Analysis of the dataset *Credit Card Default*

Gaspare Ferraro (520549)
Javad Khalili (546677)
Mario Matovic (583449)

January 9, 2019

University of Pisa
Exam: Data Mining
Year: 2018/2019
Instructors: Dino Pedreschi, Anna Monreale, Riccardo Guidotti

Contents

1. Introduction

This report is aimed to illustrate the phases and the results of the analysis that we have conduced regarding the customers default payments in Taiwan. In particular our target is to better understand under which conditions we should consider a client credibile or not.

Each customer is modelled by a record in the dataset, which is composed by 24 attributes that describes its personal information and its banking data (like the credit limit, payment amount and others). The analysis is composed in 4 phases:

- Semantical analysis and data manipulations of each customer (data cleaning, variables transformation, redundant variables, ...)
- Use of 3 clustering algorithms (K-Means, DBSCAN, Hierarchical clustering) to group customers according to similarity properties in order to formulate hypotheses about customers credibility.
- Verification of the hypotheses given in the previous phase via the determination of frequent itemsets and association rules, in order to find co-occurrences between attributes.
- Classification of the customers between who fail to make a payment by time and regular customers.

2. Data Understanding

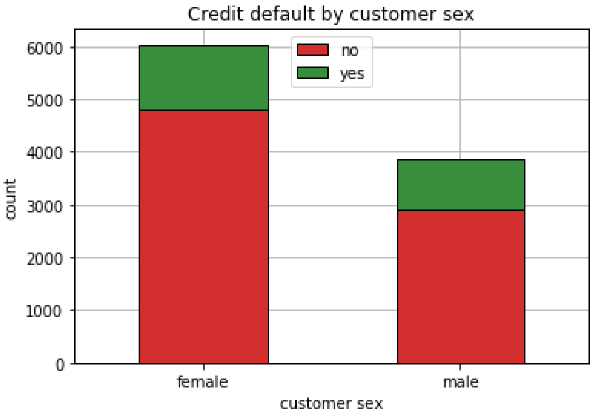
The dataset is composed by 10000 records. Each record represents a customer, described by 24 different attributes.

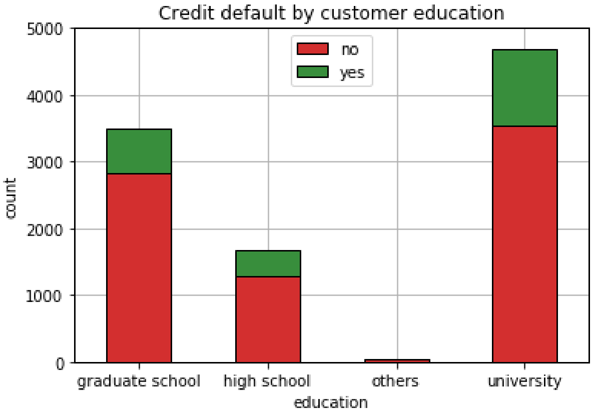
2.1 Data semantics, distribution and statistics

In this section we will analyze, for each attribute, its semantic and we will show interesting statistic and plot. We have used two different colors for who went in credit default (green) and not (red) in order to better visualize their distribution among the different attributes.
We have discretized the continuous attributes by using the natural binning method.
For these attributes the mode of the bin has also been reported as it is more representative.

Sex

Gender of the customer.
A binary attribute that can assume the values of *male* (3868 of 10000) or *female* (6032 of 10000). Both of the gender values have a similar credit default rate (25% for males and 20% for females).



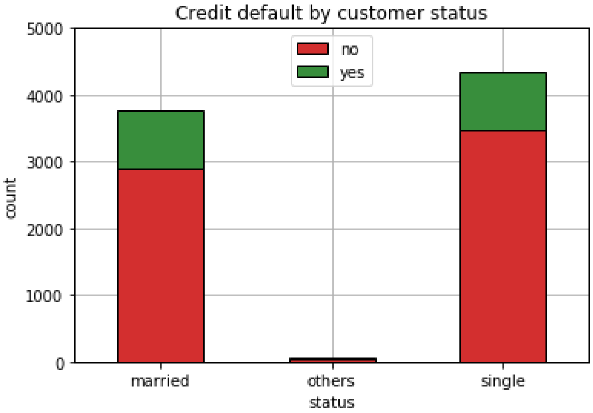


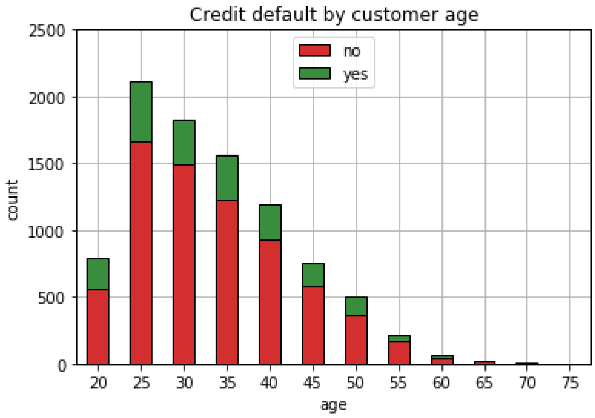
Education

Qualification of the customer.
A categorical attribute that can assume the values of *university* (4685 of 10000), *high school* (1672 of 10000), *graduate school* (3480 of 10000) or *others* (36 of 10000). The default rate is again very similar for all the qualifications (around the 20%), except for the *others* which is equal to 5%, but its number of records is very low to make any assumptions.

