PREDICTIVE MODELING
(O2023_MCD3396A)
ITESO

Department of Mathematics and Physics

Professor:
Ph. D. Riemman Ruiz Cruz

Application Project 1
(MAGIC Gamma Telescope)

Students:
Abraham Isidoro Muñoz
Diego Fernando Arriaza Alonzo
Gregorio Alberto Alvarez Alvarez

October 11th, 2023

Introduction

From professor's notes [1]:

THE PURPOSE OF THIS PROJECT IS TO EVALUATE THE KNOWLEDGE ACQUIRED ABOUT MODELS BASED ON SUPPORT VECTOR AND LINEAR MODELS TO SOLVE A PREDICTIVE PROBLEM.

As is already known, two main types of problems can be solved with models with supervised training (knowing the output variable to estimate).

- *A regression problem is considered when it is required to estimate the function that maps a data set X to a data set Y, where $Y \in \mathbb{R}$ is a cuantitative variable.*
- *A classification problem is considered when it is required to estimate the function that maps a data set X to a data set Y, where $X \in \mathbb{Z}^+$ is a categorical variable.*

Based on the written above, a dataset from https://archive.ics.uci.edu/datasets was downloaded, whose problems belong to classification tasks. Information and context about this dataset will be described below.

Title of Database: MAGIC gamma telescope data 2004.

Sources:

(a) Original owner of the database:

R. K. Bock

Major Atmospheric Gamma Imaging Cherenkov Telescope project (MAGIC)

http://wwwmagic.mppmu.mpg.de

rkb@mail.cern.ch

(b) Donor:

P. Savicky

Institute of Computer Science, AS of CR

Czech Republic

savicky@cs.cas.cz

(c) Date received: May 2007

Relevant Information:

The data are MC generated (generated via Monte Carlo simulations) to simulate registration of high energy gamma particles in a ground-based atmospheric

Cherenkov gamma telescope using the imaging technique. Cherenkov gamma telescope observes high energy gamma rays, taking advantage of the radiation emitted by charged particles produced inside the electromagnetic showers initiated by the gammas, and developing in the atmosphere. This Cherenkov radiation (of visible to UV wavelengths) leaks through the atmosphere and gets recorded in the detector, allowing reconstruction of the shower parameters. The available information consists of pulses left by the incoming Cherenkov photons on the photomultiplier tubes, arranged in a plane, the camera. Depending on the energy of the primary gamma, a total of few hundreds to some 10000 Cherenkov photons get collected, in patterns (called the shower image), allowing to discriminate statistically those caused by primary gammas (signal) from the images of hadronic showers initiated by cosmic rays in the upper atmosphere (background).

Typically, the image of a shower after some pre-processing is an elongated cluster. Its long axis is oriented towards the camera center if the shower axis is parallel to the telescope's optical axis, i.e. if the telescope axis is directed towards a point source. A principal component analysis is performed in the camera plane, which results in a correlation axis and defines an ellipse. If the depositions were distributed as a bivariate Gaussian, this would be an equidensity ellipse. The characteristic parameters of this ellipse (often called Hillas parameters) are among the image parameters that can be used for discrimination. The energy depositions are typically asymmetric along the major axis, and this asymmetry can also be used in discrimination. There are, in addition, further discriminating characteristics, like the extent of the cluster in the image plane, or the total sum of depositions.

The data set was generated by a Monte Carlo program, Corsika, described in D. Heck et al., CORSIKA, A Monte Carlo code to simulate extensive air showers, Forschungszentrum Karlsruhe FZKA 6019 (1998). The program was run with parameters allowing to observe events with energies down to below 50 GeV.

Objectives (Problem definition)

The main objective in this task is to perform and compare different linear classification techniques applied to the mentioned dataset. Techniques such as reduction of predictor variables based on correlations (heuristics), PCA, LDA and SVM with optimization of variables will be applied and compared their results. For the dataset in discussion, the tasks are focused in the classification of (simulated) photons caused by primary gammas ‘g’ or by hadronic showers ‘h’.

Development of Practice

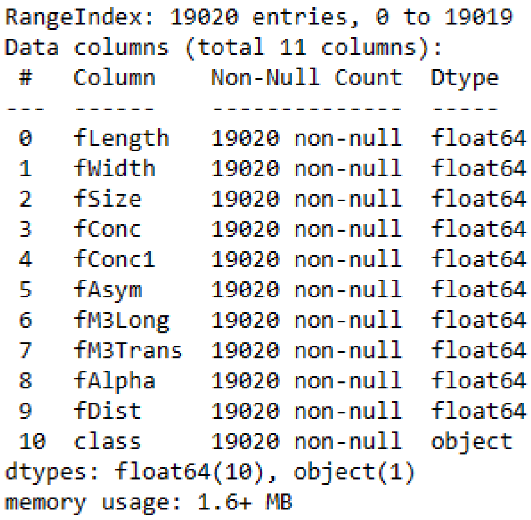
Description of the dataset and exploratory data analysis

The dataset comprehends 10 predictors and 1 output class variables. Names and description of the variables are shown in Table 1.

Variable	Type	Description
fLength	Continuous	Major axis of ellipse [mm]
fWidth	Continuous	Minor axis of ellipse [mm]
fSize	Continuous	10-log of sum of content of all pixels [in #phot]
fConc	Continuous	Ratio of sum of two highest pixels over fSize [ratio]
fConc1	Continuous	Ratio of highest pixel over fSize [ratio]
fAsym	Continuous	Distance from highest pixel to center, projected onto major axis [mm]
fM3Long	Continuous	3rd. root of third moment along major axis [mm]
fM3Trans	Continuous	3rd. root of third moment along minor axis [mm]
fAlpha	Continuous	Angle of major axis with vector to origin [deg]
fDist	Continuous	Distance from origin to center of ellipse [mm]
class	g,h (Categorical)	Gamma (signal); Hadron (background)

Table 1: Description of variables

After evaluating that there are no missing values in the dataset and the type of variables loaded in the Python DataFrame object (see Table 2) there were plotted both pairplots and histograms, as shown Figures 1 and 2, for which the output class ‘g’ was converted to 0 and the output class ‘h’ converted to 1 (binary classification).



```
RangeIndex: 19020 entries, 0 to 19019
Data columns (total 11 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   fLength   19020 non-null  float64
 1   fWidth    19020 non-null  float64
 2   fSize     19020 non-null  float64
 3   fConc     19020 non-null  float64
 4   fConc1    19020 non-null  float64
 5   fAsym     19020 non-null  float64
 6   fM3Long   19020 non-null  float64
 7   fM3Trans  19020 non-null  float64
 8   fAlpha    19020 non-null  float64
 9   fDist     19020 non-null  float64
 10  class     19020 non-null  object 
dtypes: float64(10), object(1)
memory usage: 1.6+ MB
```

Table 2: Description of variables

Regarding to the pairplots, there were not observed highly linear correlations except for few variables like *fConc*, *fConc1* and *fWidth,* as is also shown in Tables 3 and 4 with the correlation and covariance matrices. On the other hand, it was observed high skew from predictor variables such as *fLength*, *fWidht*, *fSize, fConc, fConc1, fAlpha*, while the output *class* variable showed lack of balance in the output, with the quantity of *0 'g'* class around 1.84 times the quantity of *1 'h'* class.

Based on the previous observations, and for better comparison between different models, it was decided to perform some experiments with the predictor variables standardized and non-standardized.

