Data Mining: Foundations

Data analysis of the HR-IBM dataset

PROJECT REPORT

MARIA CASSESE, DARIA MIKHAYLOVA, SIRIO PAPA, BEATRICE ROSI

PROF. DINO PEDRESCHI
PROF. MIRCO NANNI
PROF. RICCARDO GUIDOTTI
TA DOTT. SALVATORE CITRARO

Università di Pisa. Corso di Laurea Magistrale in Informatica Umanistica.

a.a. 2020-2021

Contents

Introduction

Employee voluntary turnover (also known as attrition) is one of the key indicators for organization health and performance in the domain of HR management. The cost of employee turnover is substantial and may reach 90 to 200 percent of salary of single leaver, so organizations use a series of methods to track, measure and prevent turnover (Edwards, 2016). The main question for HR management is "Why a person has left?" and the following connected "Who will leave next?".

This project has a goal to investigate methods of descriptive statistics and exploratory data analysis for prediction of employee attrition based on previously available data. To conduct the research, we use modified for the didactic purposes IBM Watson – HR data set. We may consider this set appropriate for analysis as the available data shows the *separation rate*, common measure of employee flow (Edwards, 2016), is 16% and it is around the average value for turnover in US in 2016 (when the dataset was released), so we can hope that attrition is not just given by random fluctuations, that have nothing to do with business operations.

In HR professional literature the voluntary attrition is often divided between positive and negative. Positive turnover happens when a leaver receives a better offer from a competitor or more generally changes the employer for better condition. Negative attrition refers to leavers who go away from the employer due to specific or general unsatisfaction. Therefore, our effort for descriptive part is concentrated on finding attributes that may contribute to positive or negative decision of the employee to leave.

Data understanding

1. Data semantics

For data understanding task we merged two provided files "training" and "test", so our final dataset contains 1470 observations for 33 attributes. The attributes of the dataset are constituted by eighteen categorical variables, of which ten are ordinal and fifteen numerical variables, of which seven are continuous and eight discrete. Pivot grouping showed that values of the attributes are syntactically accurate. However, we noticed some semantic inaccuracy. For instance, for the same data objects value *YearsAtCompany* is larger than *TotalWorkingYears* in 390 records and in 234 records the value of *TotalWorkingYears* is greater than the one of the *Age*. This problem was approached on the stage of data cleaning.

We may also consider the dataset unbalanced as it presents only small fraction of employees that left the company in respect to those who remained： ‘*Yes': 237, 'No': 1233* and biased for the task of predicting attrition, as we have observations only for those people that were hired by the company.

One of the main limitations of the dataset is the absence of measurements taken in different points in time: for instance, we know the last value for all *Rating* and *Satisfaction* variables but is not possible to trace the change in those values over time; the attrition could be connected not only with absolute value but rather with its change.

2. Distribution of the variables and statistics

Correlation and redundant variables

For initial analysis of variables, we plotted correlation matrix (Pearson correlation) across all continuous variables, it is shown in Figure 1.

The variables that best predict the respective value are visualized in the Figure 2.

The correlation analysis detects some redundant variables. For instance, *YearsInCurrentRole* and *YearsWithCurrManager have* linear correlation of 0.71 and a very similar distribution, both of them are strongly correlated with *YearsSinceLastPromotion* (0.55 and 0.51 respectively). All these variables are mildly correlated with *JobLevel.* So we may deduce that *JobLevel* in general is connected with job experience. Based on this analysis we exclude all but one correlated variable from datasets used for model building.

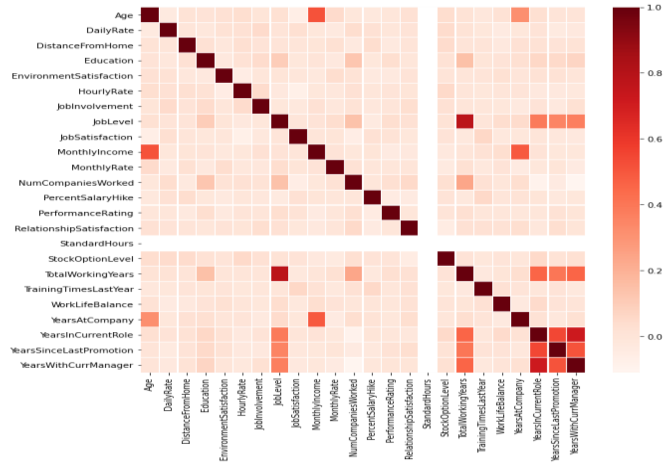


Figure 1. Correlation matrix (Pearson coefficient) for all variables of dataset.