Status

Marital status of the customer.
A categorical attribute that can assume the values of *married* (4685 of 10000), *single* (3757 of 10000) or *other status* (75 of 10000).
The default rate is very similar for all the status (around the 25%).





Age

Age of the customer.

An attribute that in the dataset assume integer values in $[21, 75]$, the lower limit to 21 is due that in Taiwan the age of majority is 20, on the other hand the upper limit can be as high as humanly possible. The average age is 35.49, the standard deviation is 9.22. The mode is 29 and the median is 34. The 50% of the ages lie in $[28, 41]$.

We decide to set a bin to 5 year as it represent a good trade-off between the size and number of bins. The bin with most elements is 25.

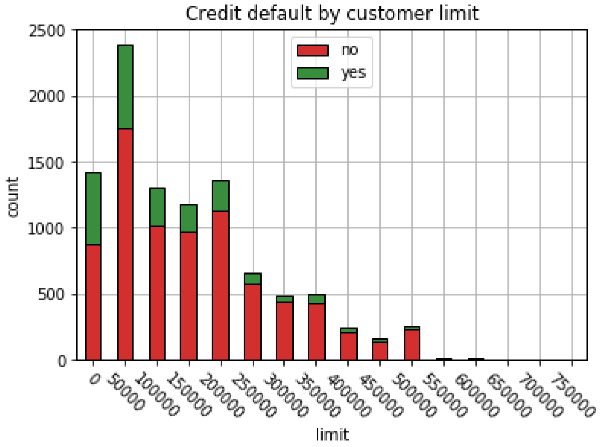
Again the default rate is similar for all the bins (around 25%).

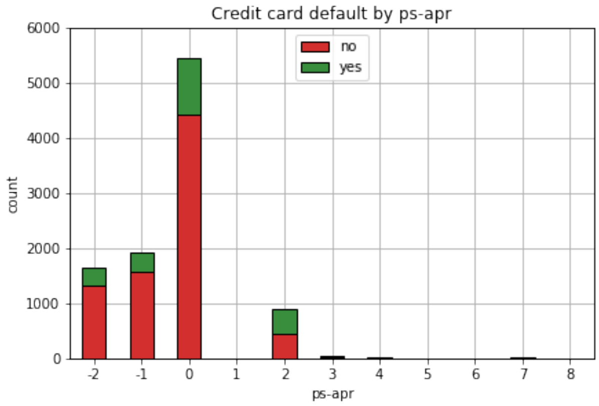
Limit

Limit of the credit card (expressed in NT dollar). It is the maximum amount the credit card company will let borrow on the account, a continuous attribute that can assume values in $[10000, 780000]$ (all values are multiples of 10000).

The average is 167197 and the standard deviation is 128975, 50% of the ages lie in $[50000, 240000]$. The bin with most elements is 50000.

The default rate in this case is very different for each bin as it decrease with the limit: the first bin has a default rate of 38%, the second one of 26% and around 10% for the last bin.





Payment status

History of past payments.

Six categorical attributes that represent the repayment status, one for each month between April and September. A payment status is an integer number in the range $[-2, 9]$ where:

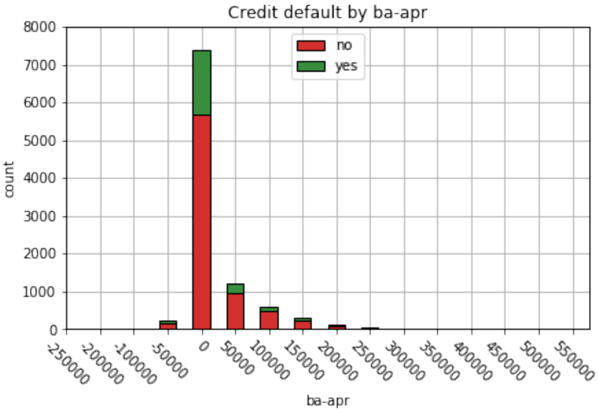
-2 = no consumption
-1 = paid in full
0 = the use of revolving credit
1 = payment delay for one month
2 = payment delay for two months
...

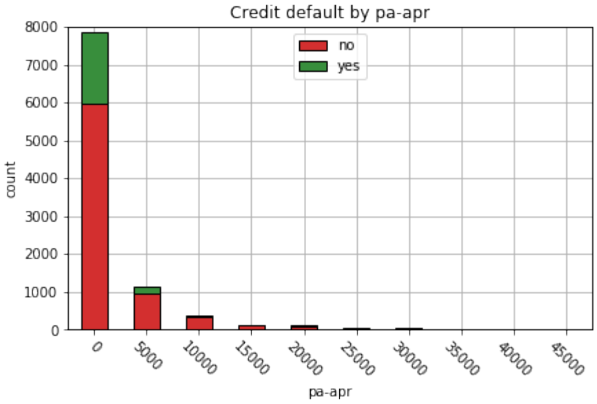
The distribution of the payment status over the six months are all very similar to each other so we only illustate the first one for simplicity. The most frequent bin is always the 0 and around 90% of the customers always lies between -2 and 0.

Bill Amount

Amount of bill statement (expressed in NT dollar). Six continuous attributes, again one for each month between April and September.

The bill amount is the sum of all the purchases, payments and other debits and credits made to your credit card account within the billing cycle. The values are in range $[-209051, 616836]$. As in the payment status, the distribution of the bill amount are nearly the same over the six month. The most frequent by is always the 0 (from 0 NTD to 49999 NTD). The mean increases according to the month, it start from 38621 of April up to 51490 of September. On ther other hand the credit default rate is always around the 20% for all the bins (except some border case for small bin).