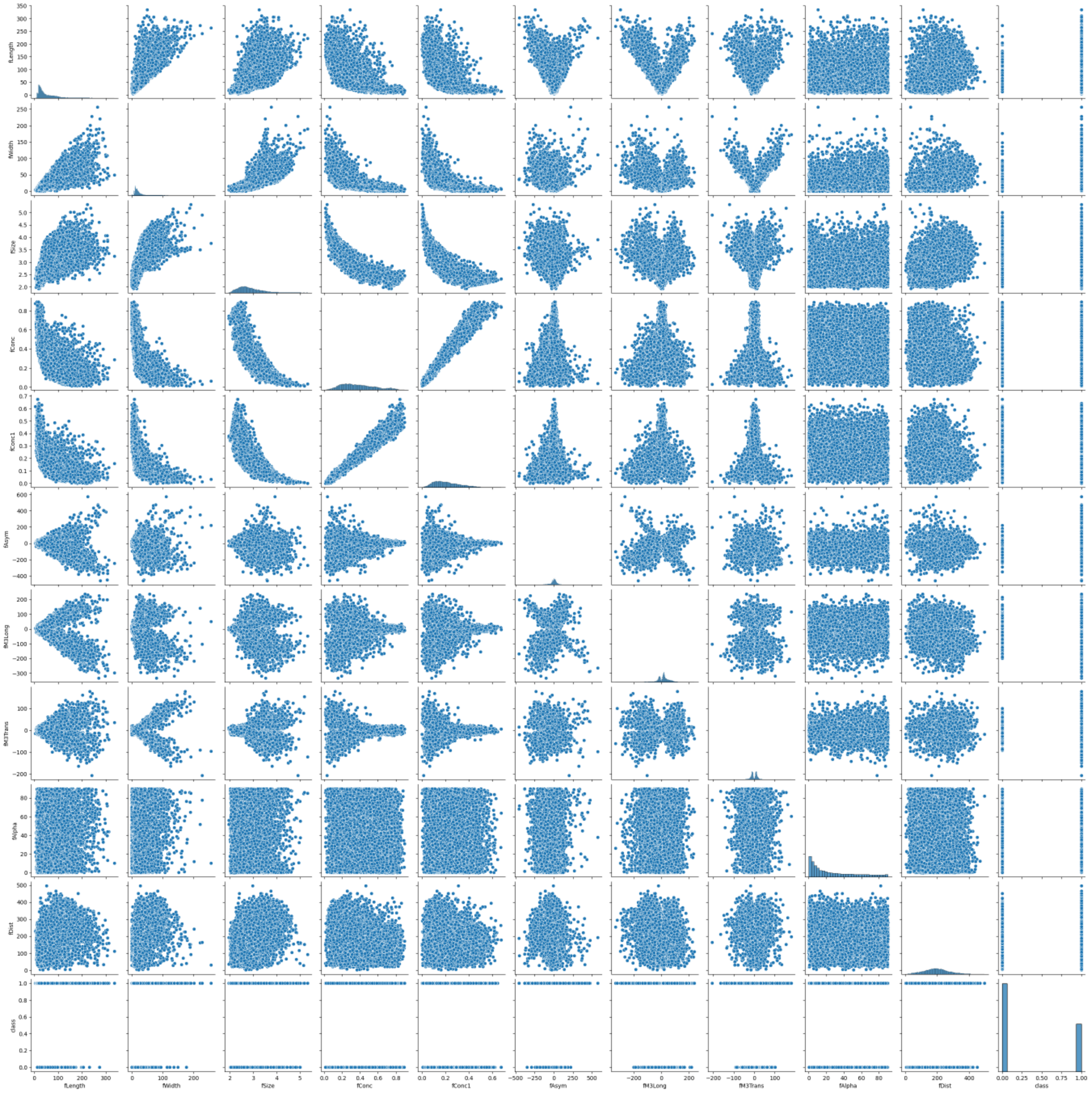


Figure 1: Pairplot of dataset

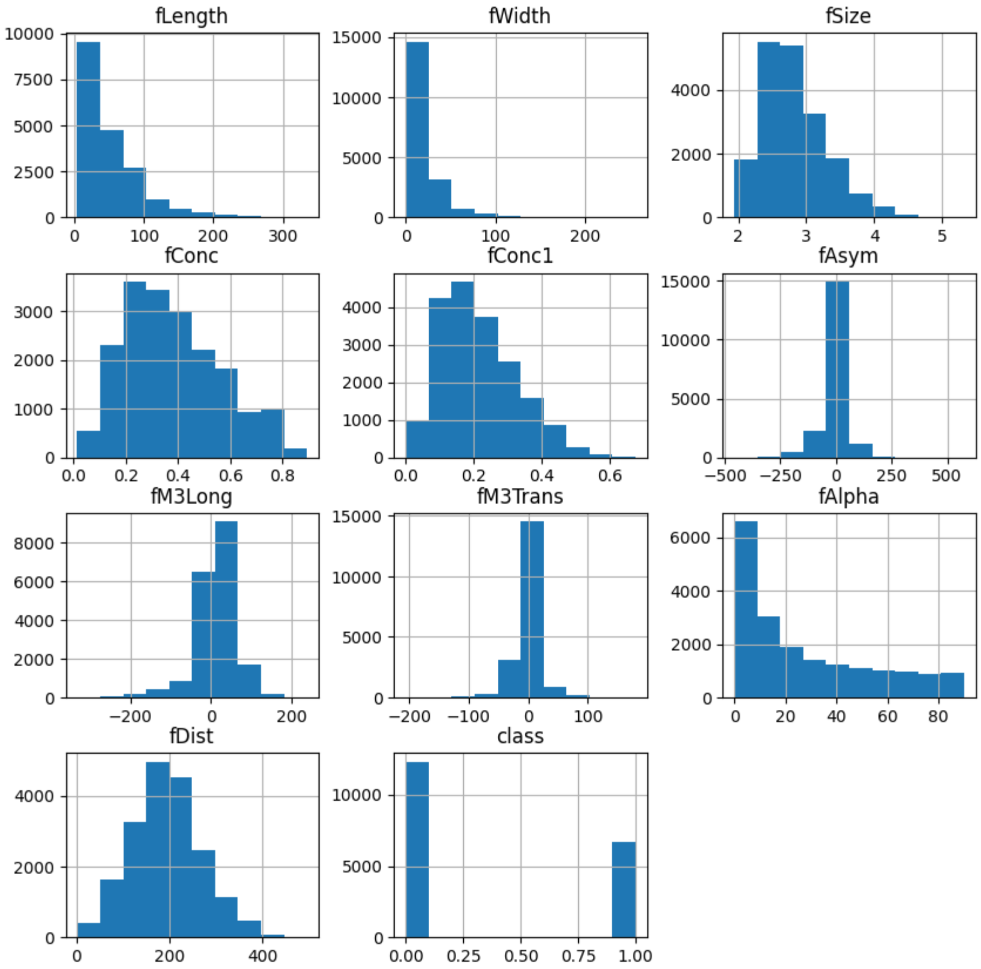
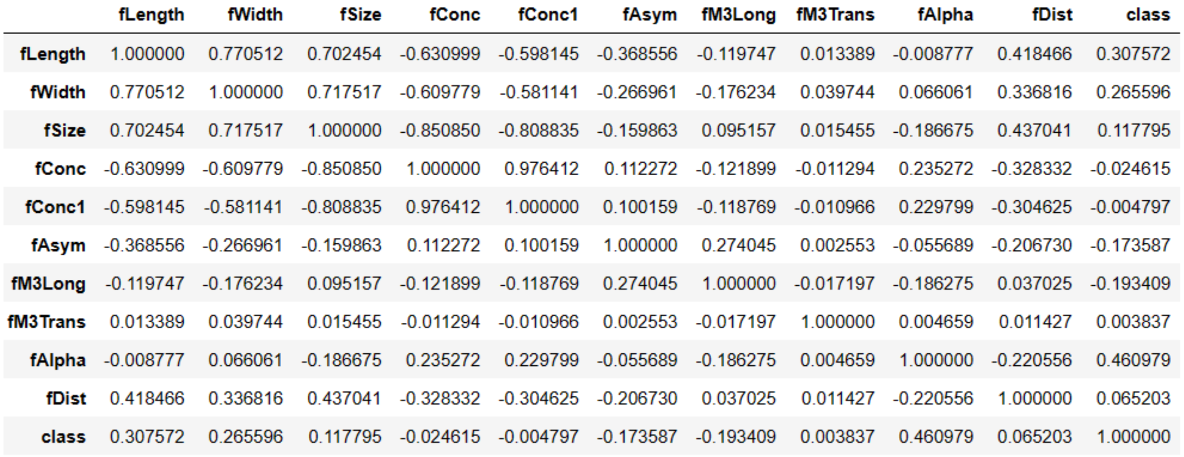
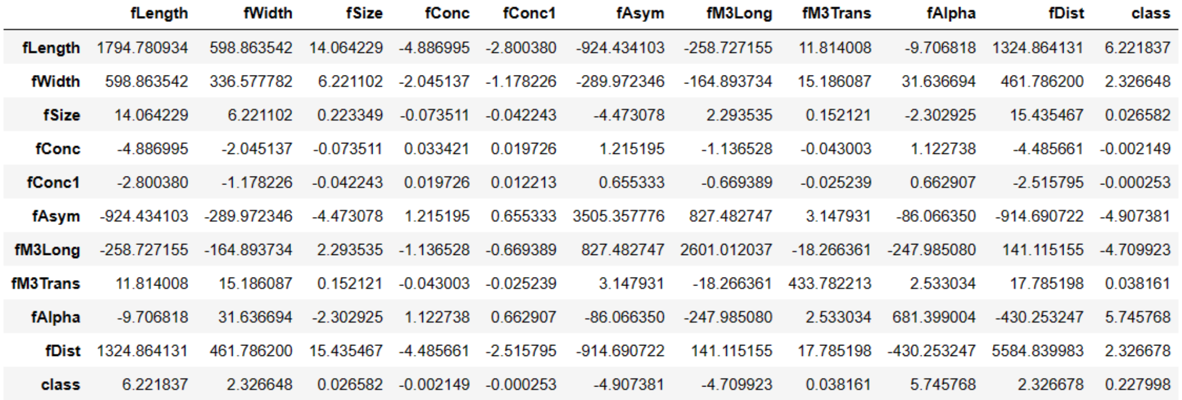


