Sup. 35]
Rule: HOMA Q2- <- Glucose Q2-[Conf. 71,Sup. 36]
Rule: Patient <- Resistin Q2+[Conf. 71,Sup. 35]
Rule: Glucose Q2+ <- HOMA Q2+[Conf. 71,Sup. 35]
```

The topmost rule `Glucose Q2+ <- Patient[Conf. 67,Sup. 37]` is actually the rule described above. The inverted rule `Patient <- Glucose Q2+[Conf. 75,Sup. 37]` indicates that with 75 % confidence you are a patient if your "Glucose" level is high. Note the high confidence on the rules that involve the attribute pair "HOMA" and "Insulin" that were found to have a large linear coefficient.

To find more features and feature combinations that are linked to being a patient the support and confidence need be reduced. Before proceeding it is illustrative to consider the probability interpretation of the support s.

$$s(Y \leftarrow X) = p(Y \cap X) = p(Y|X)p(X) \begin{cases} < p(Y)p(X) & Y \text{ less likely given } X \\ = p(Y)p(X) & X \text{ \& } Y \text{ independent} \\ > p(Y)p(X) & Y \text{ more likely given } X \end{cases}$$

And the confidence c.

$$c(Y \leftarrow X) = \frac{p(Y \cap X)}{p(X)} = p(Y|X) \begin{cases} < p(Y) & Y \text{ less likely given } X \\ = p(Y) & X \text{ \& } Y \text{ independent} \\ > p(Y) & Y \text{ more likely given } X \end{cases}$$

Since the data was binarized at the meadian a support of more than 25 % for an attribute pair indicates that the pair occurs more frequent than if the two attributes were independent. Further, looking for features that increase the likelihood of being a "Patient", the confidence should be greater than the support for "Patient". Therefore, in an exhaustive search, the minimum support and confidence to be used in the Apriori algorithm should be 25 % and 55 %, respectively. Below only the rules that point to "Patient" are shown when running exhaustively.

```
Association rules indicating Patient:
Rule: Patient <- Leptin Q2-[Conf. 59,Sup. 29]
Rule: Patient <- HOMA Q2+ Insulin Q2+[Conf. 69,Sup. 33]
Rule: Patient <- Glucose Q2+ Insulin Q2+[Conf. 79,Sup. 26]
Rule: Patient <- Glucose Q2+ HOMA Q2+ Insulin Q2+[Conf. 79,Sup. 26]
Rule: Patient <- HOMA Q2+[Conf. 69,Sup. 34]
Rule: Patient <- Glucose Q2+ HOMA Q2+[Conf. 78,Sup. 28]
Rule: Patient <- BMI Q2-[Conf. 57,Sup. 28]
Rule: Patient <- Insulin Q2+[Conf. 66,Sup. 33]
Rule: Patient <- MCP1 Q2-[Conf. 55,Sup. 28]
Rule: Patient <- MCP1 Q2+[Conf. 55,Sup. 28]
Rule: Patient <- [Conf. 55,Sup. 55]
Rule: Patient <- Glucose Q2+[Conf. 75,Sup. 37]
Rule: Patient <- Resistin Q2+[Conf. 71,Sup. 35]
Rule: Patient <- Age Q2-[Conf. 60,Sup. 30]
Rule: Patient <- Adiponectin Q2+[Conf. 60,Sup. 30]
```

Reducing the minimum support reveals feature combination that yield a support of close to 80 %, which is higher than any single-feature rule. Interestingly, most of the single-feature rules agree qualitatively with the coefficients found in the logistic regression (Fig. 14). The mining result could suggest that combining e.g. "HOMA" and "Insulin" in the logistic regression could prove interesting.

It is reassuring to see that the three rules listed below are essentially identical.

```
Rule: Patient <- Glucose Q2+ Insulin Q2+[Conf. 79,Sup. 26]
Rule: Patient <- Glucose Q2+ HOMA Q2+ Insulin Q2+[Conf. 79,Sup. 26]
Rule: Patient <- Glucose Q2+ HOMA Q2+[Conf. 78,Sup. 28]
```

This can easily be understood from the high correlation between "Insulin" and "HOMA" (see e.g. Figs. 5 and 13).