Payment amount

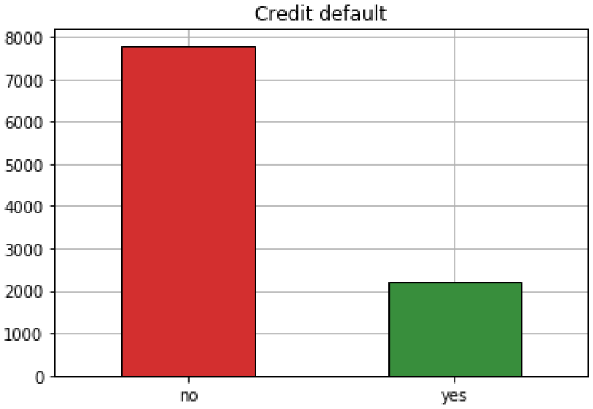
Amount of payment (expressed in NT dollar).

Six continuous attribute, one for each month between April and September. It represent the sum of all the payments made by the customer with the credit card in the last month. The distribution of the payment status over the six months are again all very similar to each other. The most frequent bin is always the 0 (from 0 NTD to 4999 NTD), which always include around the 75% of the customers. The mean is around 4719 and 5973 without any particular distribution other the months.

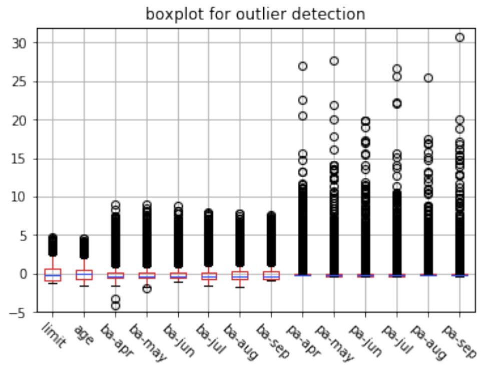
Credit default

Credit card default.

A boolean attribute that represents the credibility of the customer. The credit card default is a status that is applied to the customer when a list of terms are missed (like make the minimum required payment), and it affects for example your possibility to demand for a loan or to get another credit card. In the dataset we have 2212 customer in credit card default and 7788 not.



2.2 Assessing data quality

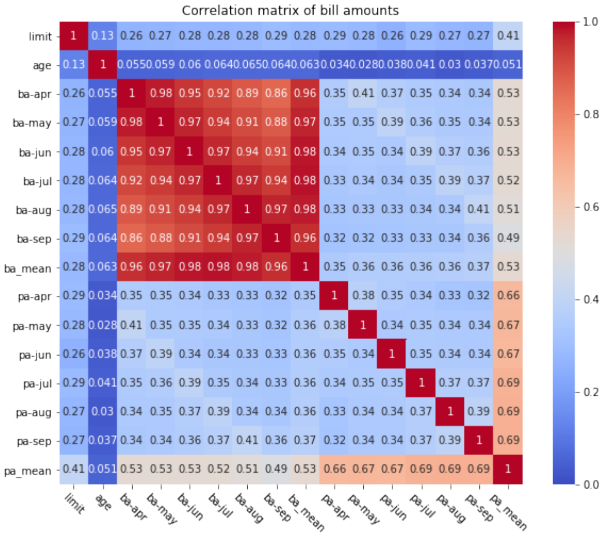


In this part we have checked for duplicates, missing values and outliers, in order to reduce the quantity of data and to avoid that their presence could negatively affect the analysis.
Regarding the duplicates, in the dataset there are only two identical records so we decide to not remove the single duplicate as it is not a big problem. Regarding the missing values we have 100 records with missing sex, 127 records, 1822 records with missing status and 951 records with missing age. We decided to not remove them as they are a big part of the dataset, instead we replace the missing ages with the average age (which is 35.49). We set the missing statuses to *others* as we think that it can include some particular cases (like engaged or cohabitant). We again set the missing educations to *others* and the missing sex to female as they are the more frequents in the dataset.
For the outliers we first normalize all the continuous attributes using the z-score algorithm (in this way we can visualize all the attributes in the same scale). Then we plot them in a boxplot and we can see that in the dataset only 3 values are under the lower adjacent value, instead over the upper adjacent value we have 50 limits, 211 limits and an average of 895 for the bill/payment amounts. As the attributes related to the payment amounts are the only one that have values over the 10 times the std, we decide to remove them to not have to work with outliers too high. After the cleaning part the dataset is composed of 9121 customers.

2.3 Variables transformations

We decide to transform the *credit_default* attribute from yesno to 12 in order to better visualize them in the following plots. For the same reason and to limit the dispersion we decided to divide in bin the continuous attributes. In particular we divide ages in bin of 10 years, payment amounts in bin of 1000 NTD, limits and bill amounts in bin of 10000 NTD.
As last transformation we have normalized all the continuous attribute using the min-max scaling.

2.4 Correlations and redundant variables



Analyzing the correlation matrix of all the continuous attribute we can clearly see that all the attributes related to the bill amounts are strongly correlated.
All the pairs of bill amounts have a correlation between 0.8 and 0.95 so we decide to analize further this 6 attributes.
So we add two new attributes to the dataset called *ba_mean* and *pa_mean* calculated as the mean of all the bill/payment amount for each customer.
We again plot the correlation matrix and we can see that both *ba_mean* and *pa_mean* have a high correlations with their relative attributes. For this reasons we drop all the twelve bill/payment amounts attributes and introduce their mean as new attributes.