Figure 2: Histogram of dataset



	fLength	fWidth	fSize	fConc	fConc1	fAsym	fM3Long	fM3Trans	fAlpha	fDist	class
fLength	1.000000	0.770512	0.702454	-0.630999	-0.598145	-0.368556	-0.119747	0.013389	-0.008777	0.418466	0.307572
fWidth	0.770512	1.000000	0.717517	-0.609779	-0.581141	-0.266961	-0.176234	0.039744	0.066061	0.336816	0.265596
fSize	0.702454	0.717517	1.000000	-0.850850	-0.808835	-0.159863	0.095157	0.015455	-0.186675	0.437041	0.117795
fConc	-0.630999	-0.609779	-0.850850	1.000000	0.976412	0.112272	-0.121899	-0.011294	0.235272	-0.328332	-0.024615
fConc1	-0.598145	-0.581141	-0.808835	0.976412	1.000000	0.100159	-0.118769	-0.010966	0.229799	-0.304625	-0.004797
fAsym	-0.368556	-0.266961	-0.159863	0.112272	0.100159	1.000000	0.274045	0.002553	-0.055689	-0.206730	-0.173587
fM3Long	-0.119747	-0.176234	0.095157	-0.121899	-0.118769	0.274045	1.000000	-0.017197	-0.186275	0.037025	-0.193409
fM3Trans	0.013389	0.039744	0.015455	-0.011294	-0.010966	0.002553	-0.017197	1.000000	0.004659	0.011427	0.003837
fAlpha	-0.008777	0.066061	-0.186675	0.235272	0.229799	-0.055689	-0.186275	0.004659	1.000000	-0.220556	0.460979
fDist	0.418466	0.336816	0.437041	-0.328332	-0.304625	-0.206730	0.037025	0.011427	-0.220556	1.000000	0.065203
class	0.307572	0.265596	0.117795	-0.024615	-0.004797	-0.173587	-0.193409	0.003837	0.460979	0.065203	1.000000

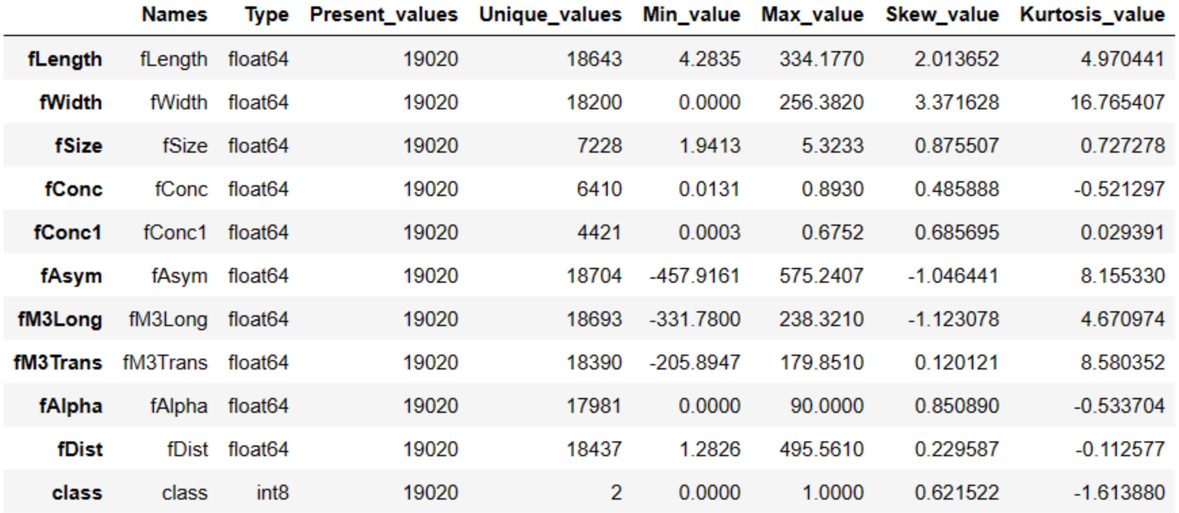
Table 3: Correlation table



	fLength	fWidth	fSize	fConc	fConc1	fAsym	fM3Long	fM3Trans	fAlpha	fDist	class
fLength	1794.780934	598.863542	14.064229	-4.886995	-2.800380	-924.434103	-258.727155	11.814008	-9.706818	1324.864131	6.221837
fWidth	598.863542	336.577782	6.221102	-2.045137	-1.178226	-289.972346	-164.893734	15.186087	31.636694	461.786200	2.326648
fSize	14.064229	6.221102	0.223349	-0.073511	-0.042243	-4.473078	2.293535	0.152121	-2.302925	15.435467	0.026582
fConc	-4.886995	-2.045137	-0.073511	0.033421	0.019726	1.215195	-1.136528	-0.043003	1.122738	-4.485661	-0.002149
fConc1	-2.800380	-1.178226	-0.042243	0.019726	0.012213	0.655333	-0.669389	-0.025239	0.662907	-2.515795	-0.000253
fAsym	-924.434103	-289.972346	-4.473078	1.215195	0.655333	3505.357776	827.482747	3.147931	-86.066350	-914.690722	-4.907381
fM3Long	-258.727155	-164.893734	2.293535	-1.136528	-0.669389	827.482747	2601.012037	-18.266361	-247.985080	141.115155	-4.709923
fM3Trans	11.814008	15.186087	0.152121	-0.043003	-0.025239	3.147931	-18.266361	433.782213	2.533034	17.785198	0.038161
fAlpha	-9.706818	31.636694	-2.302925	1.122738	0.662907	-86.066350	-247.985080	2.533034	681.399004	-430.253247	5.745768
fDist	1324.864131	461.786200	15.435467	-4.485661	-2.515795	-914.690722	141.115155	17.785198	-430.253247	5584.839983	2.326678
class	6.221837	2.326648	0.026582	-0.002149	-0.000253	-4.907381	-4.709923	0.038161	5.745768	2.326678	0.227998

Table 4: Covariance table

In addition to the previous analyses, Table 5 shows a summary report, where are included skew and kurtosis values, as well as the scale of the columns by attending to the minimum and maximum values of each column. From this is considered that methods for selection of predictors based on explained variance like PCA will show different results between standardized and non-standardized data.