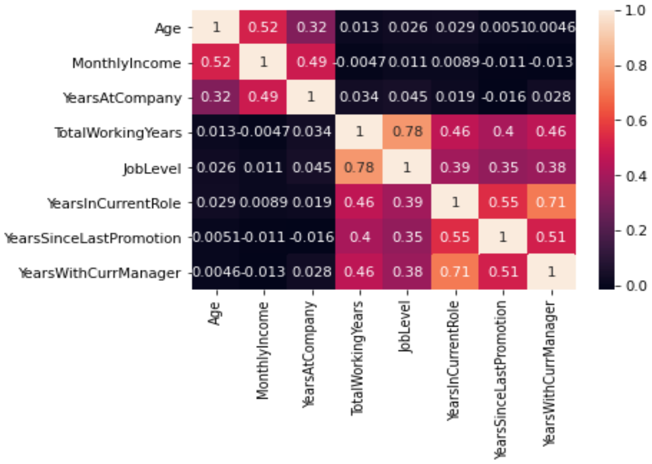


Figure 2. Correlation matrix (Pearson coefficient) for some variables of dataset

Categorical variables generally are not correlated, however *Department* and *JobRole* attributes represent the same information with different level of granularity, we use one of them for model building.

Redundant by their semantics variables are: *Over18* and *StandardHours,* they always take the same values, so can be eliminated from the dataset without a loss of information.

Transformed variables

We have transformed some variables to fill missing values and to perform statistical analysis. Categorical discrete variables were transformed in (0,1) values, categorical ordinal variables were substituted with integers with equal distance. All ratings were aggregated in two variables (see discussion below). *YearsAtCompany* variable was aggregated by 5 years to build a pivot table in order to replace missing values.

Accessing data quality

Missing values

The set has 2421 missing values in 9 attributes. Table 1 shows the distribution and Figure 3 shows the frequency of missing values.

N	Attribute	Number of missing values
1	Age	212
2	BusinessTravel	131
3	Gender	75
4	MonthlyIncome	280
5	Over18	468
6	PerformanceRating	172
7	StandardHours	717
8	TrainingTimesLastYear	292
9	YearsAtCompany	74
	Total	2421

Table 1. Missing values for variables

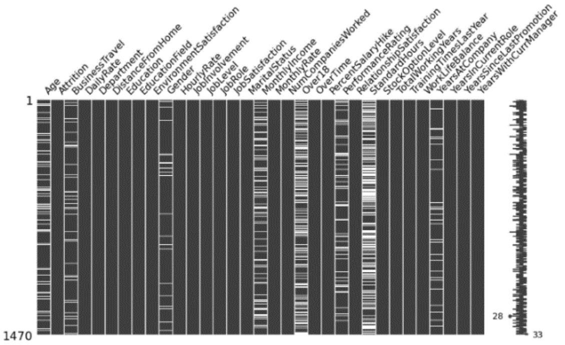


Figure 3. Frequency of missing values across the dataset

The simple elimination of missing values would produce a dataset of 207 observations, 86% of the observations would be lost, a rate that is not acceptable for further exploratory analysis, for this reason we adapt different techniques to substitute missing values, they are discussed further.

Age

The missing values calculation has been performed computing the difference between Age and *TotalWorkingYears* has been calculated to observe the starting age of all employees.

However, in several cases, *TotalWorkingYears* > *Age*. In order to prevent this data from misleading the calculation, we made a substitution in the Difference Series (*Age-TotalWorkingYears*) of those values that were <16. In essence, we considered wrong values those in which the age of starting work was less than 16 years.

Then, we calculated the rounded average of the starting age (30 years), and we added that value to the corresponding value of *TotalWorkingYears*.

BusinessTravel, Gender, PerformanceRating

The categorical attributes *BusinessTravel*, *Gender* and *PerformanceRating* are not significantly correlated with any other variable, so in all these cases we've randomly substituted the missing values with possible ones (*Travel_Rarely*, *Travel_Frequently* and *Non-travel* for *BusinessTravel* ; *Male* or *Female* for *Gender*, *3* and *4* for *PerformanceRating*) keeping their initial proportion in the dataset.