3. Clustering

3.1 K-means

3.1.1. Choice of attributes and distance function

We have selected 3 attributes to performe the clustering via the K-Means algorithm. In particular we have ignored the categorical attributes as there isn't a proper metric to define a distance function over these kinds of attributes.
The used attributes are only: *limit*, *ba_mean*, *pa_mean*. Note that we have excluded the age in this list as it is the only one that isn't measured in NTD.
For these attributes we used the euclidean distance as metric function as it is more natural for this kind of attributes.

3.1.2. Choise of the best value of k

To choose the best value of k we have plotted for each K between 2 and 50 the values of the SSE and the silhouette score. Each value is calculated as the best among 10 runs (with the lowest SSE) and with a maximum of 300 iterations of the K-Means algorithm.

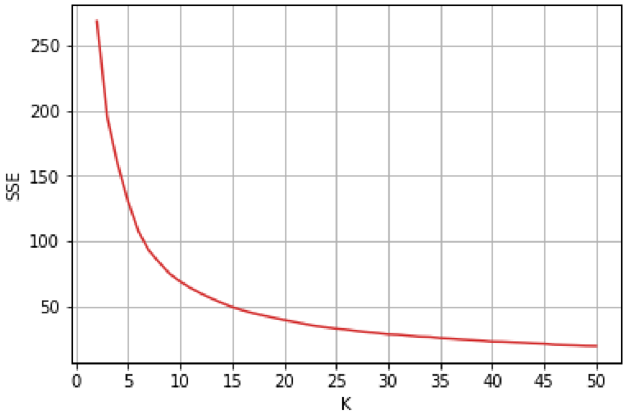


Figure 3.1: SSE value for each K.

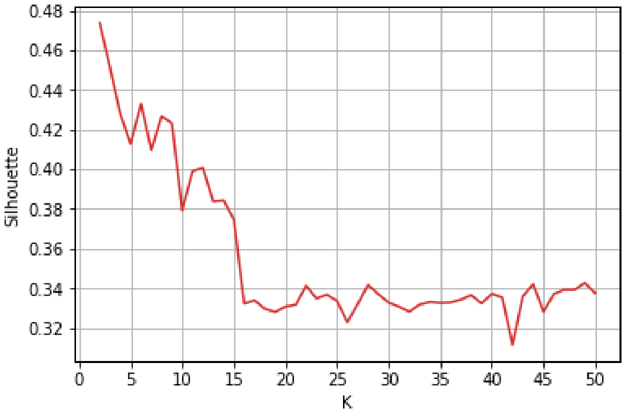
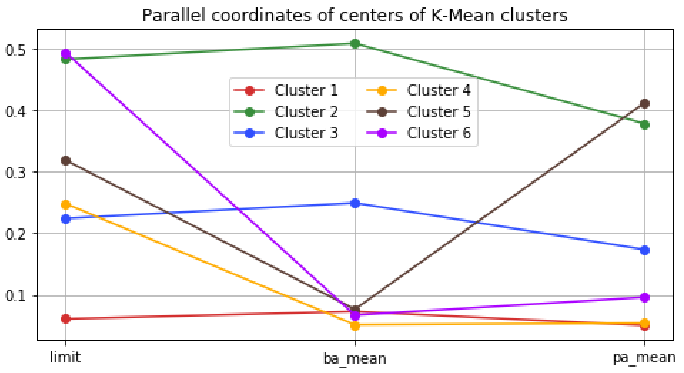


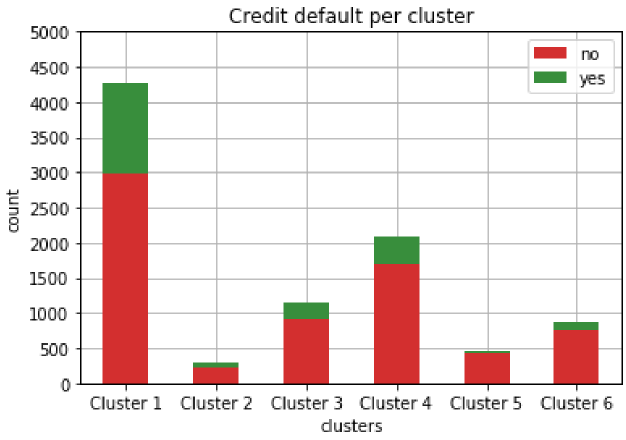
Figure 3.2: silhouette score for each K.

Using the elbow method we decide to set $k = 6$ as it represents a good trade-off between SSE and data interpretability.

3.1.3. Cluster analysis

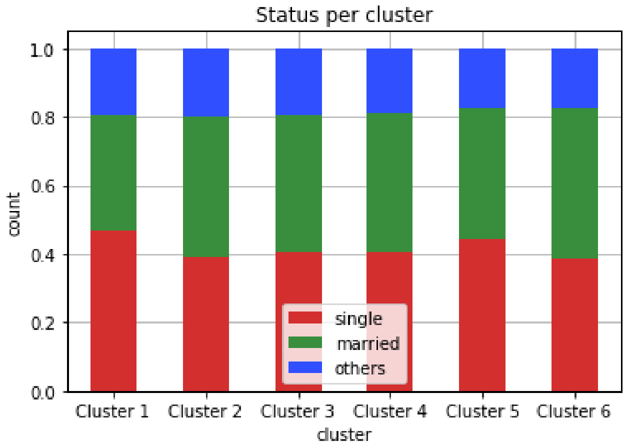
To better visualize the obtained clusters we have plotted the centers in parallel coordinates.
From a first point of view we can see that all of the centers are separated in three class (low, medium, high) for all of the three used attributes.
We want to go further in order to understand the different kind of customers and if there are some interesting correlation.