	Names	Type	Present_values	Unique_values	Min_value	Max_value	Skew_value	Kurtosis_value
fLength	fLength	float64	19020	18643	4.2835	334.1770	2.013652	4.970441
fWidth	fWidth	float64	19020	18200	0.0000	256.3820	3.371628	16.765407
fSize	fSize	float64	19020	7228	1.9413	5.3233	0.875507	0.727278
fConc	fConc	float64	19020	6410	0.0131	0.8930	0.485888	-0.521297
fConc1	fConc1	float64	19020	4421	0.0003	0.6752	0.685695	0.029391
fAsym	fAsym	float64	19020	18704	-457.9161	575.2407	-1.046441	8.155330
fM3Long	fM3Long	float64	19020	18693	-331.7800	238.3210	-1.123078	4.670974
fM3Trans	fM3Trans	float64	19020	18390	-205.8947	179.8510	0.120121	8.580352
fAlpha	fAlpha	float64	19020	17981	0.0000	90.0000	0.850890	-0.533704
fDist	fDist	float64	19020	18437	1.2826	495.5610	0.229587	-0.112577
class	class	int8	19020	2	0.0000	1.0000	0.621522	-1.613880

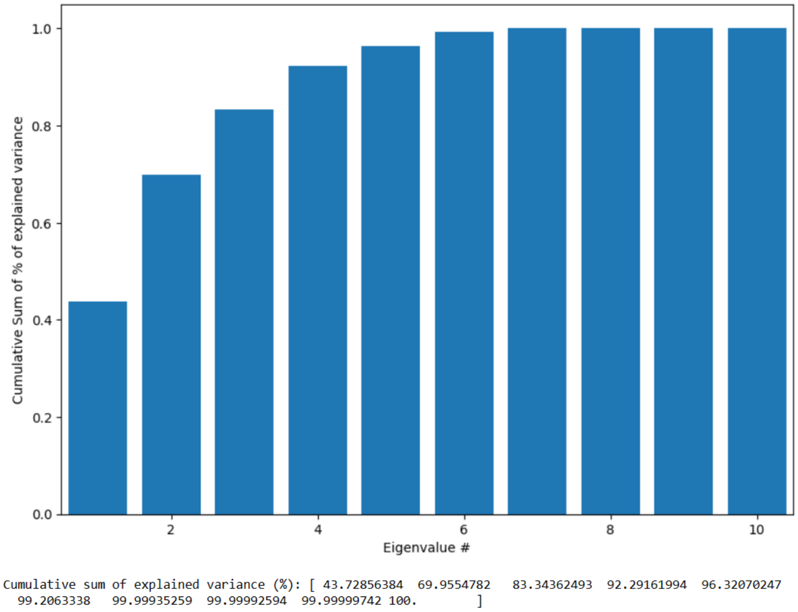
Table 5: Summary report

Classification models

Different models for classification and considerations on the data were studied, and results of most of them will be described below.

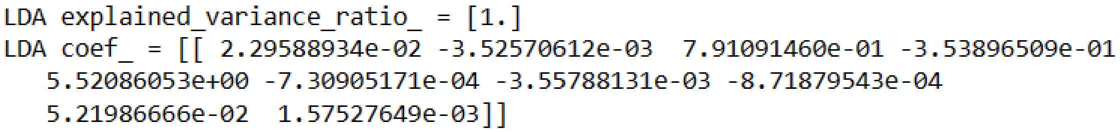
Classification with no reduction and no standardization of predictors. Logistic, logistic with PCA, and LDA.

After considering the cross-validation technique observed in class, where there were only 3 subsets -train, cross-validation and test-, different methods were considered, which are: 1) logistic regression with no reduction of predictors, 2) logistic regression with reduction of predictors via PCA and 3) LDA classification. It is important to notice that with PCA the number of predictors was reduced to 4, with an explanation of ~92% of variance, as shown in Figure 3, while for LDA the number of predictors was reduced to 1, but with all the weights for original predictors with values different to zero, as shows Figure 4.



```
Cumulative sum of explained variance (%): [ 43.72856384  69.9554782   83.34362493  92.29161994  96.32070247
  99.2063338   99.99935259  99.99992594  99.99999742 100.        ]
```

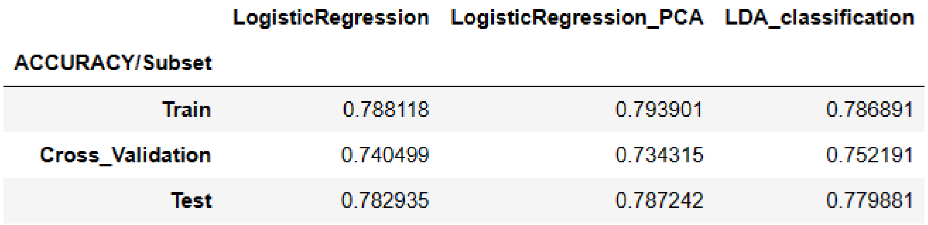
Figure 3: Cumulative sum of explained variance – PCA method



```
LDA explained_variance_ratio_ = [1.]
LDA coef_ = [[ 2.29588934e-02 -3.52570612e-03  7.91091460e-01 -3.53896509e-01
   5.52086053e+00 -7.30905171e-04 -3.55788131e-03 -8.71879543e-04
   5.21986666e-02  1.57527649e-03]]
```

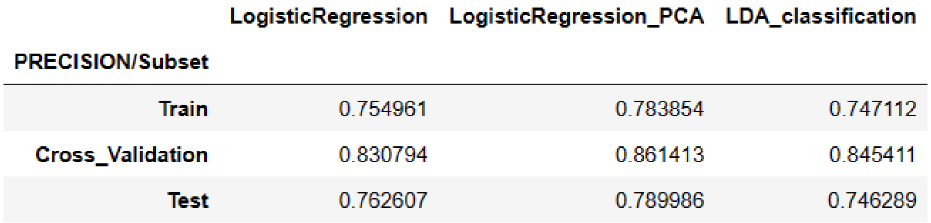
Figure 4: Explained variance and weights for original predictors – LDA method

A comparison of results such as accuracy, precision and recall metrics is shown in Tables 6 to 10. Similar results are obtained in the three subsets, with less than 3 percent of difference between different methods for the same metric and subset.



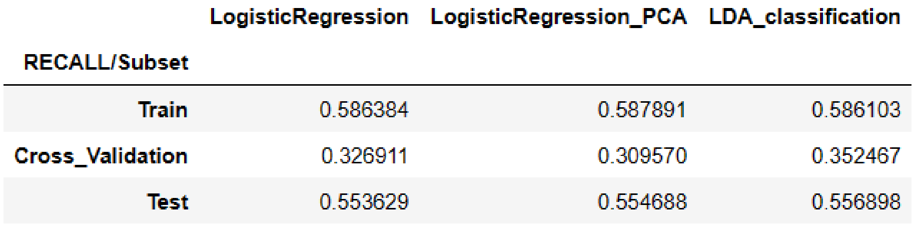
ACCURACY/Subset	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.788118	0.793901	0.786891
Cross_Validation	0.740499	0.734315	0.752191
Test	0.782935	0.787242	0.779881

Table 6: Accuracy metrics for the methods considered



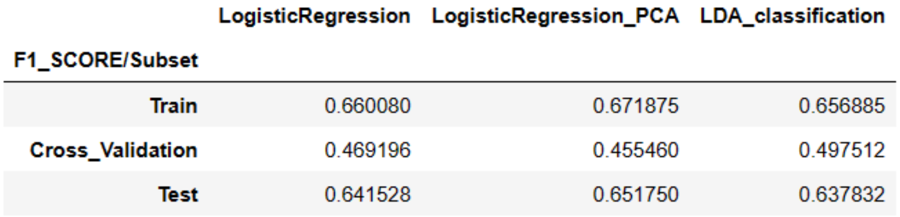
PRECISION/Subset	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.754961	0.783854	0.747112
Cross_Validation	0.830794	0.861413	0.845411
Test	0.762607	0.789986	0.746289

Table 7: Precision metrics for the methods considered



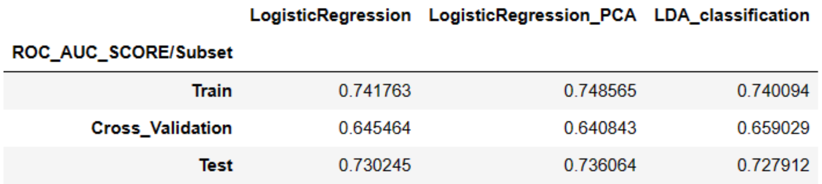
RECALL/Subset	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.586384	0.587891	0.586103
Cross_Validation	0.326911	0.309570	0.352467
Test	0.553629	0.554688	0.556898

Table 8: Recall metrics for the methods considered



F1_SCORE/Subset	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.660080	0.671875	0.656885
Cross_Validation	0.469196	0.455460	0.497512
Test	0.641528	0.651750	0.637832