The rule `Patient <- Age Q2-[Conf. 60,Sup. 30]` is somewhat counter intuitive. Even if the rule only has a confidence, which is slightly higher than the support in "Patient" I would have expected the opposite rule (`Patient <- Age Q2+`) to be valid. Either my intuition is not correct or this rule could be an artifact of the selection of individuals for the data sample where the patient group is over-represented compared to the wild.

On the same note, it is important to remember that the confidence of the rules described above apply only to the data sample studied here. In Section 8: Discussion I will show that the patient support of 55 % is (thankfully) not representative of the average incidence rate in the wild.

8 Discussion

The data sample includes 64 patients with breast cancer and 52 healthy controls. In the wild, the breast cancer incidence rate is about 100 per 100,000 women [4, 5]. Further, the U.S. Preventive Services Task Force notes [2] that there are risks associated with recording mammograms and also that false positives result in unnecessary psychological harms:

> *The USPSTF found adequate evidence that screening for breast cancer with mammography results in harms for women aged 40 to 74 years. The most important harm is the diagnosis and treatment of noninvasive and invasive breast cancer that would otherwise not have become a threat to a woman's health, or even apparent, during her lifetime (that is, overdiagnosis and overtreatment). False-positive results are common and lead to unnecessary and sometimes invasive follow-up testing, with the potential for psychological harms (such as anxiety). False-negative results (that is, missed cancer) also occur and may provide false reassurance. Radiation-induced breast cancer and resulting death can also occur, although the number of both of these events is predicted to be low.*

Therefore, it is interesting to discuss how the prediction rates from this report transfer into a real scenario. In this discussion I will disregard the effect of age.

The prediction rates I will use for this example are obtained from fitting decision tree with a maximal depth of three (see Sec. 4.1) using a holdout-splitter with a test proportion of 50 %. The obtained results are summarized in Fig. 27. The errors in Fig. 27 can actually not be used to estimate the model performance for unseen data. But as noted in Section 4.2 the variations in the estimated generalization error are quite big due to the small data sample. Therefore, with little confidence in the estimates, I will proceed and use the errors in Fig. 27 to illustrate how to asses the model performance in the wild.

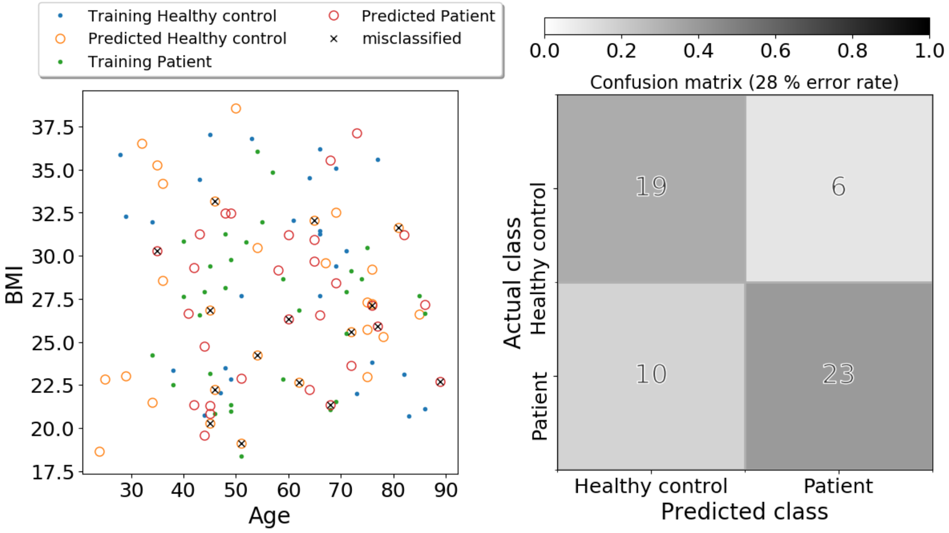


Figure 27: Classification plot for all data using a decision tree with a maximal depth of three. The data sample is split into a test and a training set using a holdout-splitter with a test proportion of 50 %. In the left pane the data is projected on the two first attributes. The right pane shows the confusion matrix.