MonthlyIncome

The distribution of variable *MonthlyIncome* is skewed on the right, so there are more employees that have lower monthly income. Contrary to what we expected, the variable's mean and median do not change significantly with respect to Job Role or Department.

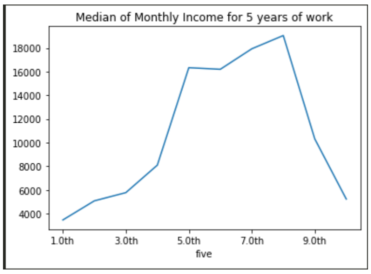


Figure 4. Median of monthly income calculated for each 5 years of experience

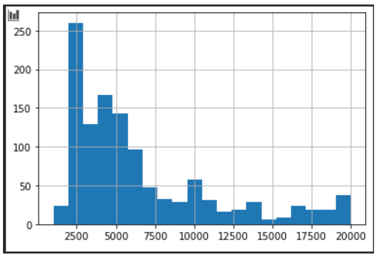


Figure 5. Initial distribution of MonthlyIncome

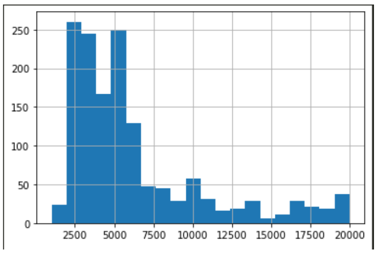


Figure 6. Distribution of monthly income after replacement of missing values

Instead, it has the strongest correlation with the variable *YearsAtCompany* (spearman rho - 0.45 before the filling of missing data). While there are some unexpected values, general tendency is that monthly income increases with seniority of employee. The distribution of median values of monthly income with respect to seniority is demonstrated in Figure 4. Therefore, we have replaced the missing values with median value of MonthlyIncome for each 5 years of working experience incrementally (bellow 5, 5-10 years of experience and so on). The distribution of the values (Figure 6) changed, as there are a lot more employees with short working experience.

TrainingTimesLastYear

This variable can assume just seven values (from 0 to 6). Also, in this case, the variable is not significantly correlated to any other and the distribution remains basically the same in any condition. Median and mean are quite close, and histogram shows that the values follow normal distribution. We've considered the mutual influence of values of *YearsAtCompany* and *JobRole* on TrainingTimesLastYear, so the missing values were replaced by the mean for groups created by intersection of JobRole and 5-year span of YearsAtCompany (for example HR - 5-10 years of experience, Technician – 10-15 years and so on). Replacing of the missing values did not influenced the total distribution of the variable.

YearsAtCompany

This variable doesn't have a normal distribution and isn't significantly correlated with any other variable, except *MonthlyIncome* which however have many missing values. Analyzing the mean and the median with respect to other variables (*JobRole*, *JobLevel*, *JobSatisfaction* and *Department*), we noticed that they don't change significantly. We decided to replace the missing values with *YearsInCurrentRole*.

4. Descriptive statistics

In this section we describe meaningful results of visual and statistical exploration of dataset. As a result of statistical analysis, we identified the variables that could be important for prediction of the attrition, they are described in the Table 2. The details are discussed below.

Categorical	
OverTime	If the employee works more than defined by the contract

StockOptionLevel	The possibility of the employee to purchase certain amount of stock for a fixed price
JobRole or Department	The employee's position, in some cases with department name attached
JobLevel	The position of the employee according to internal hierarchy
MaritalStatus	Whether the employee is married, not married or divorced
Gender	Whether the employee's gender is male or female
Education	Level of education (Below College, BA, MA, PhD)
Numerical	
MonthlyIncome	Average monthly income for some period.
TotalWorkingYears	The total years of employment
YearsAtCompany	How long an employee was working for the company
YearsSinceLastPromotion or YearsInCurrentRole	How many years passed from the moment the employee has been promoted or given salary hike How long the employee was working in current role (see JobRole）
DistanceFromHome	How far the employee's home is from the office
Age	The age of employee
Satisfaction	Aggregated rating given by employee
Performance	Aggregated rating given by manager

Table 2. Significant variables found with statistical analysis

The overall understanding of the factors influencing Attrition based on exploration of the dataset are summarized in the map on **Figure** 7.