Table 9: F1 Score metrics for the methods considered



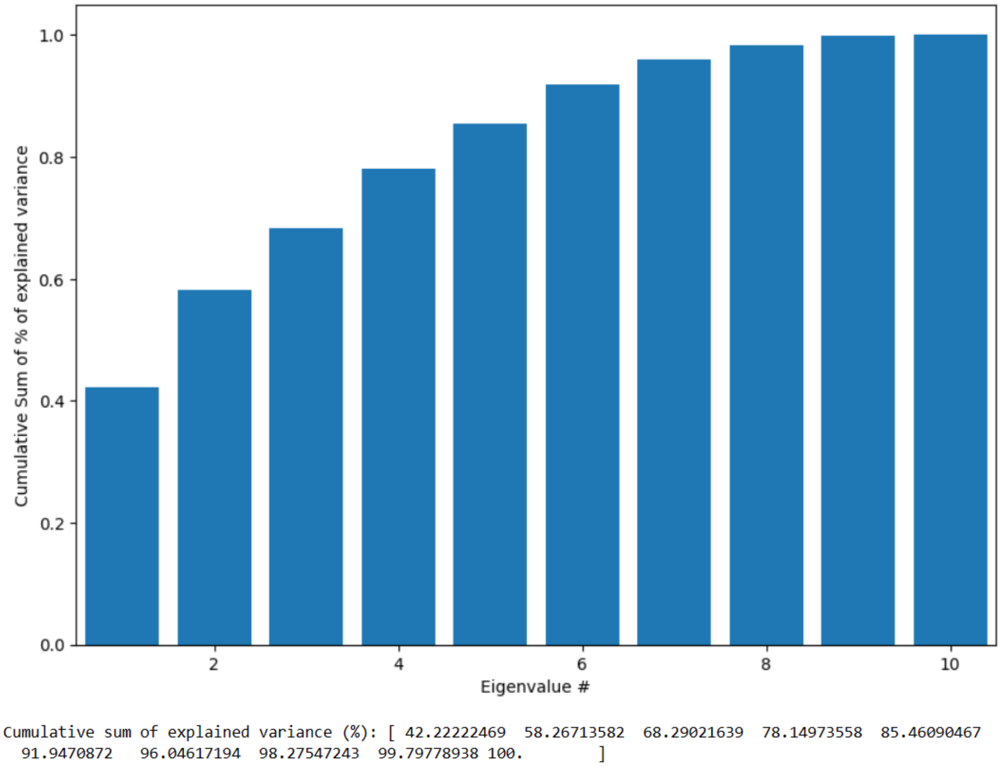
ROC_AUC_SCORE/Subset	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.741763	0.748565	0.740094
Cross_Validation	0.645464	0.640843	0.659029
Test	0.730245	0.736064	0.727912

Table 10: ROC_AUC Score metrics for the methods considered

Classification with no reduction but with standardization of predictors. Logistic, logistic with PCA, and LDA.

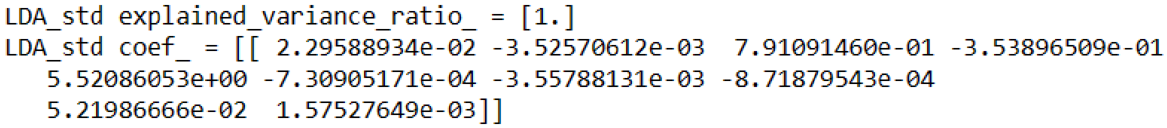
Just to compare possible differences with the previous values, and for better comparison with SVM techniques in next sections, a standardization of variables was implemented. There were expected the most significant differences in PCA, because the method relies on the covariance matrix instead of the correlation matrix, being susceptible to scales of each predictor, so a different number of predictors explaining same variance was highly possible. For LDA was expected only a change in weights.

In this case, with PCA the number of predictors was reduced from 10 to 6, with an explanation of ~92% of variance, as shown in Figure 5, while for LDA the number of predictors was reduced to 1, but with all the weights for original predictors with values different to zero, as shows Figure 6.



```
Cumulative sum of explained variance (%): [ 42.22222469  58.26713582  68.29021639  78.14973558  85.46090467
  91.9470872   96.04617194  98.27547243  99.79778938 100.        ]
```

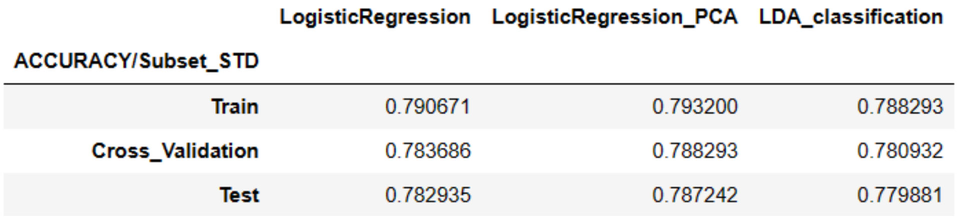
Figure 5: Cumulative sum of explained variance – PCA method



```
LDA_std explained_variance_ratio_ = [1.]
LDA_std coef_ = [[ 2.29588934e-02 -3.52570612e-03  7.91091460e-01 -3.53896509e-01
   5.52086053e+00 -7.30905171e-04 -3.55788131e-03 -8.71879543e-04
   5.21986666e-02  1.57527649e-03]]
```

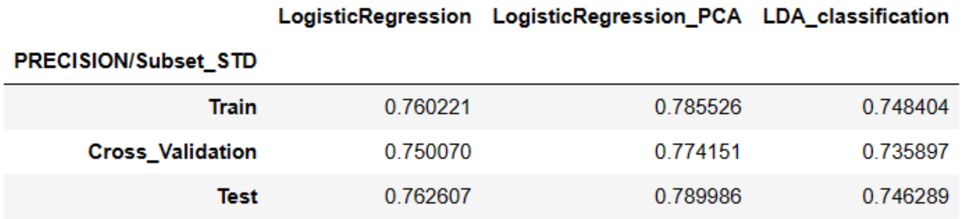
Figure 6: Explained variance and weights for original predictors – LDA method

A comparison of results is shown in Tables 11 to 15. Similar results are also obtained in the three subsets, with less than 3 percent of difference between different methods for the same metric and subset. However, are observed more homogeneous values of cross validation results in precision and recall metrics compared to the other subsets, even when the seed was the same to perform the data splitting.



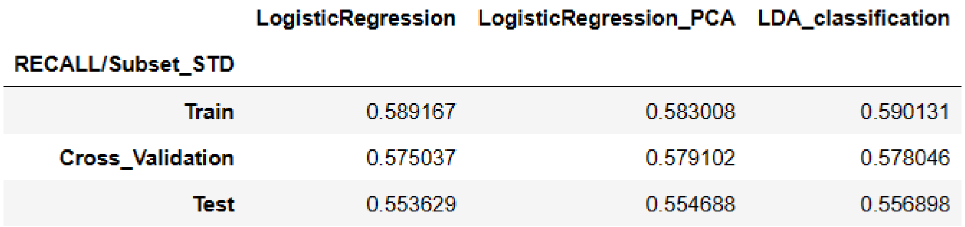
ACCURACY/Subset_STD	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.790671	0.793200	0.788293
Cross_Validation	0.783686	0.788293	0.780932
Test	0.782935	0.787242	0.779881

Table 11: Accuracy metrics for the methods considered with standardized predictors



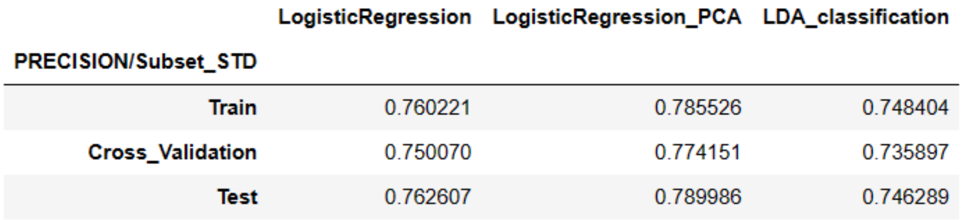
PRECISION/Subset_STD	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.760221	0.785526	0.748404
Cross_Validation	0.750070	0.774151	0.735897
Test	0.762607	0.789986	0.746289

Table 12: Precision metrics for the methods considered with standardized predictors



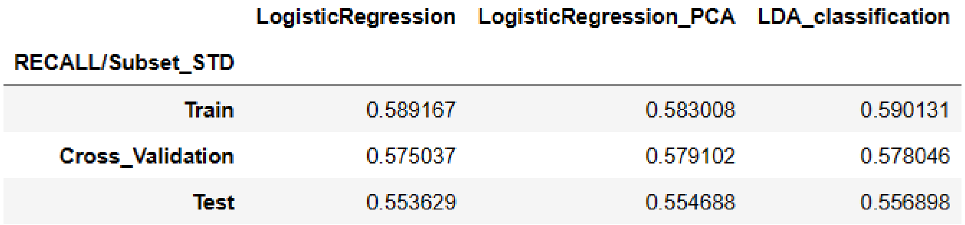
RECALL/Subset_STD	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.589167	0.583008	0.590131
Cross_Validation	0.575037	0.579102	0.578046
Test	0.553629	0.554688	0.556898

Table 13: Recall metrics for the methods considered with standardized predictors



PRECISION/Subset_STD	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.760221	0.785526	0.748404
Cross_Validation	0.750070	0.774151	0.735897
Test	0.762607	0.789986	0.746289

Table 14: F1 Score metrics for the methods considered



RECALL/Subset_STD	LogisticRegression	LogisticRegression_PCA	LDA_classification
Train	0.589167	0.583008	0.590131
Cross_Validation	0.575037	0.579102	0.578046
Test	0.553629	0.554688	0.556898

Table 15: ROC_AUC Score metrics for the methods considered

Classification with Logistic Regression with remotion of skew, reduction of predictors based on correlations and standardized.