The conditional probabilities corresponding the the confusion matrix in Fig. 27 are

$$\begin{aligned} P(+|\mathrm{S},\mathrm{A}) &= 0.70 \\ P(-|\mathrm{S},\mathrm{A}) &= 0.30 \\ P(+|\mathrm{H},\mathrm{A}) &= 0.24 \\ P(-|\mathrm{H},\mathrm{A}) &= 0.76 \end{aligned} \tag{1}$$

The symbol "A" marks the data sample population. The symbol "+" is used to indicate a positive test while the symbol "$-$" is used to indicate a negative test. The symbol "S" is used to indicate a a sick individual while the symbol "H" is used to indicate a healthy individual. In the wild population (label B) I will, based on the incidence rate previously mentioned, assume that the probability of being sick is $P(\mathrm{S}|\mathrm{B}) = 0.001$ and that the above conditional probabilities derived in A are also valid in B. Then, using the product rule, the following rates r are obtained for 10.000 women.

$$\begin{aligned} r(+,\mathrm{S}|\mathrm{B}) &= 7 \\ r(-,\mathrm{S}|\mathrm{B}) &= 3 \\ r(+,\mathrm{H}|\mathrm{B}) &= 2398 \\ r(-,\mathrm{H}|\mathrm{B}) &= 7592 \end{aligned} \tag{2}$$

The probability of testing positive in B $P(+|\mathrm{B})$ is (2,398 + 7)/10,000. Using Bayes' theorem the following probabilities may be calculated.

$$\begin{aligned} P(\mathrm{S}|+,\mathrm{B}) &= 2.9 \cdot 10^{-3} \\ P(\mathrm{S}|-,\mathrm{B}) &= 4.0 \cdot 10^{-4} \\ P(\mathrm{H}|+,\mathrm{B}) &\approx 1.0 \\ P(\mathrm{H}|-,\mathrm{B}) &\approx 1.0 \end{aligned} \tag{3}$$

As expected the false-positive rate is quite substantial at 24 %, which is also reflected in the likelihood of being healthy given a positive test $P(\mathrm{H}|+,\mathrm{B})$ is close to unity.

If a blood sample analysis were to be used as a pre-screening method for women that today would all be screened using mammography one would only have to record about 24 % of the mammograms that are recorded today. Considering the price and the risk of radiation induced breast cancer associated with recording mammogram this is a benefit. However, the high false positive rate would most likely increase psychological harms inflicted on the women. Also, three in 10,000 women would not be diagnosed as having breast cancer, which should be compared to the false-negative rate of mammography.

As previously noted there are rather big variations is the predictions of the model from run to run and also uncertainty associated with the optimal model complexity. Therefore, I would only take the analyses presented here as a coarse indication of the applicability of blood samples as a screening tool for breast cancer.

The mining analysis suggests that the most important attributes are the combination of "Glucose" and "HOMA" (or "Insulin"). The properties included in the optimally pruned tree (Appendix B) are "Glucose", "Resistin", "Age", "Insulin" and "BMI". The coefficients resulting from a logistic regression of all data shows that high values of especially "Glucose", "Insulin", and "Resistin" increase the likelihood of belonging to the "Patient" class while high values of especially "BMI" and "Age" increase the likelihood of belonging to the "Healthy control" class. Between these models there is good agreement about which attributes are important for the predictions.

References

[1] mrsonne's github repository. URL https://github.com/mrsonne/ML18.

[2] U.S. Preventive Services Task Force. Final recommendation statement: Breast cancer screening, 2016. URL https://www.uspreventiveservicestaskforce.org/Page/Document/RecommendationStatementFinal/breast-cancer-screening1.

