Will our employee leave the company?

Data Mining 2017-2018

University of Pisa

Gabriele Leone - 563955

Alessandro Riglietti - 561751

Abhi Shek - 561797

Federica Trevisan - 568019

Contents

Chapter 1

Data Understanding

The dataset has been obtained from kaggle.com. It comes along with 14999 records, each one representing an employee of the company.

Each record is described by 10 features.

In the following table (Table 1.1) the types of each feature are detailed.

Table 1.1: Dataset attributes

Type	**Attributes**
Binary (numeric)	Work_accident, promotion_last_5years, left
Discrete (numeric)	number_project, time_spend_company
Continuous (numeric)	satisfaction_level, last_evaluation, average_monthly_hours,
Categorical (non numeric)	sales, salary

The meaning of the 10 attributes is now briefly introduced:

- *left* shows if an employee has left the company, it can be 0 if the employee is still working in the company and 1 otherwise; the main goal of our project is to predict the value of this variable and the reasons behind this.
- *Work_accident* shows if an employee has had any accident at work. As before, 0 means no accidents, 1 otherwise;
- *promotion_last_5years* is 1 if any promotion has been granted during the last 5 years at work, 0 otherwise;
- *number_project* shows the number of projects completed since the beginning of the career in that specific company;
- *time_spend_company* corresponds to the years spent at work in that company;
- *satisfaction_level* is an index of the personal satisfaction level, it can vary from 0 to 1;
- *last_evaluation* is the score of the last evaluation assigned to an employee by the company, also this variable can assume values from 0 to 1;
- *average_montly_hours* represents the average monthly hours spent at the workplace;

- *sales* is the categorical variable that identifies every department of the company. The departments are 10 in total: sales, technical, support, IT, marketing, research and development, product management, management, accounting and human resources;
- *salary* indicates the salary of the employee. It can be low, medium or high.

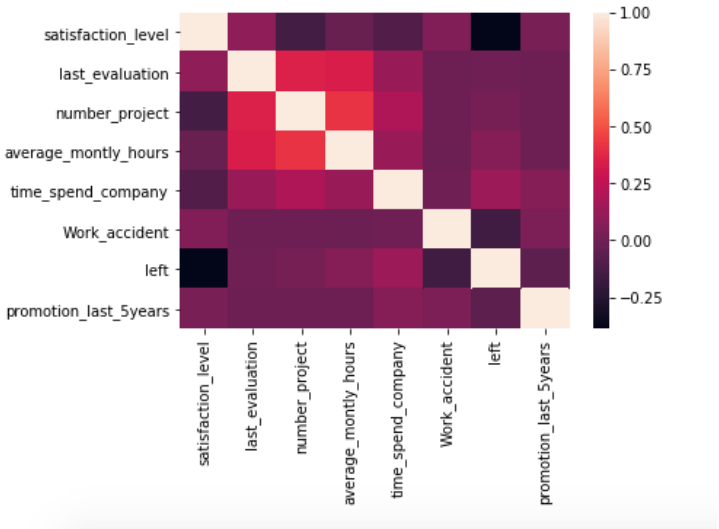
From a first look at the dataset we can assume that most of the employees are quite satisfied (mean 0.612834 and standard deviation 0.248631). We can also see that most of the employees have a high evaluation, whilst very few ones get a promotion or leave the company.

1.1 Outliers and Missing Values

In the dataset there are no missing values. Boxplots have been generated for each variable in order to detect outliers. Our attention has been focused on the boxplot for the variable *time_spend_company*, which is the only one showing, apparently, some outliers. These anomalous values are not a hint for a quality problem in the dataset, in fact they are not caused by typos or errors in the data. These records just represent a minority of the employees with high seniority in the company, but still not too few to be discarded. So we have decided to keep the records, since they represent useful information for the continuation of the project.

1.2 Pairwise Correlation

Figure 1.1: Correlation Matrix



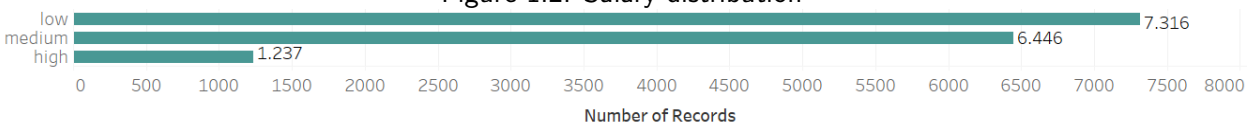
A correlation matrix (Fig. 1.1) has been computed, but no couples of attributes have a correlation coefficient higher than the default acceptance threshold, which is equal to |0.80|. Therefore there are no strong correlations and none of the attributes will be discarded. In fact the strongest correlation is equal to 0.417211 and it is verified between the attributes *average_montly_hours* and *number_projects*. This can lead us to think that the more projects an employee has had in charge since the beginning of the

work career, the more time they have spent working. This may seem obvious from a logical point of view, but it actually is not because the dataset is being simulated: we still have to investigate on the data.

1.3 Attributes Distribution and Statistics

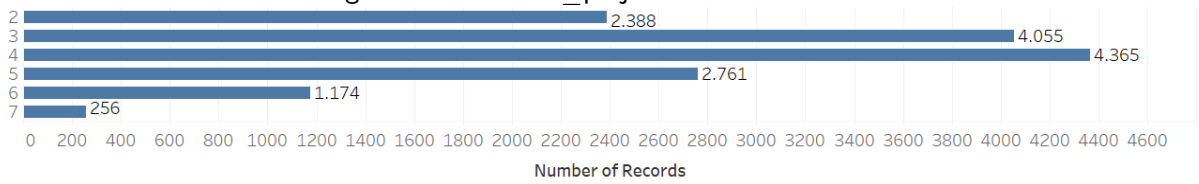
Salary: As shown in the bar plot (Fig. 1.2) it is clear that the majority of the workers get a low salary. The number of workers getting a medium salary is only slightly lower than the former, while it is important to notice that only few employees are highly paid.