Interested in the effects of removing skewness of some predictors, the data was firstly processed to remove 'redundant' columns with high correlation. From 10 initial predictors the process reduced the number to 7 predictors, as shown in Figure 7.

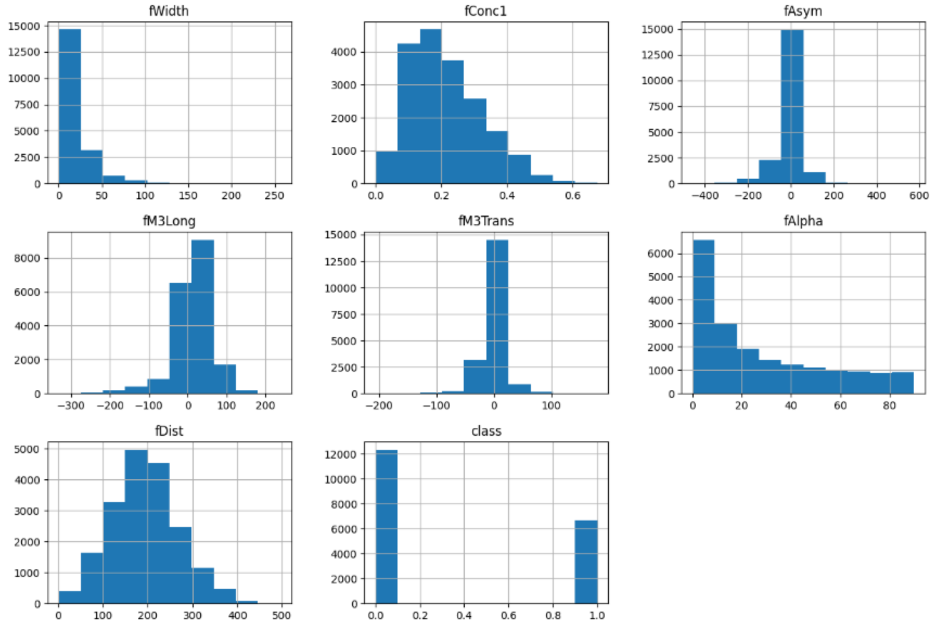


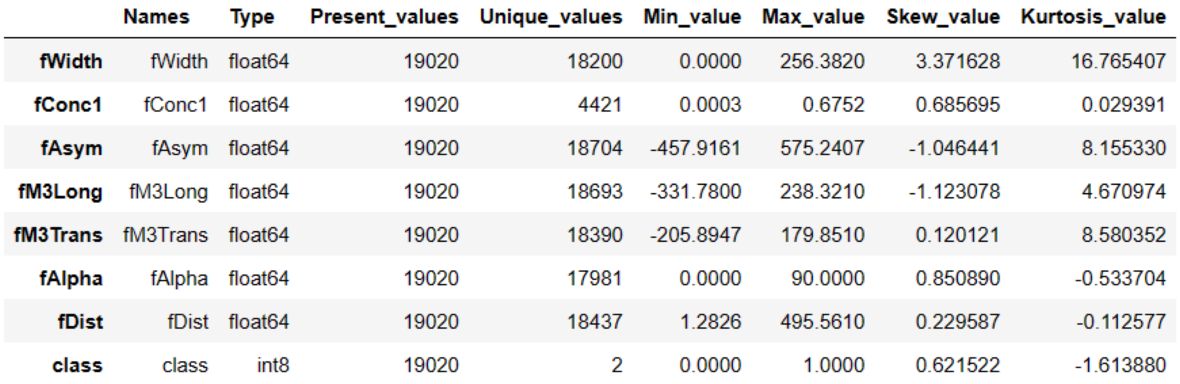
Figure 7: Histogram of predictors conserved after the remotion based on correlations

For the algorithm and criteria defined to remove variables that were redundant (highly correlated), the following steps were applied:

1. Define a threshold criteria to remove variables. 0.7 was chosen arbitrarily.
2. Calculate of a correlation matrix
3. Find the pair of variables with highest correlation
4. Check if correlation of the pair of variables was above the correlation threshold.
5. Calculate the average correlation of each variable of the pair against all other variables.
6. The variable with the highest average correlation is removed from the dataset.
7. Repeat until the highest correlation of a pair variables is below the correlation threshold.

As observed in Figure7, by applying the previous criteria for removing variables, the new dataset was left with the following variables: *fWidth*, *fConc1*, *fAsym*, *fM3Long*, *fM3Trans*, *fAlpha*, *fDist*. The removed variables were the following: *fLength*, *fSize and fConc*.

As it is shown in Table 16, there are some variables that have high skewness. So, a Yeo-Johnson Transformation was used to transform the variables of *fWidth*, *fConc1 and fM3Long*. Although *fAlpha* and *fAsym* have high skewness, they were not transformed because it shifted their distribution very drastically. Figure 8 shows the resultant distributions.



	Names	Type	Present_values	Unique_values	Min_value	Max_value	Skew_value	Kurtosis_value
fWidth	fWidth	float64	19020	18200	0.0000	256.3820	3.371628	16.765407
fConc1	fConc1	float64	19020	4421	0.0003	0.6752	0.685695	0.029391
fAsym	fAsym	float64	19020	18704	-457.9161	575.2407	-1.046441	8.155330
fM3Long	fM3Long	float64	19020	18693	-331.7800	238.3210	-1.123078	4.670974
fM3Trans	fM3Trans	float64	19020	18390	-205.8947	179.8510	0.120121	8.580352
fAlpha	fAlpha	float64	19020	17981	0.0000	90.0000	0.850890	-0.533704
fDist	fDist	float64	19020	18437	1.2826	495.5610	0.229587	-0.112577
class	class	int8	19020	2	0.0000	1.0000	0.621522	-1.613880