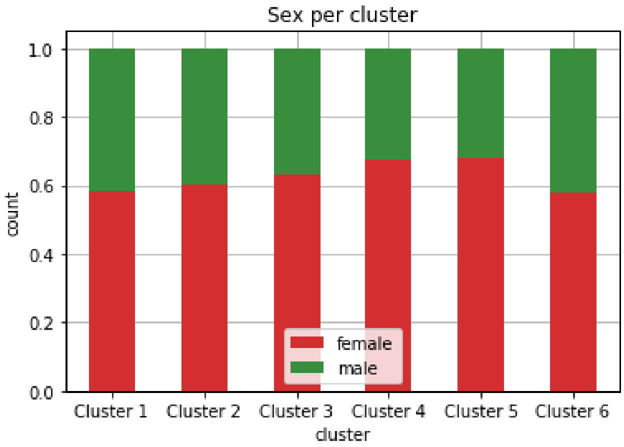




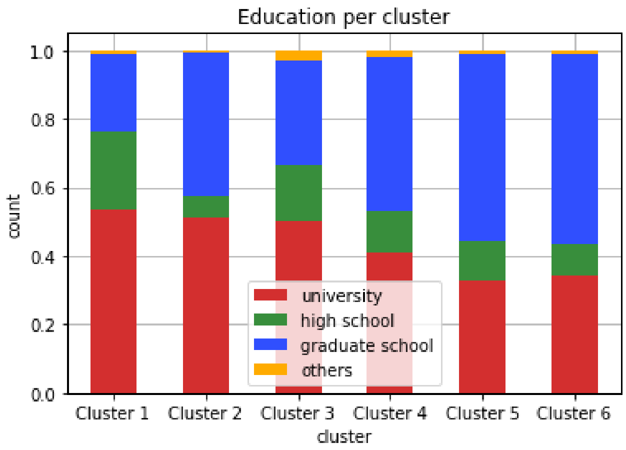
We have also plotted the number of credit defaults for each cluster to trying to understand the distribution of the customers. The cluster with the most credit default is the first one, with a ratio of 30% and the cluster with the lowest credit default is the fifth, with a ratio of 6%.

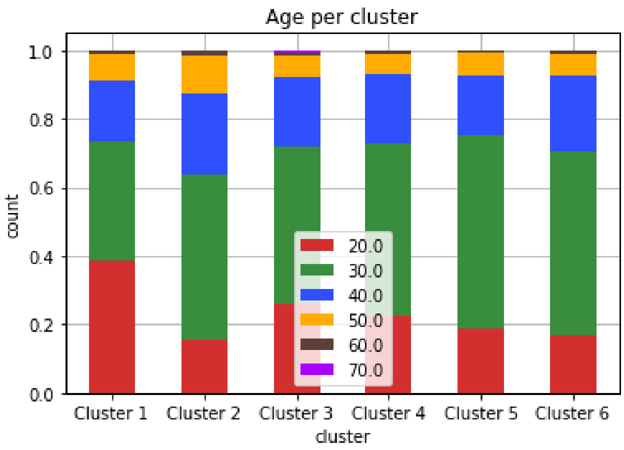
From the figures below we can see that the distributions of sex and status are the same for all the clusters, in particular their distributions are the same in the whole datasets so we can't use these attributes to make any assumptions over the obtained clusters.





More useful are instead the distributions of the attributes relative to the educations and ages:





We can discuss each cluster individually to see the most interesting properties.

Cluster 1 is the biggest cluster, with 4276 total elements include nearly half of the customers, and it is also the cluster with the highest credit default ratio (40%). From the parallel coordinates plot we can see that the customers in this class have the credit cards with the lowest limit and also have the lowest bill/payment amounts over the six months. The lowest limit could be that this cluster is the one with most of young people (40% are below the 30), but also is the cluster with most customer with an education in university and in high school. The high positive credit default ratio could be a combination of the distribution of these attributes (younger and university/high school).

Cluster 2 with a total of 283 customers is the smallest cluster. It has a credit default ratio of 21% and it is the cluster with the older customers, nearly 40% of the customers have an age higher that 40. From the parallel coordinates we can see that is the only cluster with high limit and high bill amount and payment amount mean.

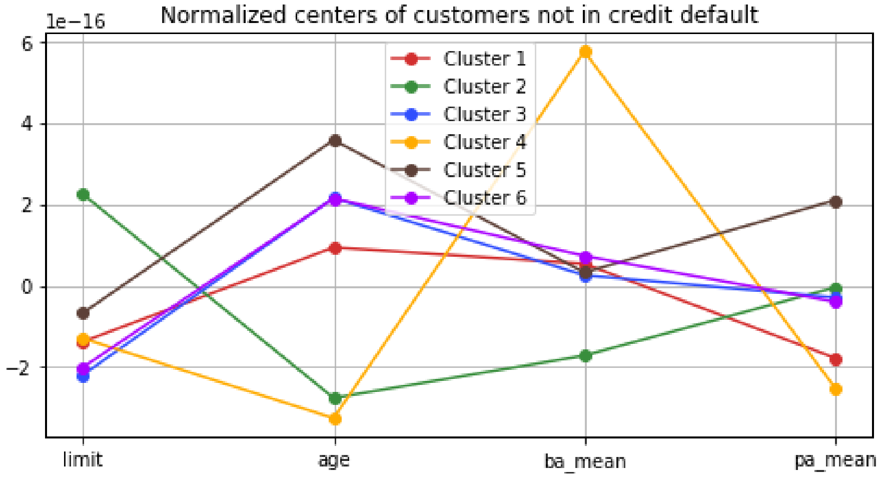
Cluster 3 has a total of 1143 customers and a positive credit default ratio of 19%. In this cluster we have customers with a medium value of all the attributes.