Figure 1.2: Salary distribution



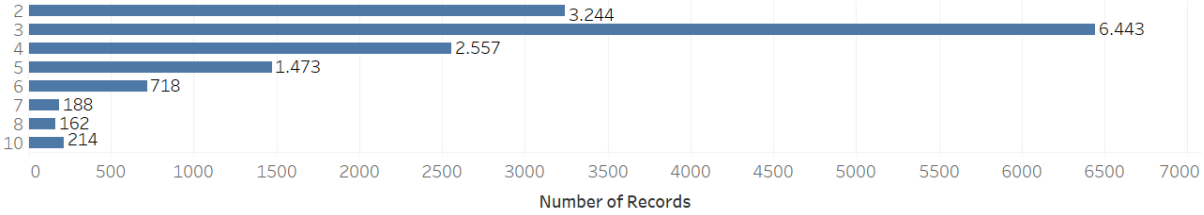
Number of projects: It is possible to observe a Gaussian-like distribution and that most of the workers have had in charge 3 or 4 projects (Fig. 1.3).

Figure 1.3: Number_projects distribution



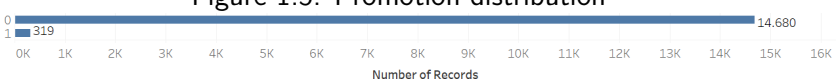
Time spent company: Most workers have been in the company for 3 years, and a number of workers equal to about half of the former have been working for 2 years. The remaining part of the employees have been working for more than 3 years (Fig. 1.4).

Figure 1.4: Time_spend_company distribution



Promotions: The bar plot (Fig 1.5) shows that very few employees got a promotion during their employment at the company.

Figure 1.5: Promotion distribution

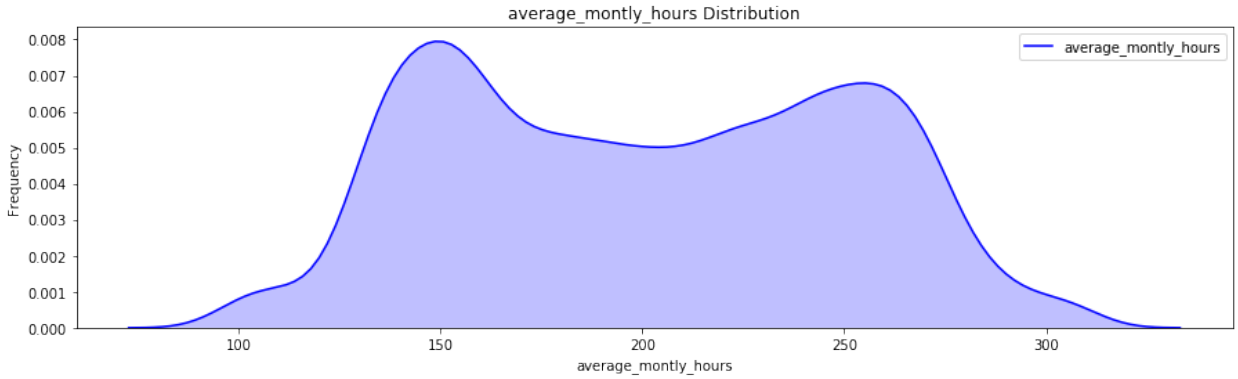


Work accident: The bar plot (Fig. 1.6) shows that work accidents are unlikely to happen.

Figure 1.6: Work accident distribution

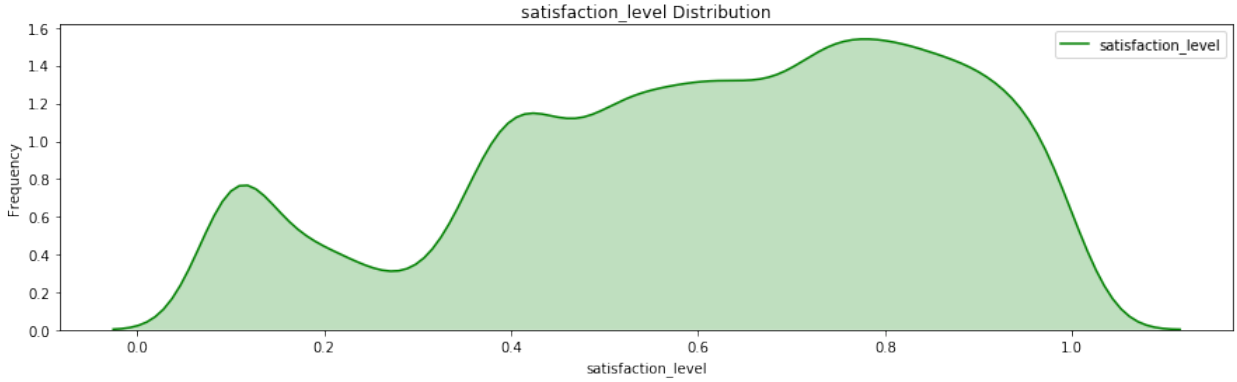
Average monthly hours: The plot (Fig. 1.7) shows that the majority of the employees work between 150 and 250 hours monthly and that the attribute *average_monthly_hours* follows a bimodal distribution.

Figure 1.7: Average monthly hours distribution



Satisfaction level: The plot (Fig. 1.8) shows that most of the values of the attribute *satisfaction_level* are comprised between 0.4 and 1, with a remarkable peak near 0.15.

Figure 1.8: Satisfaction level distribution



Departments: The bar plot (Fig. 1.9) of the variable *sales* shows us that the most popular departments are: sales, technical and support. The remaining ones have no significative difference in popularity.

Last evaluation: The plot (Fig. 1.10) shows that the attribute *last_evaluation* follows a bimodal distribution, with the peaks being near the values 0.5 and 0.9.

Left: The bar plot (Fig. 1.11) shows that a quarter of the total number of employees left the company. The scatter plot (Fig. 1.12) shows the concentration of the employees who left (shown in green) with regard to the attributes *average_monthly_hours* and *satisfaction_level*.