Table 16: Summary report for the reduced set of variables

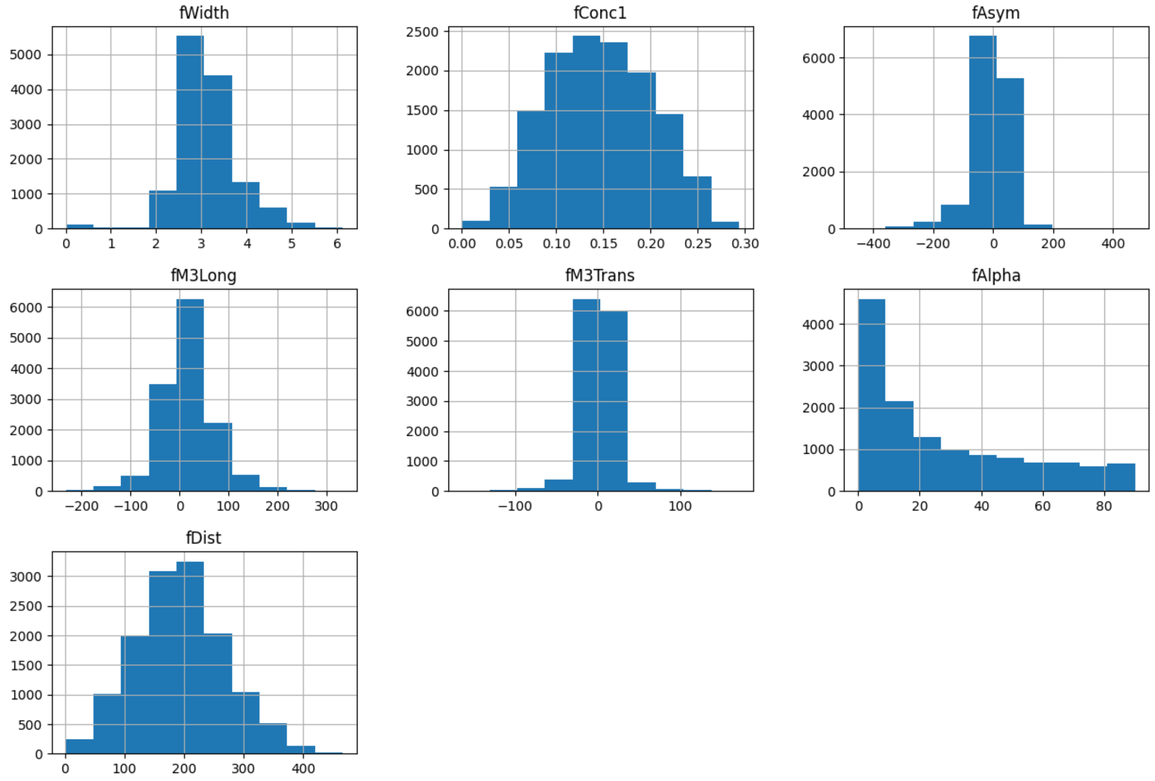


Figure 8: Reduced set of predictors after remotion of skew

After these processes, and in addition with a standardization of the transformed variables, which gave the histograms observed in Figure 9, a logistic regression was performed. A comparison of results for the three subsets of training, cross validation and test, is shown in Table 17.

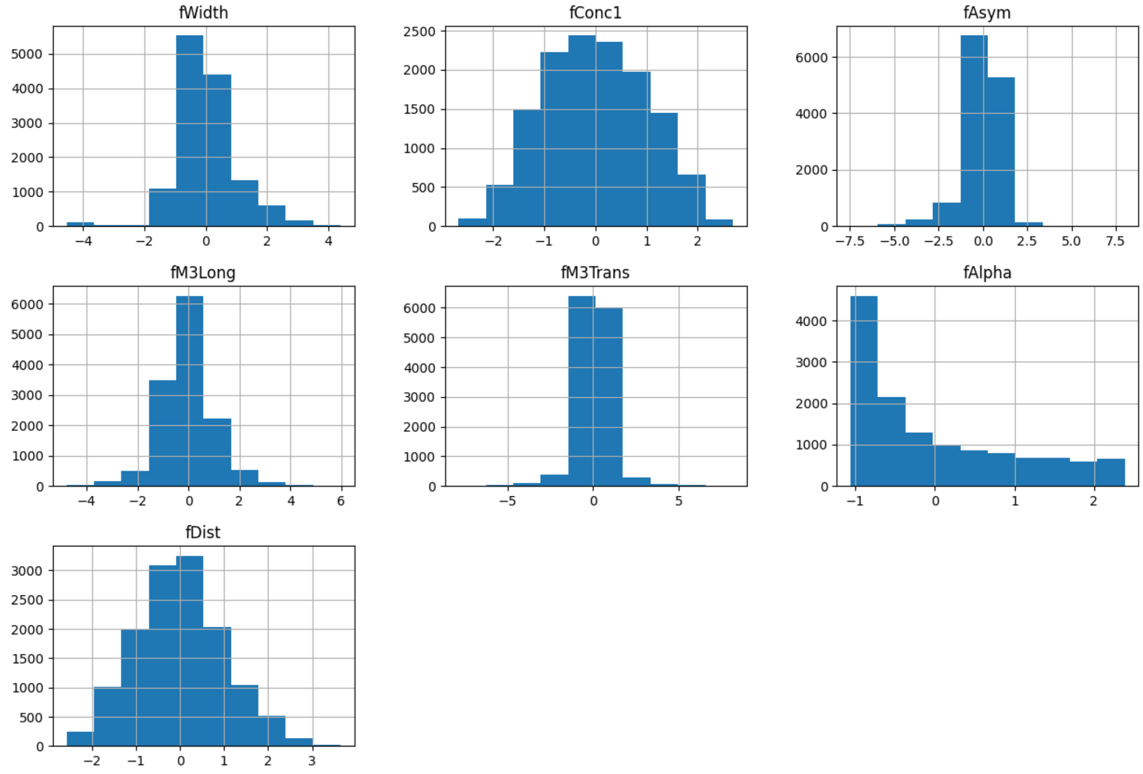
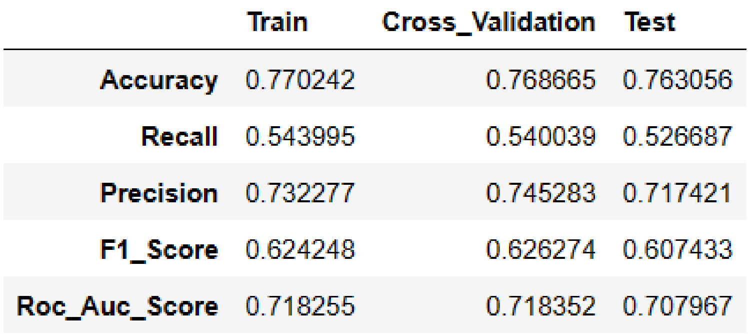


Figure 9: Reduced set of predictors after remotion of skew and standardization.



	Train	Cross_Validation	Test
Accuracy	0.770242	0.768665	0.763056
Recall	0.543995	0.540039	0.526687
Precision	0.732277	0.745283	0.717421
F1_Score	0.624248	0.626274	0.607433
Roc_Auc_Score	0.718255	0.718352	0.707967

Table 17: Metrics of reduced set of predictors with reduction of skew and standardized

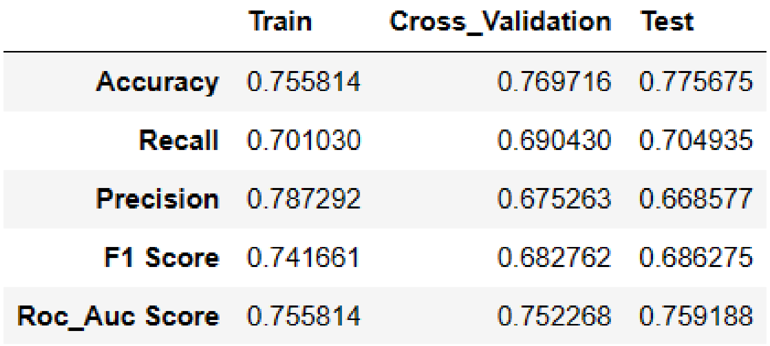
Classification with reduction of predictors based on correlations, standardized, reduction of skewness and applying SMOTE resampling.

Given that, in Figure 7 and Table 16 attending to distributions and skew, it is observed that the output *class* variable is slightly imbalanced, specifically 65%-35%. Therefore, an oversampling criteria called SMOTE was executed.

SMOTE stands for Synthetic Minority Over-sampling Technique. The way it works is the following:

1. Starts by randomly selecting a minority class instance from the dataset. This will serve as the basis for generating synthetic examples.
2. For the selected instance, SMOTE identifies its k-nearest neighbors within the minority class. The value of 'k' is a user-defined parameter that influences the amount of oversampling.
3. SMOTE generates synthetic instances by interpolating between the selected instance and its k-nearest neighbors. This is typically done by selecting a random number between 0 and 1 for each feature, and using this value to determine the feature values of the synthetic instances. The synthetic instances are created by blending the attributes of the selected instance and its neighbors.
4. Steps 1-3 are repeated until the desired level of oversampling is achieved. The user can specify the amount of oversampling by setting a parameter, often denoted as the "sampling ratio.

Results of this approach are shown in Table 18, where slight differences were observed against the previous approach.



	Train	Cross_Validation	Test
Accuracy	0.755814	0.769716	0.775675
Recall	0.701030	0.690430	0.704935
Precision	0.787292	0.675263	0.668577
F1 Score	0.741661	0.682762	0.686275
Roc_Auc Score	0.755814	0.752268	0.759188

Table 18: Metrics of reduced set of predictors with reduction of skew, standardized and with SMOTE resampling

Classifications with Support Vector Machines.

Classification with no predictor's reduction

For this case, the entire set of independent variables was divided into two subsets: the training set, which comprised 70% of the data, and the test set, which comprised the remaining 30%. To facilitate the computation process, a standard scaling technique was applied.