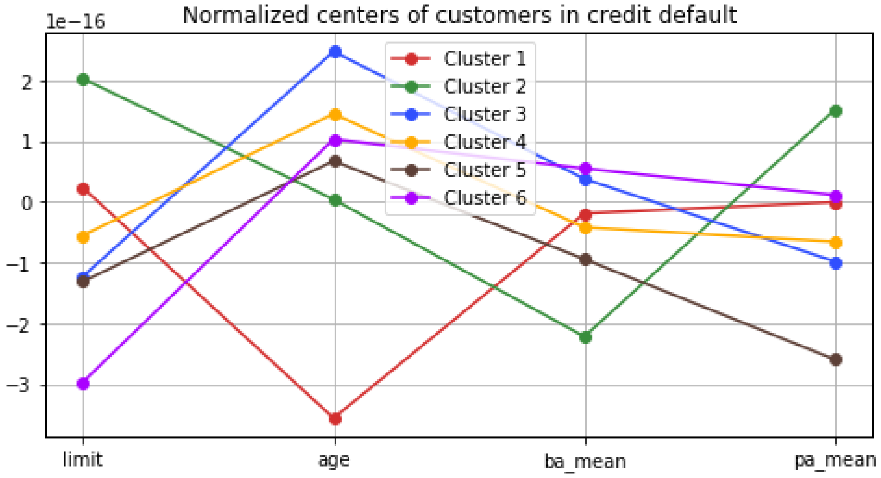
Cluster 4 with a total of 2083 customers is the second cluster for size and is has a positive credit default ratio of 21%. In this cluster we have customers with a medium credit card limit and with the lowest bill/payment amount mean.

Cluster 5 has a total of 457 customers and is the cluster with the lowest positive credit default ratio (only 6%). In this cluster we have customers with a credit card of medium limit, with a low bill amount mean but with the highest payment amount mean. From a point of view of the education in this cluster (and in the last cluster) we have the lowest number of education in university but the highest number of education in graduate school.

Cluster 6 has a total of 879 customers. It is the cluster with the highest limit in the credit cards but it has a low bill/payment amount mean.

We decided to analize further the obtained cluster, so we separate the customers with a positive credit default from the customers with a negative credit default and plot all the normalized centers of the continuous attributes in two separated parallel coordinates plots:





The most interesting cluster are Cluster 1 and Cluster 5, which are the ones with the highest and lowest positive credit default ratio, so we decided to analyze only these two clusters.

For the Cluster 1 the customers in credit defaults have an higher credit card limit and payment amount mean but a lower average age (compared to the customers not in credit default from the same cluster).

For the Cluster 5 the customers in credit defaults have a lower age and bill/payment amount mean (compared to the customers not in credit default from the same cluster).

3.2 DBSCAN

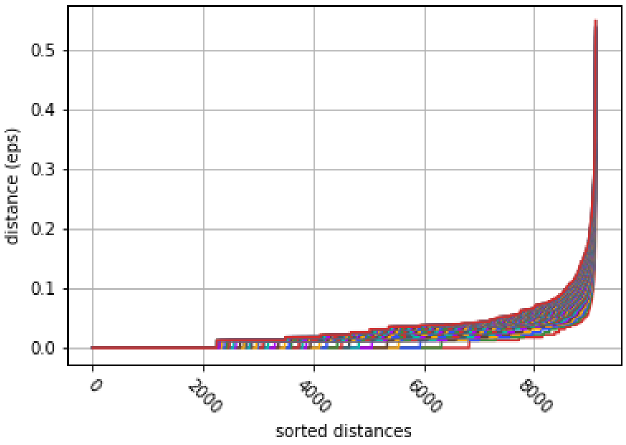
3.2.1. Choice of attributes and distance function

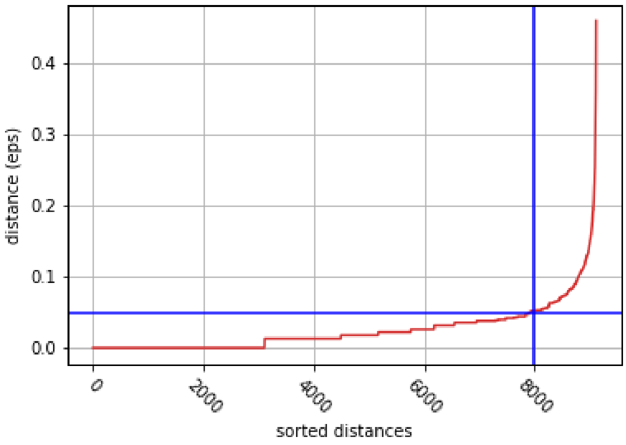
For the clustering via the DBSCAN algorithm we have used the same attributes of the K-Means algorithm (*limit*, *ba_mean* and *pa_mean*). As metric we have used again the euclidean distance for the same reasons.

3.2.2. Study of the clustering parameters

In order to execute the DBSCAN algorithm we need to choose the values of two parameters: *epsilon* for the radius of the circle for each point and *min-pts* as the minimum number of points (including itself) withing the radius to consider a customer as core point.