[3] T. Herlau, M.N. Schmidt, and M. Mørup. *Introduction to Machine Learning and Data Mining*. Technical Univerty of Denmark, 2018.

[4] World Cancer Research Fund International. URL https://www.wcrf.org/dietandcancer/cancer-trends/breast-cancer-statistics.

[5] Susan G. Komen Organization. URL https://ww5.komen.org/BreastCancer/Statistics.html.

[6] M. Patrício, J. Pereira, J. Crisóstomo, P. Matafome, M. Gomes, R. Seiça, and F. Caramelo. Using resistin, glucose, age and bmi to predict the presence of breast cancer. *BMC Cancer*, 18, 2018. URL https://bmccancer.biomedcentral.com/articles/10.1186/s12885-017-3877-1.

[7] UCI. Machine learning repository. URL https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Coimbra.

Appendices

A Z-score matrix

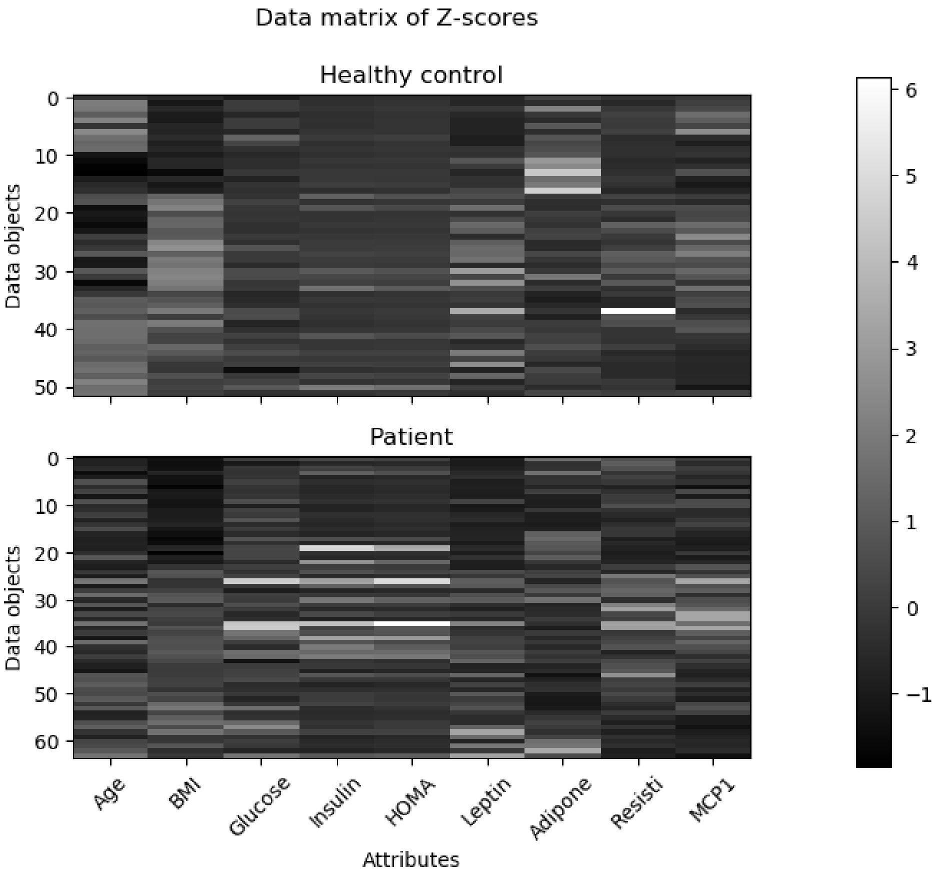


Figure A.1: Z-score matrix.