Figure 1.9: Sales distribution

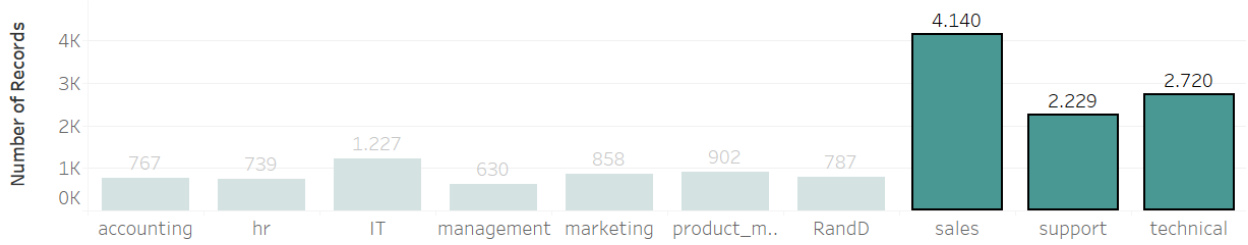


Figure 1.10: Last evaluation distribution

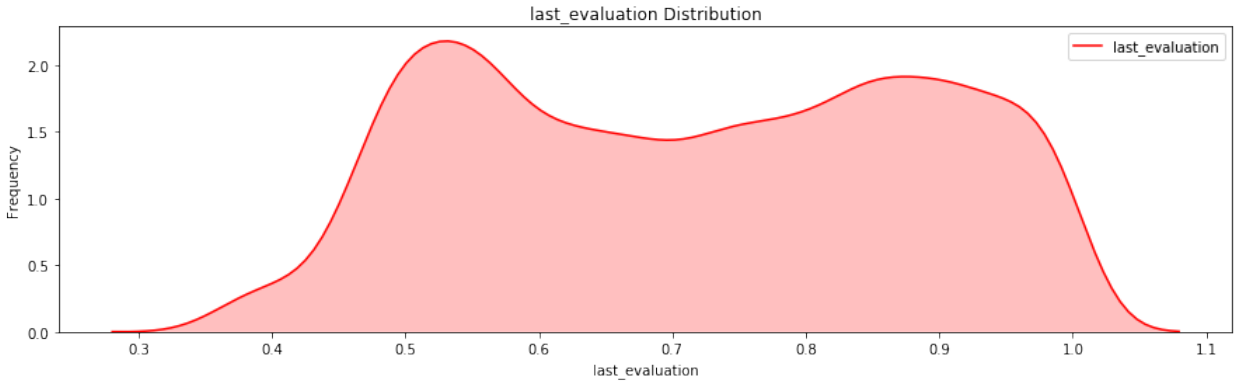
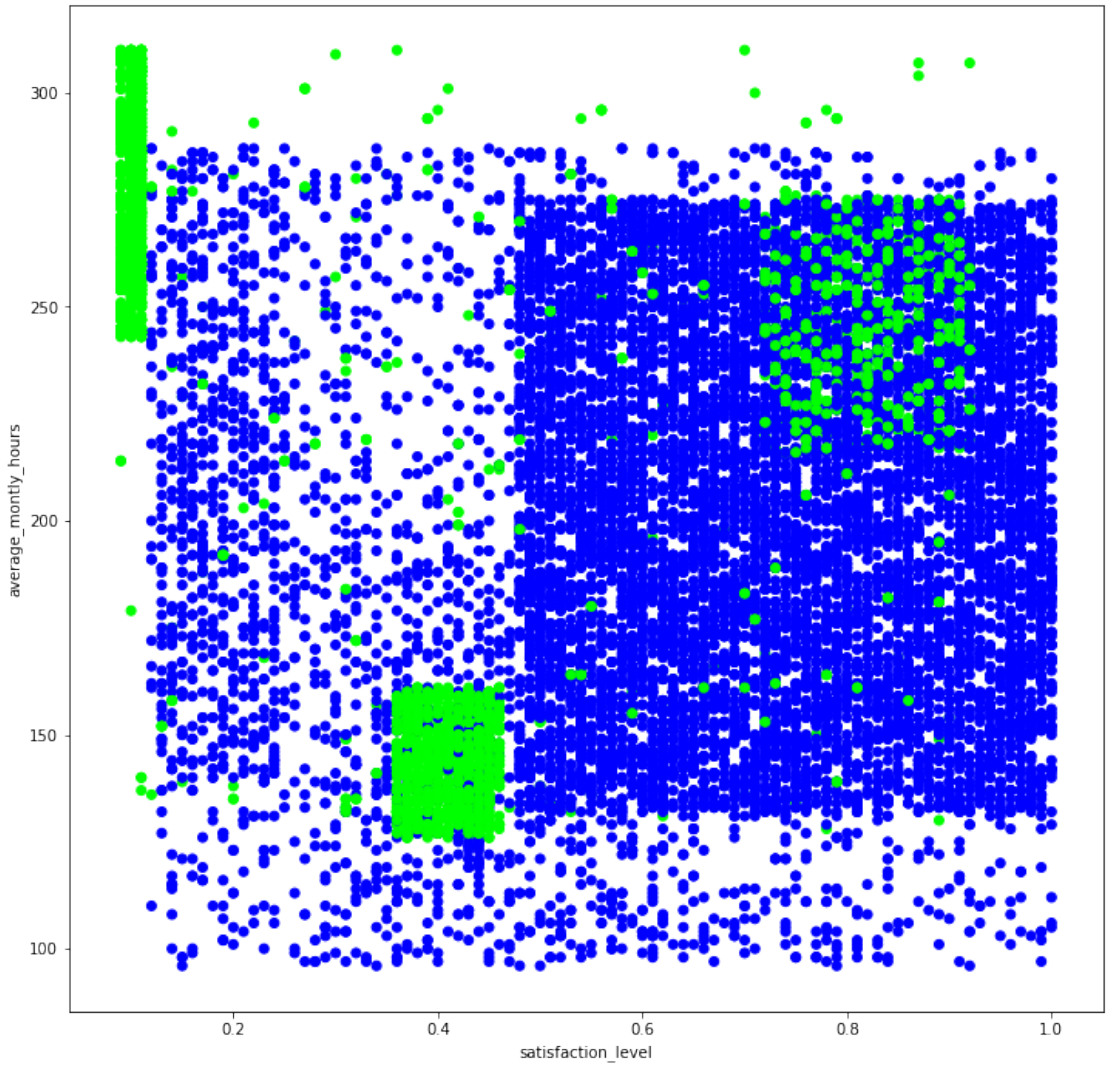


Figure 1.11: Left distribution

1.4 Variable manipulation

The proper manipulations have been applied differently with respect to each task of the project, since each task requires a different kind of manipulation. Therefore each chapter will have its own part dedicated to variable manipulation.