To choose the best value of *epsilon* we have plotted, for each value of *min-pts* in range $[2, 50]$ the sorted distances from the *min-pts*-th point.

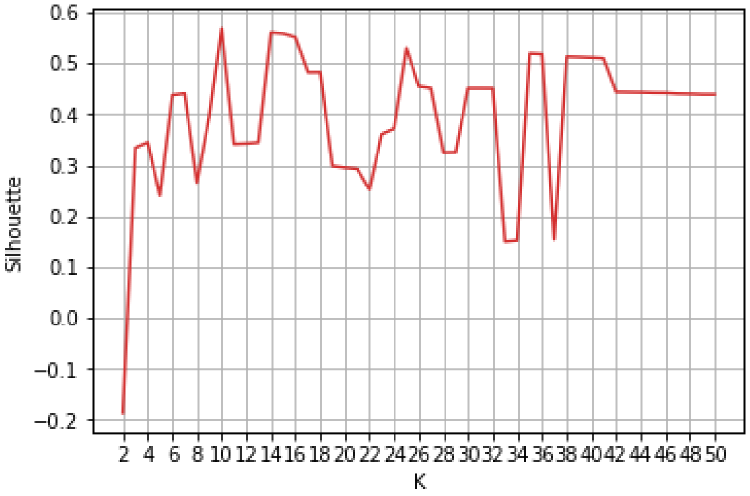




From the plot on the left we can see that all the curves have an elbow at x close to 8000, for this value of x the corresponding *epsilon* is 0.05.

To choose the best value of *min-pts* now we can fix *epsilon* to 0.05 and plot, for every *min-pts* in range $[2, 50]$, the silhouette coefficients.

By analyzing the plot on the left and with some practical test over the dataset we decided to use *min-pts=10*, in this way we obtained 10 clusters.



3.2.3. Characterization and interpretation of the obtained clusters

3.3 Hierarchical clustering

3.3.1. Choice of attributes and distance function

3.3.2. Discussion of dendograms using different algorithms

3.4 Evaluation of clustering approaches and comparison of the clustering obtained

4. Association Rules Mining

4.1 Frequent patterns extraction with different parameters

As we have a lot of categorical attributes with a lot of different values we don't have any frequent pattern with an high support.
The firsts frequent itemsets appears when we set the minimum support to 40%, in this case we have a total of 34 itemsets, the first 10 are:

1. support: 49.55%, items: (ps-jun=0, ps-may=0).
2. support: 48.99%, items: (ps-apr=0, ps-may=0).
3. support: 48.44%, items: (sex=female, default=false).
4. support: 48.21%, items: (ps-jul=0, ps-jun=0).
5. support: 47.69%, items: (ps-sep=0, ps-aug=0).
6. support: 47.58%, items: (ps-aug=0, ps-jul=0).
7. support: 45.75%, items: (ps-apr=0, ps-jun=0).
8. support: 45.42%, items: (ps-jul=0, ps-may=0).
9. support: 45.31%, items: (ps-aug=0, ps-jun=0).
10. support: 44.81%, items: (ps-may=0, default = false).

In particular we have 9 itemsets with default equals to false and none with credit default equals to false.

Choosing a minimum support of 30% results in 131 itemsets and half of them include the credit default equals to false.

The first frequent itemsets that contains the credit default equals to true is (sex=female, default=true) with a support of 13%, unfortunatelly we don't any significant and not naive frequen itemset with a positive credit default so we can only focus on the other ones.

4.2 Discussion of the most interesting frequent patterns

An interesting frequent itemset is the one which have all the six payment statuses equals to 0 and default to false with a support of 30.13%. This means, due to the properties of the support, that all its subsets (in particular the ones which include default to false and one or more payment status to zero) are frequent itemsets too.

Exluding itemsets which include only payment status attributes or the naive ones, we have also other interesting itemsets, for example:

1. support: 22.44%, items: (education=university, sex=female, default=false).
2. support: 21.29%, items: (age=30, sex=female, default=false).
3. support: 20.78%, items: (status=single, sex=female, default=false).

We can see that there are some interesting correlation with the sex=female and others frequent attibutes (and of course, as said before, with all the payment status equals to 0).

4.3 Association rules extraction with different values of confidence

As a consequence of what we have seen before, from practical experiments in the dataset we don't have any significant rule which include a positive credit default, so we again concentrate on the rules with default=false.

Trying with some different combination of support/confidence we can find some interesting rules. Rules for which it is default=false:

- (sex=female, at least 4 different payment statuses equals to 0):
  support of 20% and confidence of 90%.
- (sex=female, ba_mean=50000 NTD, at least 4 different payment statuses equals to 0):
  support of 2% and confidence of 95%.

Rules that doesn't include credit default attribute:

- (ps-sep=0, at least two payment statuses equals to 0 (from april to july)) → (ps-aug=0): support of 30% and confidence of 99%.

4.4 Discussion of the most interesting rules

We can see again that the most interesting attribute values in the rules are sex=female and payment status=0 for all the months. There are a lot of rules with both high accuracy and support which consists in combinations of these two kind of attributes.

A couple of rules that don't involve payment amounts:

- (education=graduate school, sex=female) → (credit default = false):
  accuracy 81.68%, support 16.87%
- (age=30-39, sex=female) → (credit default = false):
  accuracy 80.71%, support 21.29%

4.5 Use the most meaningful rules to replace missing values

The dataset doesn't contains a lot of missing values related to the most interesting rules, so we decide to simulate the substition, in this way we can evalute the accuracy of these rules.