During the fitting process, an RBF (Radial Basis Function) kernel was computed, and the optimal hyperparameters were determined using Bayesian Optimization. The cost function used for this optimization process was based on the accuracy of the test set.

Following the Bayesian Optimization process, the following functions were obtained in the hyperparameters space:

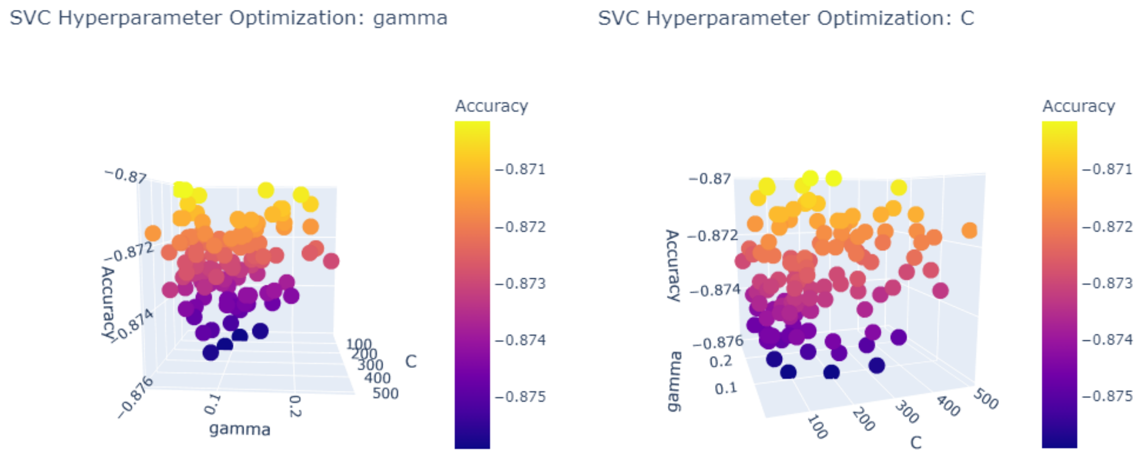


Figure 10. Accuracy vs hyperparameters plot. The accuracy values have been adjusted for presentation purposes. Left: This plot provides a perspective from the gamma side. It can be observed that the optimal values for gamma fall within the range of 0.01 to 0.25, with a minimum value around 0.09. Right: This plot provides a perspective from the C side. The function is also convex in this case, and the minimum values for C range between 90 and 300.

Classification with predictor's reduction

For this case, the independent variables were selected using a pair-correlation process, where variables with a correlation greater than 0.7 were chosen. This process resulted in a reduction of 3 variables.

After the train-test-split, a power transformation was applied to three variables, namely *fWidth*, *fConc1*, and *fM3Long*. These variables were selected based on their visually observed high skewness.

Furthermore, all variables were standardized to ensure that they were on the same scale.

Similar to the previous model, an RBF kernel was used to determine the optimal hyperparameters. In this example, a 5-fold cross-validation was performed on the training set. The cost function, which was calculated as 2 times the test accuracy plus the mean cross-validation accuracy, was used to find the maximizers.

In contrast to the previous cost plot, the cost function exhibited a damped sine wave shape in the *gamma* direction. In the *C* direction, there was a tendency for small values with little to no variation in accuracy observed throughout both directions.

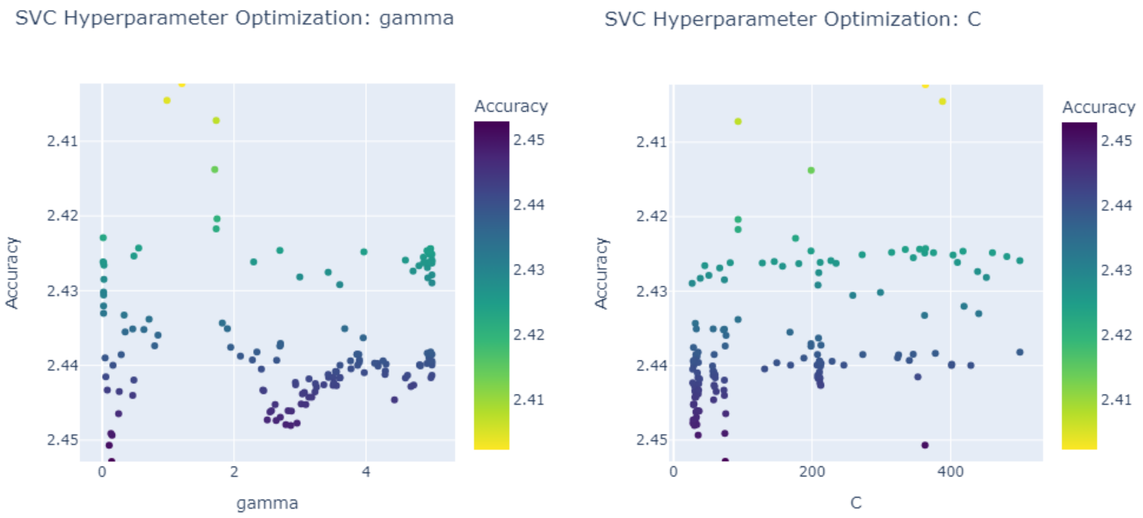
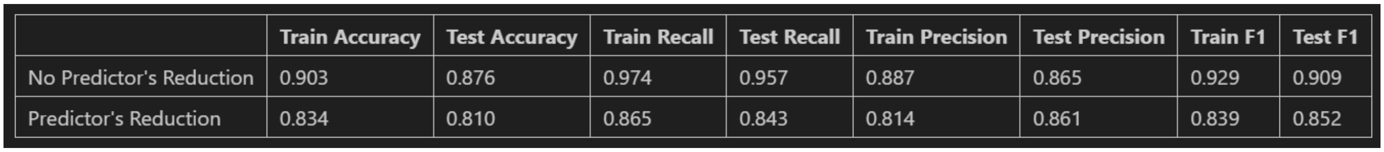


Figure 11. On the left side, the plot shows the relationship between accuracy and gamma. On the right side, the plot displays the relationship between accuracy and C. It is evident from both plots that there is minimal variation in accuracy across the range of both variables.

Results

Results of the two methods using RBF kernels in SVM are shown in Table 19. Significant differences are shown between these two, in favor of better metrics for the method where no remotion of predictors was performed. In general, SVM shows better results than the other approaches were, for example, the accuracy metrics were no bigger than 0.8.



	Train Accuracy	Test Accuracy	Train Recall	Test Recall	Train Precision	Test Precision	Train F1	Test F1
No Predictor's Reduction	0.903	0.876	0.974	0.957	0.887	0.865	0.929	0.909
Predictor's Reduction	0.834	0.810	0.865	0.843	0.814	0.861	0.839	0.852

Table 19: Results of the two methods using SVM

Discussion of results and Conclusions

For Logistic Regression with and without reduction of predictors and with and without standardization, the results were below 0.8 in all accuracy metrics and even for the recall metric the results were below 0.7. With variables transformation, reduction of 3 predictors and applying SMOTE technique for oversampling it was expected to see a significant improvement in metrics results. However, this was only achieved for the recall metric, where the results for Train and Test were 0.7 and for the precision metric and accuracy metric there was no improvement. Even the precision metric for the Test dataset was significantly worse.

It was shown that the best metrics were obtained with the SVC model after the hyperparameters optimization. This is expected due to the non-linear boundaries that are found with the use of kernelized data.

Based on our findings, it was demonstrated that the SVC model, after undergoing hyperparameter optimization, yielded the best metrics. This outcome aligns with expectations, as the utilization of kernelized data enables the identification of non-linear boundaries.

The superiority of the SVC model can be attributed to its ability to handle non-linear boundaries through the use of kernelized data. This feature enables the model to uncover complex patterns that may not be discernible with linear models like Logistic Regression.

It is worth noting that feature transformation, such as scaling or applying other transformations to the data, did not significantly improve the performance of the models. This suggests that the original features, after standardization, already contained sufficient information for accurate predictions.

Bibliography & references

[1] *Class Notes*. Riemman Ruiz Cruz.

[2] Kuhn, M., & Johnson, K. (2018). *Applied Predictive Modeling*. Springer.

[3] Cristianini, N., Shawe-Taylor, J., & Shawe-Taylor, D. O. C. S. R. H. J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press.

[4] Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for machine learning*. Cambridge University Press.

[5] Rogers, S., & Girolami, M. (2016*). A first course in machine learning*. Chapman & Hall/CRC.