Figure 1.12: Left/Not left employees grouping



Chapter 2

Clustering

2.1 Data preparation

Before proceeding with the three clustering techniques taken into account (K-Means, DBSCAN, Hierarchical Clustering), the dataset had to be preprocessed. *Sales* and *salary* are categorical attributes, but it makes no sense to assign numerical values to them because they are not incremental, so they have been scrapped for the clustering. The attribute *left* has been removed since it is the target, as well as the attributes *promotion_last_5years* and *Work_accident* because of their binary values. The min-max normalization has been applied to the remaining attributes last_evaluation, satisfaction_level, average_montly_hours, number_projects and time_spend_company in order to have all the values in the dataset varying from 0 to 1.

2.2 K-Means

The SSE plot (Fig. 2.1) has made it possible to identify the right value of k, which is 10, by looking at the coordinates corresponding to the elbow. The algorithm has been used with the Euclidean distance because the Manhattan distance proved to yield worse results.

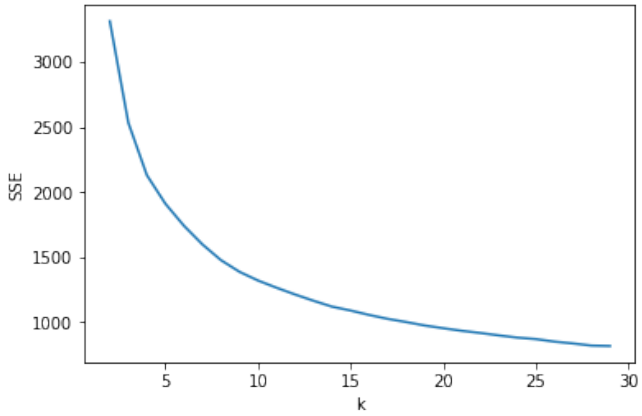


Figure 2.1: SSE Plot

The larger clusters are the less meaningful, mainly because the points they are made of share a wide range of values for most of the attributes considered in the computation. Therefore they are charaterized by a low degree of compactness and are not informative. The smaller clusters have a high degree of compactness and are the most meaningful.

By computing the satisfaction_level - average_monthly_hours scatterplot it is possible to see the 3 most meaningful clusters (Fig. 2.2): the first one (colored in yellow) includes people with a satisfaction level between around 0.35 and 0.45 and who work between around 125 and 160 average monthly hours. The second cluster includes people with a satisfaction level lower thah 0.2 and the highest average monthly hours, more than 240 (colored in green). The last cluster is for people having a satisfaction level higher than 0.75 and an average monthly work hours going from more than about 240 to 300 (colored in orange).

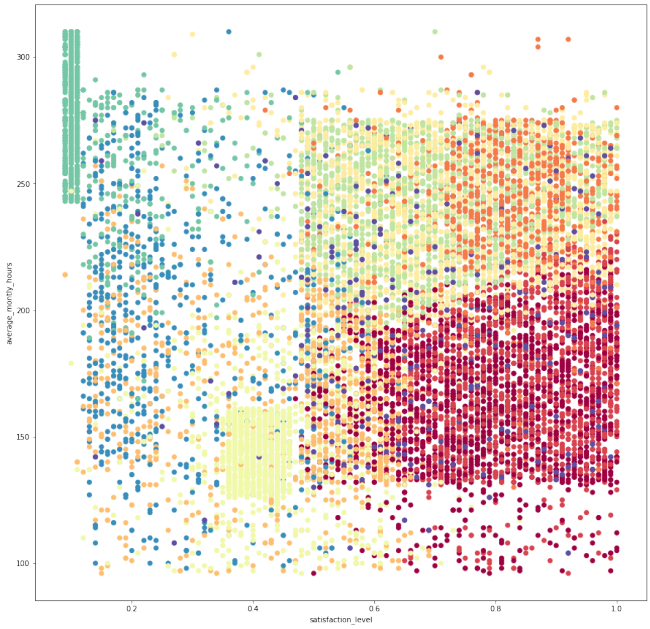


Figure 2.2: Clusters found by using K-Means

2.3 DBSCAN

In order to apply the DBSCAN algorithm, it is important to identify the right value of eps based on the value of K. The value chosen for K is equal to 4, because of two reasons: it is the standard in literature and the results obtained by using it have not been very different from the ones obtained using other values, such as 5, 6 and 8. The ideal value for eps is equal to 0.10 and it has been obtained using the **dbscan** package in R with the KNNDistPlot command. The eps value has been chosen by taking into account the elbow of the resulting curve (Fig. 2.3). The algorithm has been computed using the euclidean distance and five main clusters have been identified (Fig. 2.4).

Three of them are too sparse to be of any use and we can discard them as uninformative. The other two clusters detected by the DBSCAN are the one in the top-left corner of the scatterplot (colored in yellow) and the one in the bottom portion (colored in purple). The yellow one groups together the employees with less than 0.2 as *satisfaction_level* and between 240 and 310 as *average_montly_hours*, while the

purple one groups the employees with a *satisfaction_level* between 0.35 and 0.45 and values between 125 and 160 as *average_montly_hours*.