- (age=30-39, status=married, education=university) → (sex=female):
  accuracy 73.71%, support 3.36%
- (payment status equals to 0 from april to august) → (ps_sep=0):
  accuracy 91.85%, support 36.59%
- (payment status equals to 0 from april to july) → (ps_aug=0, ps_sep=0):
  accuracy 84.18%, support 39.92%
- (payment status equals to 0 from april to june) → (ps_jul=0, ps_aug=0, ps_sep=0):
  accuracy 76.43%, support 43.97%
- (payment status equals to 0 from april to may) → (ps_jun=0, ps_jul=0, ps_aug=0, ps_sep=0):
  accuracy 68.60%, support 48.99%

4.6 Use the most meaningful rules to predict credit card defaults

We first decide to test some rules over only the attributes relative to the payment status:

- (all 6 payment statuses equals to 0) → (credit_default = false):
  accuracy 89.66%, support 33.61%.
- (at least 5 payment statuses equals to 0) → (credit_default = false):
  accuracy 85.88%, support 43.18%.
- (at least 4 payment statuses equals to 0) → (credit_default = false):
  accuracy 83.14%, support 50.66%.
- (at least 3 payment statuses equals to 0) → (credit_default = false):
  accuracy 80.90%, support 57.37%.
- (at least 2 payment statuses equals to 0) → (credit_default = false):
  accuracy 78.98%, support 64.93%.
- (at least 1 payment statuses equals to 0) → (credit_default = false):
  accuracy 78.61%, support 69.88%.

With an high accuracy and support we can conclude that these are good rules to predict the negative credit default.

By introducing the condition sex=female in the previous rules doesn't change a lot the results, for example:

- (sex=female, all 6 payment statuses equals to 0) → (credit_default = false):
  precision 90.22%, support 20.86%
- (sex=female, at least 5 payment statuses equals to 0) → (credit_default = false):
  precision 88.80%, support 26.58%

In particular we have a big decrease in support with a small increase for the precision.

5. Classification

5.1 Choice of attributes for the decision trees

For the classification task we decide to use the Random Forest Classifier, to choose the model iperparameters we have divided before the dataset in training set (80% of the customers) and test set (the remaining 20% of the customers), then we have executed a grid search over 8 iperparameters, for a total of 2304 combinations. In particular we have searched over:

Parameter	Description	Values
n_estimators	The number of trees in the forest	100
bootstrap	Whether bootstrap samples are used when building trees	True
max_depth	The maximum depth of the tree	$[6, 14]$
max_features	The number of features to consider when looking for the best split	[1, #features]
min_samples_split	The minimum number of samples required to split an internal node	$\{10\%, 1\%, 0.1\%\}$
min_samples_leaf	The minimum number of samples required to be at a leaf node	$\{5\%, 0.5\%, 0.05\%\}$
criterion	The function to measure the quality of a split	gini, entropy
class_weight	Weights associated with classes	balanced, not balanced

Then we have executed a 10-cross validation over the training set for each combinations in order to find the best models. In particular we best ones are:

- First model:

  Accuracy in validation phase: 81.48% (std: 0.02)

  Parameters: (bootstrap=True, class_weight=None, criterion='entropy', max_depth=12, max_features=1, min_samples_leaf=0.005, min_samples_split=0.01, n_estimators=100)

- Second model:

  Accuracy in validation phase: 81.46% (std: 0.02)

  Parameters: (bootstrap=True, class_weight=None, criterion='gini', max_depth=11, max_features=1, min_samples_leaf=0.0005, min_samples_split=0.01, n_estimators=100)

For both of the models the two most important feature are ps-sep and ps-aug, these are a confirmations of the rules seen in the chapter before. In particulare the importance of features are:

Attribute	Model 1	Model 2
ps-sep	25.13%	52.22%
ps-aug	16.77%	16.16%
ps-jul	15.06%	3.38%
ps-may	8.53%	2.13%
ps-jun	7.70%	3.54%
ps-apr	6.85%	2.40%
pa_mean	8.75%	5.11%
limit	5.68%	5.84%
ba_mean	2.52%	2.97%
age	1.71%	3.17%
education	1.40%	1.37%
sex	0.99%	0.62%
status	0.84%	1.03%

In both models the attributes related to the status, sex, education and ages have a very low importance (all under the 3.17%). Instead, as said before, the attributes ps-sep and ps-aug have a very importance (in particular in the second model ps-sep have an importance of 52%).

5.2 Results validation with test set and discussion of the best model

We have trained both selected models over the entire training set in order to predict the credit default over the test set. In this way we can evalute the models using records not yet observed.

Model 1 (Accuracy 0.7986):

	precision	recall	f1-score
no	0.81	0.96	0.88
yes	0.68	0.26	0.37
avg	0.75	0.61	0.63

Model 2 (Accuracy 0.8238):

	precision	recall	f1-score
no	0.84	0.95	0.89
yes	0.70	0.42	0.53
avg	0.77	0.69	0.71

From this results we can see that the second model performe better over the test set, in particular it has a higher results in the recognition of credit_default=yes.
We tried to train again the second model but this time with a max_depth equals to 5 (instead of 11), and the results are better than expected:

Model 3 (Accuracy 0.8246):

	precision	recall	f1-score
no	0.84	0.95	0.89
yes	0.71	0.42	0.53
avg	0.78	0.69	0.71

Halving the height of the model 2 we obtain a simplier model that slightly improve the recognition credit_default=yes.

5.3 Decision trees interpretation

cose

6. Conclusion

Conclusione dove parlo di cose.