B Cross-validation of a decision tree classifier

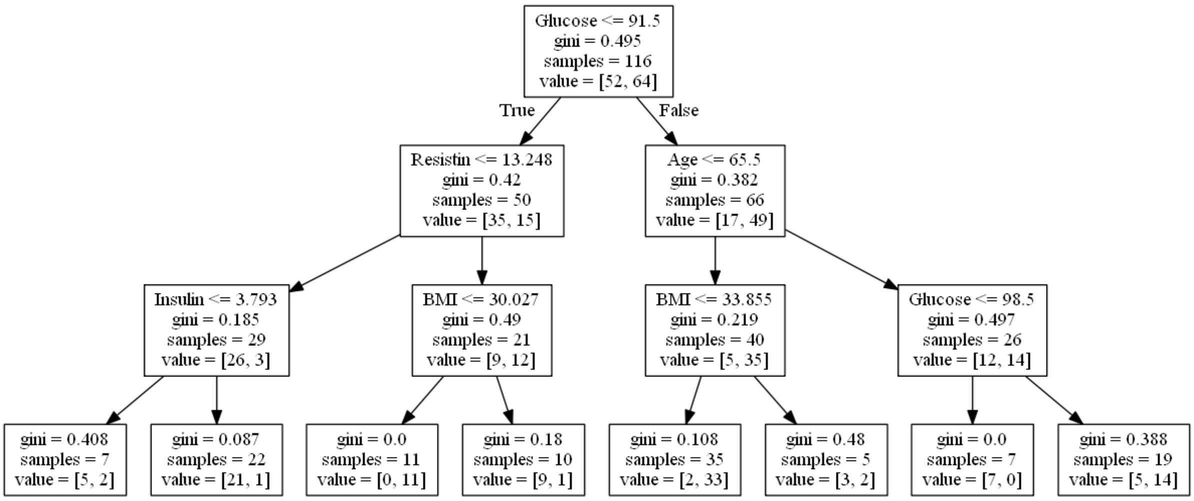


Figure B.2: Decision tree fitted to the standardized data. The optimal depth is determined from 10-fold cross-validation.

Table B.1: Results from the 10-fold cross-validation varying the `max_depth` complexity parameter from 2 to 20.

	0	1	2	3	4	5	6	7	8	9
max_depth										
2	0.50	0.33	0.25	0.25	0.42	0.33	0.27	0.27	0.55	0.00
3	0.25	0.25	0.08	0.17	0.17	0.17	0.27	0.36	0.55	0.09
4	0.25	0.25	0.08	0.33	0.17	0.17	0.27	0.36	0.55	0.18
5	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.18
6	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
7	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.18
8	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
9	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.18
10	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.55	0.18
11	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.18
12	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.18
13	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
14	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
15	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
16	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
17	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
18	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.55	0.18
19	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27