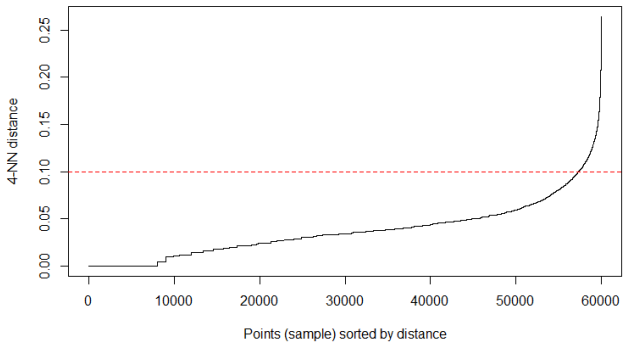


Figure 2.3: KNN distplot

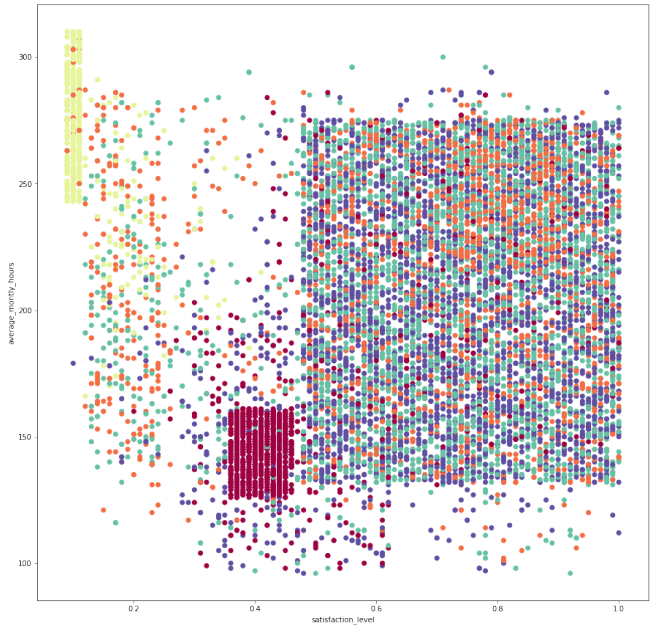


Figure 2.4: Clusters found by using DBSCAN

2.4 Hierarchical Clustering

The algorithm has been applied using different combinations of parameters on the whole dataset: every linkage method (ward, average, single and complete) and two types of distance (Euclidean and Manhattan) have been used; in almost every case the silhouette is very low.

The best strategy has been found using the ward linkage method with Euclidean distance. In this way the silhouette value is still low but higher than the ones obtained with the other combinations and this makes it possible to get three informative clusters, among others which are not.

Since the Python function imported from the *sklearn* library takes a K integer value as input, it is not required to operate a cut in the dendrogram to get the desired number of clusters (which is equal to K); the K used as input is equal to the K considered for the K-Means, that is 10.

By looking at the scatter plot (Fig. 2.5), the three informative clusters are the one in the top-left corner (colored in magenta), the one and in the bottom portion of the plot (colored in light blue) and the one near the top-right corner (colored in purple).

The magenta cluster is characterized by *satisfaction_level* values lower than 0.2 and *average_montly_hours* values between 240 and 310. The light blue cluster groups the employees having *average_montly_hours* values between 120 and 160 and *satisfaction_level* values between 0.35 and 0.45. The purple cluster is characterized by *satisfaction_level* values higher than 0.75 and *average_montly_hours* values between 240 and 300.

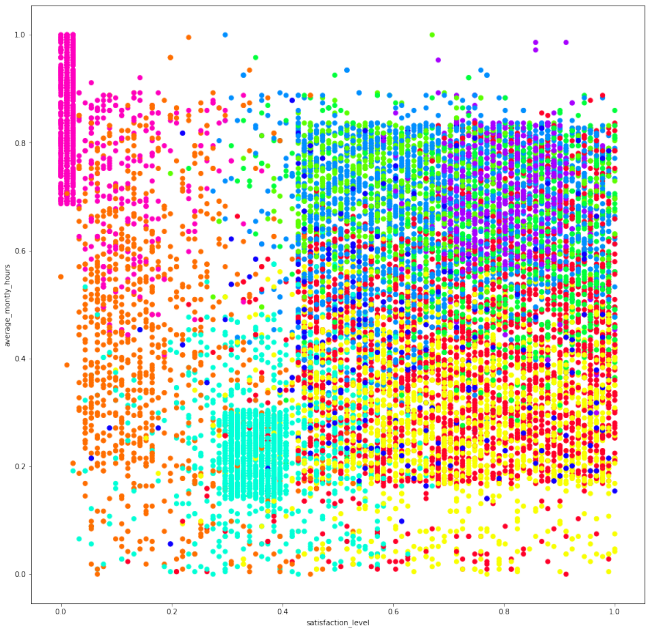


Figure 2.5: Clusters found by using Hierarchical Clustering

2.5 Comparison of the different clustering techniques

The three clustering algorithms proved to be moderately effective in detecting the three most interesting groups of leaving employees, among a lot of other uninformative and sparse clusters: the one made by employees who are highly unsatisfied and overworked, the one grouping the employees who are both overworked and quite satisfied and the last one, made by employees who are moderately unsatisfied and have a lower-than-average workload. The DBSCAN was unable to detect the second informative cluster, splitting it into many other different clusters; the K-Means proved to be the best clustering algorithm overall.

Chapter 3

Frequent Patterns and Association Rules

3.1 Data preparation

Before proceeding with the application of the Apriori algorithm for the extraction of frequent patterns, the dataset has been preprocessed in order to make it suitable for the pattern mining algorithm. In particular, the following manipulations have been made:

- the values of the continuous attribute *satisfaction_level* have been discretized into 5 bins: 'Very unsatisfied' [0 - 0.19], 'Unsatisfied' [0.2 - 0.39], 'Neutral' [0.4 - 0.59], 'Satisfied' [0.6 - 0.79] and 'Very satisfied' [0.8 - 1];
- the values of the continuos attribute *last_evaluation* have been discretized into 5 bins: 'Poor' [0.36 - 0.54], 'Fair' [0.55 - 0.65], 'Good' [0.66 - 0.75], 'Very Good' [0.76 - 0.90] and 'Excellent' [0.91 - 1];
- the values of the continuous attribute *average_montly_hours* have been discretized into 10 bins: [96-118],[119-140], [141-162], [163-184], [185-206], [207-228], [229-250], [251-272], [273-294] and [295-310];

The other attributes are left as they are; in addition all of the attributes have been considered in this task. After the discretization of the continuous attributes, an acronym or a symbolic abbreviation has been attached to the end of every value in the dataset, in order to obtain easily interpretable strings; a piece of the transformed dataset is depicted in the following figure (Fig. 3.1) (note that the last column *sales* is not totally visible in the picture).

satisfaction_level_cat	last_evaluation_cat	number_project	avg_monthly_hours_cat	time_spend_company	Work_accident	left	promotion_last_5years	sales
Insoddisfatto_SAT	Insufficiente_LE	2_NP	141-162_AMH	3_TSC	0_WA	1_L	0_P	sales_D
Molto soddisfatto_SAT	Buono_LE	5_NP	251-272_AMH	6_TSC	0_WA	1_L	0_P	sales_D
Molto insoddisfatto_SAT	Buono_LE	7_NP	251-272_AMH	4_TSC	0_WA	1_L	0_P	sales_D
Soddisfatto_SAT	Buono_LE	5_NP	207-228_AMH	5_TSC	0_WA	1_L	0_P	sales_D
Insoddisfatto_SAT	Insufficiente_LE	2_NP	141-162_AMH	3_TSC	0_WA	1_L	0_P	sales_D

Figure 3.1: Preprocessed dataset

3.2 Frequent Patterns extraction

The Frequent Patterns extraction and the Association Rules extraction have been carried out by means of the Apriori algorithm implemented in the pyfim library. After some testing, it has been decided to consider only itemsets with a minimum length higher or equal than 3, with the aim to get the most significant ones. Different values of support have been tested (10%, 20% and 30% of the total number of rows in the dataset) with regard to different types of itemsets (frequent, closed, maximal). The extracted closed itemsets have not shown any peculiarity worth discussing, so they are not covered in this section.

The most interesting patterns are shown in the following tables (Table 3.1, Table 3.2), along with a brief discussion.

Table 3.1: Number of extracted itemsets of length 3 by support and by type

Support	Frequent I.	Closed I.	Maximal I.
30%	7	7	7
20%	36	36	17
10%	189	189	75

Table 3.2: Most interesting itemsets

Support	Frequent I.	Maximal I.
30%	1. (0_L, 0_WA, 0_P) 2. (low_SA, 0_L, 0_P) 3. (medium_SA, 0_L, 0_P) 4. (3_TSC, 0_L, 0_P) 5. (low_SA, 0_WA, 0_P)	[same as frequent i.]
20%	1. (medium_SA, 0_L, 0_WA, 0_P) 2. (3_TSC', 0_L, 0_WA, 0_P) 3. (Satisfied_SA, 0_L, 0_P) 4. (Very satisfied_SAT, 0_L, 0_P) 5. (1_L, 0_WA, 0_P)	1. (2_TSC, 0_L, 0_P) 2. (1_L, 0_WA, 0_P) 3. (Poor_LE, 0_WA, 0_P) 4. (Neutral_SAT, 0_WA, 0_P)
10%	1. (1_L, low_SA, 0_P) 2. (1_L, 0_WA, 0_P) 3. (1_L, 3_TSC, 0_P) 4. (Very satisfied_SAT, 0_L, 0_WA, 0_P)	1. (2_NP, 1_L, 3_TSC, 0_P) 2. (185-206_AMH, 0_L, 0_P) 3. (207-228_AMH, 0_L, 0_P) 4. (Good_LE, 0_L, 0_WA, 0_P)

As it is possible to read from the table, the most common traits of the typical employee who doesn't leave his job are the following: no work accidents, no promotion and either low or medium salary.

Since the dataset is unbalanced towards employees who haven't left their job, it has been necessary to lower the support to 10%, both to retrieve a consistent number of itemsets which include people who left, who usually worked on a very low number of projects, and to extract association rules useful to predict if an employee is going to leave.

3.3 Association Rules extraction

As far as the Association Rules extraction is concerned, different combinations of support and minimum confidence have been tested; support may assume the values 20% and 10%, while the minimum confidence values considered are 80%, 70% and 60%. Only the rules having a lift value greater than 1.1 have been analyzed, since they are the most valuable ones. The same reason applies to the choice of confidence values, which are sufficiently high.

Rules not having the target attribute "left" as the only pattern in their right hand side are not covered in this report because of different reasons:

- there are no missing values to replace by means of association rules;
- extracted rules had a negligible value of lift, usually less than 1.05.

As already said, the support had to be lowered to 10% to extract rules with [Left = 1], while still preserving a high degree of confidence (80% or more).

Table 3.3: Extracted rules with 10% support

Support = 10%				
Min. conf.	Total no. of rules	No. of rules (lift >= 1.1)	No. of rules (Left = 1)	No. of rules (Left = 0)
80%	484	109	6	83
70%	544	117	6	135
60%	628	165	12	171

Table 3.4: Extracted rules with 20% support

Support = 20%				
Min. conf.	Total no. of rules	No. of rules (lift >= 1.1)	No. of rules (Left = 1)	No. of rules (Left = 0)
80%	114	20	0	18
70%	137	20	0	41
60%	153	28	0	49

While the Apriori algorithm extracted a great number of rules with [Left = 0] as their right hand side, due to the unbalanced dataset the maximum value for the lift is 1.29; on the other hand, it is worth noticing that all of the lift values of the rules having [Left = 1] as their consequent (which are 6 in total) are greater than 3.4. These rules predict the resignation of an employee if he has worked on only two projects during his employment at the specific company, which is shorter than 3 years, and he has been poorly evaluated.