20	0.25	0.25	0.08	0.33	0.17	0.17	0.18	0.27	0.46	0.27

Table B.2: Summary statistics of the 10-fold cross-validation varying the max_depth complexity parameter from 2 to 20.

	count	mean	std	min	25%	50%	75%	max
max_depth								
2	10.00	0.32	0.15	0.00	0.26	0.30	0.40	0.55
3	10.00	0.24	0.14	0.08	0.17	0.21	0.27	0.55
4	10.00	0.26	0.13	0.08	0.17	0.25	0.32	0.55
5	10.00	0.23	0.10	0.08	0.17	0.22	0.27	0.46
6	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
7	10.00	0.23	0.10	0.08	0.17	0.22	0.27	0.46
8	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
9	10.00	0.23	0.10	0.08	0.17	0.22	0.27	0.46
10	10.00	0.24	0.13	0.08	0.17	0.22	0.27	0.55
11	10.00	0.23	0.10	0.08	0.17	0.22	0.27	0.46
12	10.00	0.23	0.10	0.08	0.17	0.22	0.27	0.46
13	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
14	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
15	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
16	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
17	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
18	10.00	0.24	0.13	0.08	0.17	0.22	0.27	0.55
19	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46
20	10.00	0.24	0.10	0.08	0.17	0.25	0.27	0.46

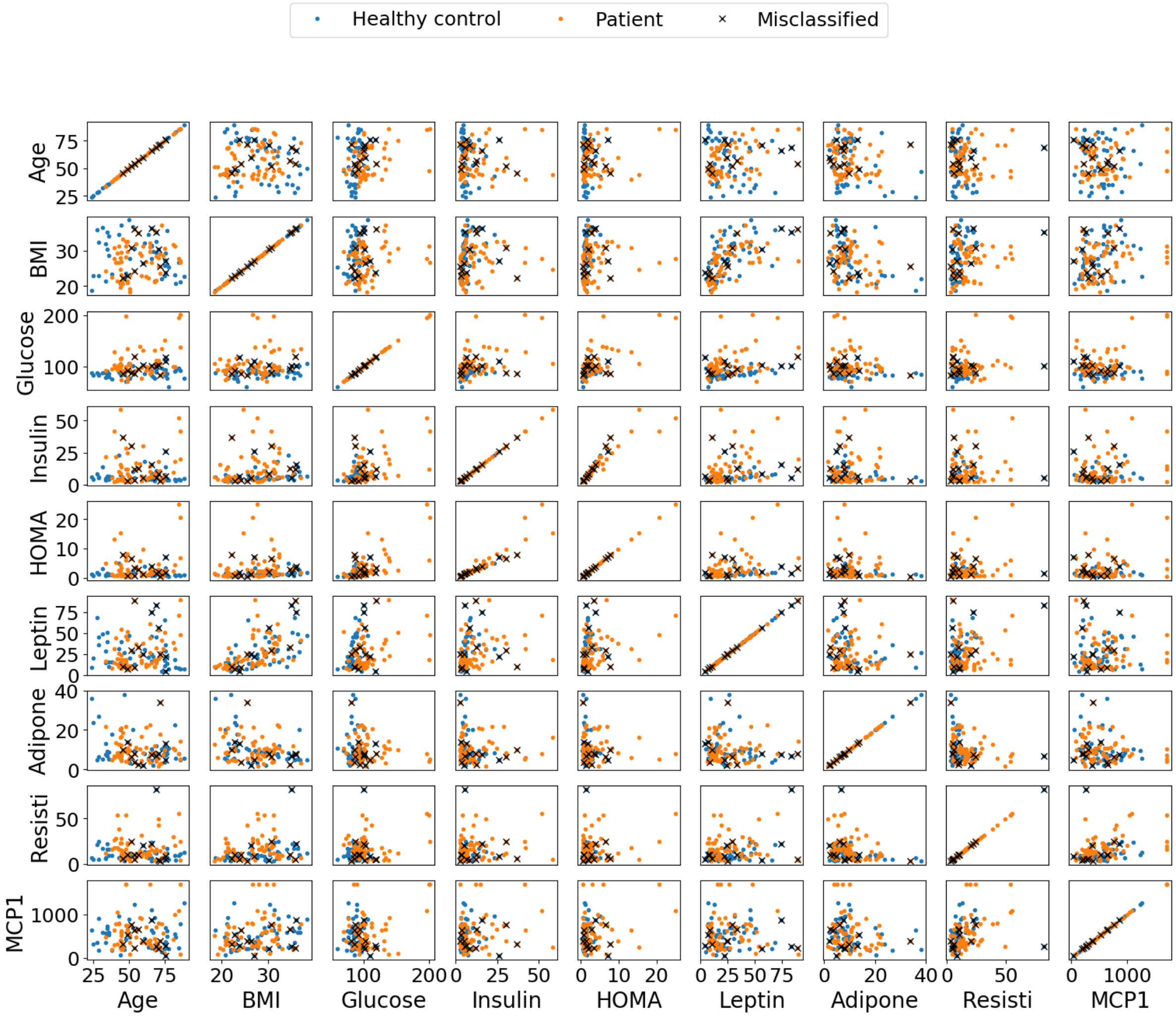


Figure B.3: Scatter plot matrix for all attributes. Each plot contains two series corresponding to the "Healthy control" class and the "Patient" class.