Let's consider the rule {2_NP, 3_TSC} -> 1_L; it is applicable to 1854 records in the dataset, classifying them as leaving employees, but only 1528 of them are actually employees who left, so there are 326 false positives; the precision is still good (82% circa). On the other hand, the rule is not able to classify as leaving employees the other 1717 employees who actually left (the total number of the leaving ones is 3571), on which the rule is not applicable, leading to poor values of recall and accuracy. The same reasoning can be applied to the other rules having [Left = 1] as consequent; these rules cannot be applied to classify the entire dataset, since they have been extracted with an inadequate value of support. It is possible to say that the extracted rules alone are not good enough to build a good overall rule-based classifier.

–

Table 3.5: Extracted rules with Left = 1

Rules with consequent [Left = 1]		
Rule	Confidence	Lift
{2_NP, Poor_LE, 0_P} ->1_L	84,27%	3.539
{2_NP, 3_TSC, 0_WA} ->1_L	84,25%	3.538
{2_NP, 3_TSC, 0_P} ->1_L	82,43%	3,463
{2_NP, 3_TSC} ->1_L	82,41%	3,461

Table 3.6: Extracted rules with Left = 0

Rules with consequent [Left = 0]		
Rule	Confidence	Lift
{3_NP, 0_P} ->0_L	98,18%	1,29
{4_NP, 0_P} ->0_L	90,46%	1,19
{Satisfied_SAT, 0_P} ->0_L	90,47%	1,19
{Very satisfied_SAT, 0_P} ->0_L	86,03%	1,28
{2_TSC, 0_P} ->0_L	98,33%	1,29
{Fair_LE} ->0_L	88,13%	1,15
{Good_LE} ->0_L	97,39%	1,27

Chapter 4

Classification

The classification task has been carried out by means of the **sklearn** library in Python 2.7.

4.1 Feature Selection

Before proceeding with the learning of the Decision Trees, feature selection has been performed on the original dataset in order to remove attributes which are irrelevant or scarcely relevant for the purpose of this project.

Such attributes are the following: *sales*, *salary*, *Work_accident* and *promotion_last_5years*; thus, the attributes which have been considered for the learning of Decision Trees are the following, ordered by their relative contribution to the classification task in descending order:

1. *satisfaction_level*;
2. *time_spend_company*;
3. *last_evaluation*;
4. *number_project*;
5. *average_montly_hours*.

4.2 Decision Trees Learning and Validation

Different combinations of gain formulas and maximum depth values have been tested, with the aim of maximizing the classification performances.
The gain formulas considered in this section are the Gini coefficient and the Entropy coefficient, while the maximum depth may assume the values 4, 5, 6 and 8; this kind of pre-pruning is needed since higher values of the maximum depth may lead to overfitting. Thus, a total of eight Decision Trees have been trained and tested.

Since a proper test set, which would be needed to evaluate the performance of Decision Trees learned on the whole dataset, is not available, two main approaches have been followed to validate the proposed model:

- Hold-out validation: 70% of the dataset (10499 rows) has been used as training set, while the remaining 30% (4500 rows) has been used as test set; accuracy, precision, recall and f-measure have been computed afterwards;
- Cross validation: the dataset has been partitioned into 10 equal-sized folds: one of the folds is used as test set, while the remaining nine are used as training set. The process is repeated 10 times, using each fold exactly once as test set, and the resulting 10 accuracy values are then averaged to get a single accuracy value.

All of the results are summarized in the following two tables; the first one (Table 4.1) is relative to the tests carried out by considering the Gini coefficient, while the second one (Table 4.2) refers to the tests carried out by considering the Entropy coefficient. The accuracy, precision, recall and f-measure values are the results of the Hold-out validation techinque, while the last column refers to the accuracy value achieved by means of Cross validation.

Table 4.1: Results achieved by means of Gini coefficient

Depth	Accuracy	Precision	Recall	F-Measure	C.V.
4	96,64%	0,966	0,966	0,966	97%
5	97,31%	0,972	0,972	0,973	97%
6	97,6%	0,976	0,976	0,975	97%
8	98,09%	0,98	0,98	0,98	98%

Table 4.2: Results achieved by means of Entropy coefficient

Depth	Accuracy	Precision	Recall	F-Measure	C.V.
4	96,44%	0,964	0,964	0,964	96%
5	97,17%	0,971	0,971	0,971	97%
6	97,46%	0,974	0,974	0,974	97%
8	97,82%	0,978	0,978	0,978	98%

Other than the eight Decision Trees previously introduced, two Random Forest Classifiers have been trained, having each one 100 trees as estimators; one of them has been trained by considering the Gini coefficient as a measure of impurity, while the other has been trained by employing the Entropy coefficient as a measure for the information gain. The two Random Forests have been tested through cross validation, achieving almost the same level of accuracy (98%).

4.3 Decision Tree interpretation

For the sake of clarity, in this subsection only the Decision Tree learned by setting Gini coefficient as split criterion and 4 as the maximum depth of the tree is analyzed. Although the tree is not so deep, its performance in the classification task are remarkable, with over 96% accuracy; the low depth allows us to achieve a good level of readability.

As it is possible to understand by reading the resulting tree (Fig. 4.1), the employees who are more prone to resign are the ones belonging to the following two main groups:

1. employees who are quite unsatisfied (satisfaction_level lower than 0.46), having a low evaluation score (last_evaluation lower then 0.45) and a very low count of projects they have worked on

(number_project equal to 2);

2. employees who are moderately or highly satisfied with their work (satisfaction_level higher than 0.46), having a high evaluation score (last_evaluation higher than 0.80) working more than 9 hours a day on average (average_montly_hours higher than 216) and who have spent at least 4 years in the company (time_spend_company higher than 4.5).

4.4 Comparison between the proposed models

As shown in the Tables 4.1 and 4.2 in the subsection 4.2, there is little to no difference between Decision Trees learned by considering the Gini coefficient or the Entropy coefficient, while the change in the depth level of the tree yields slightly better results as the depth increases, at the expenses of both readability and risk, albeit low in this particular case, of incurring in overfitting.

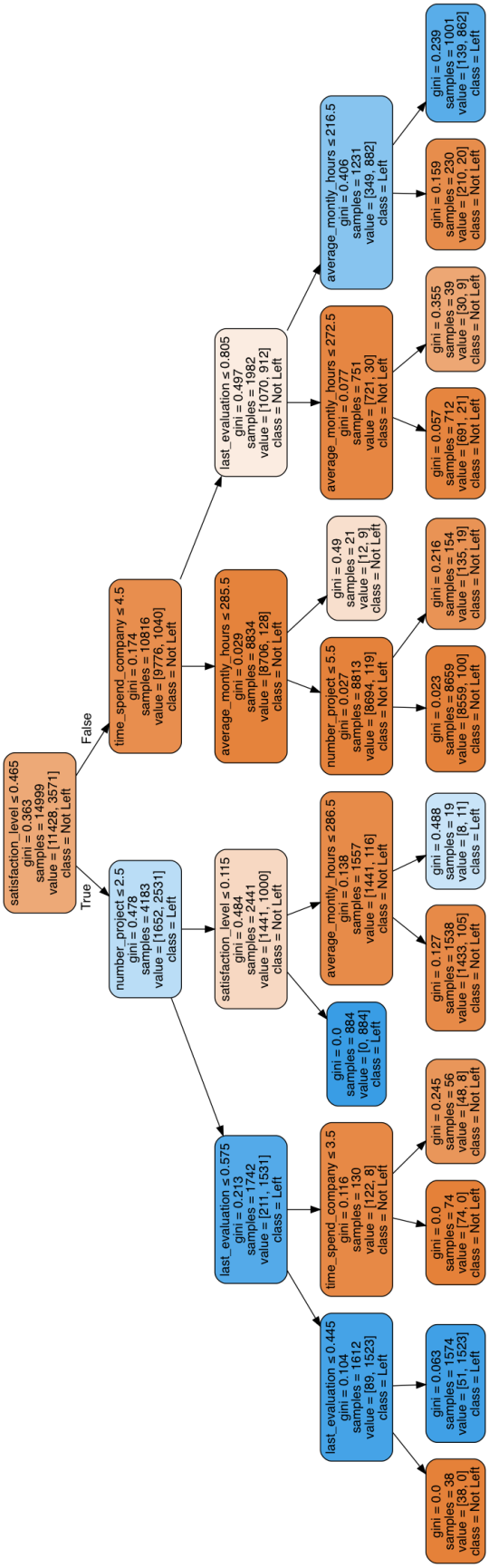
The ROC (Receiver Operating Characteristic) curve of the Decision Tree considered in the previous subsection is provided here (Fig. 4.2). The closer to the top left corner of the plot the curve is, the higher the AUC (Area Under Curve) is, and so are the general performances. The AUC of this particular model is equal to 0.95, which is quite high, since the maximum possible value is 1 in the case of a perfect classifier. Very similar AUC values are achieved by the other trained Decision Trees. The dotted line in the plot represents a "random guessing" classifier and its AUC is equal to 0.5.

With an AUC value of 0.966, the more accurate Decision Tree is the one trained by considering Gini coefficient and limiting the depth to 8. Even though the dataset is unbalanced, the tree is able to achieve very good results classifiying correclty even the minority class (Left = 1) (Fig. 4.3)

It is possibile to achieve satisfying performances even with a tree characterized by a relatively low depth level. It is worth observing that, even if marginally better results could be achieved by training Random Forests Classifiers, it is not advisable to do so, since it is a time-consuming and computationally intensive process. This is the case in which a simpler model is just as good, if not better overall, than a higher complexity one.

All in all, Decision Trees proved to be an effective classification technique with regard to the examined dataset.

Figure 4.1: Decision tree



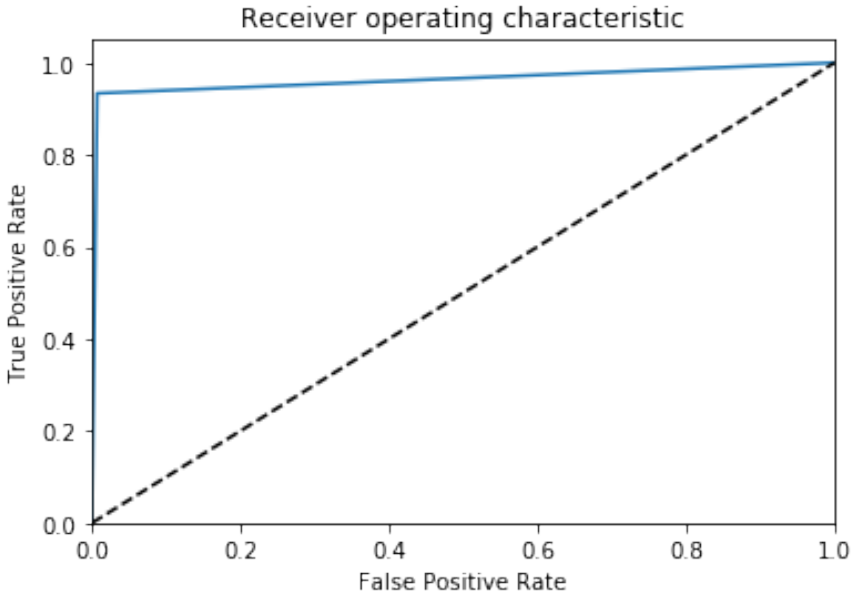


Figure 4.2: ROC curve of the DT (Gini, max depth 8)

	precision	recall	f1-score	support
Not Left	0.98	0.99	0.99	3462
Left	0.98	0.93	0.96	1038
avg / total	0.98	0.98	0.98	4500

Figure 4.3: Classifying performance of the most accurate DT