C Linear regression

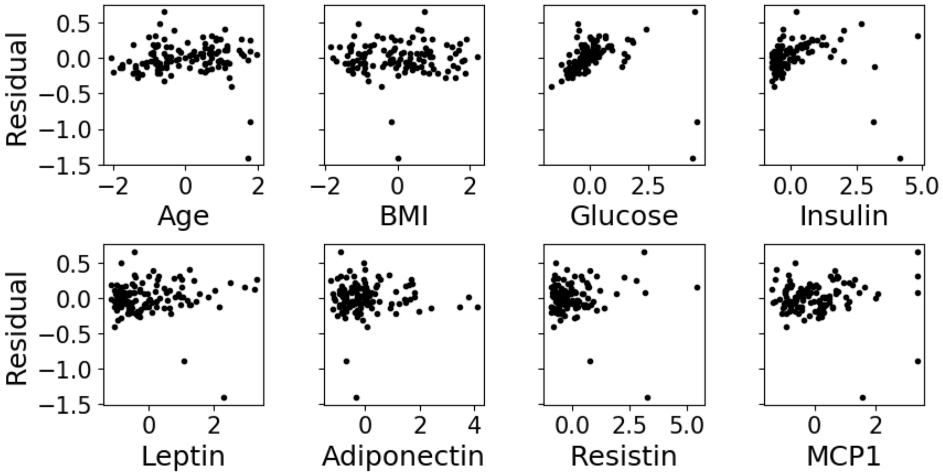


Figure C.4: Residuals from linear regression of standardized data, with no further transformations.

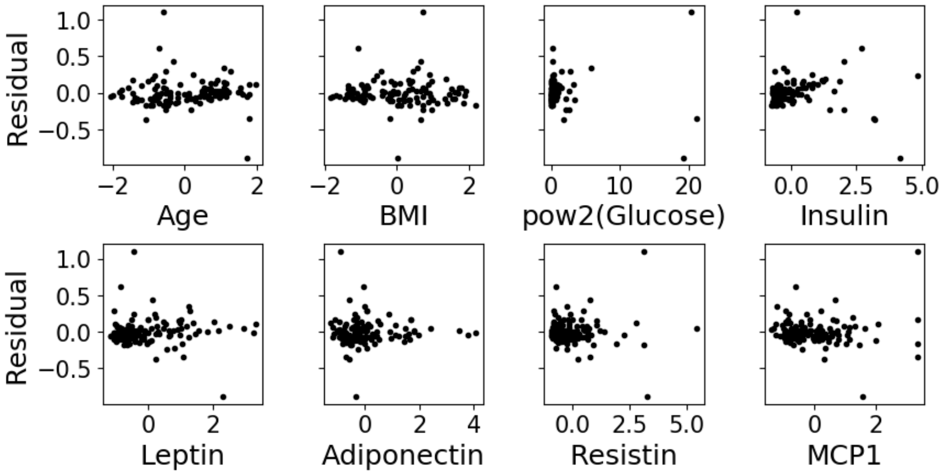


Figure C.5: Residuals from linear regression of standardized data and using the square of "Glucose" instead of simply "Glucose".

D Linear regression with forward selection

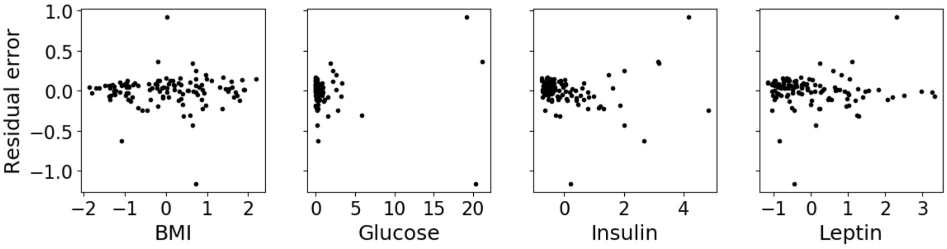


Figure D.6: Residuals from linear regression of standardized data. Features selected by forward selection.