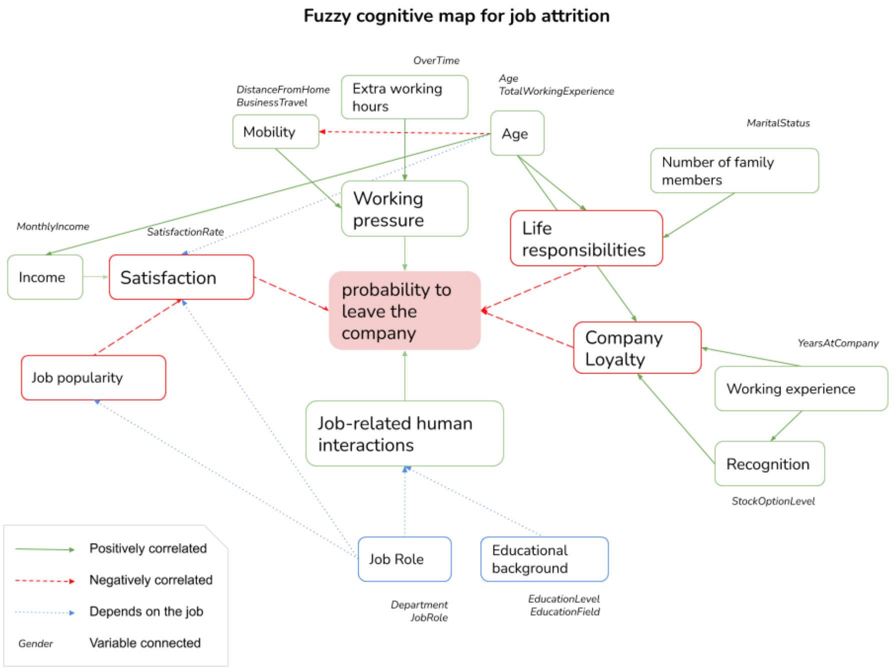


Figure **7**. Cognitive map for Attrition variable

Theil's U coefficient for categorical variables

To explore the influence of categorical attributes on attrition we computed Theil's uncertainty coefficient for the 9 categorical variables, the coefficient value for all variables is very small, but JobRole (0.07), OverTime (0,06) and StockOptionLevel (0.05) are highest among all; we can be sure that none of the variables alone may predict attrition of employee, however we will try to use the best variables in model building. In the following sections we discuss some interesting facts emerged from descriptive statistics of single variables, grouped by their typology.

Attrition vs Experience, JobRole/Department, JobLevel and Education

The **nature of job** and **position level** and working experience may affect attrition. First of all, attrition rate is higher for employees with **shorter working experience.** In the following normalized histogram plot (**Figure** 9), it can be seen that

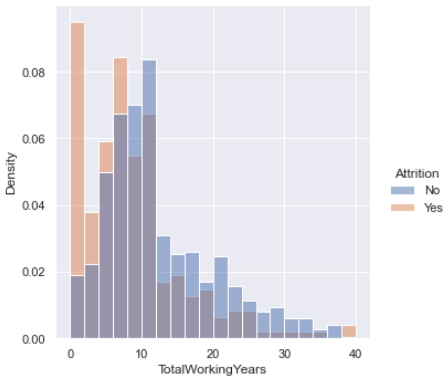


Figure 9. Normalized distribution of TotalWorkingYears

people who have fewer total working years (from 0 to 8 years old) have higher attrition rate as well as people with many working years who maybe retired.

We also noticed how it is possible to observe higher *Attrition 'Yes'* frequencies for some specific *JobRole* values. Specifically, we notice that by far the highest frequency is found in records where *JobRole* is ***Sales Representative*** (39.75%, when *Laboratory Technician* has only 23.93% probability). Doing in the same way an analysis on the *JobLevel*, it is found that employees with a low ***JobLevel* (=1)** have by far the highest probability of having *Attrition 'Yes'* compared to all the others (26.33%).

Having observed how these values of the variables influence the Attrition values, we tried to see how they act at the same time and it emerged that the probability of having *Attrition 'Yes'* for the staff with *JobRole 'Sales Representative'* and *JobLevel* 1 is well of 42.1%, when, considering the total staff, the probability is only 16.1%. In fact, employees with these characteristics make up 2% of the total but represent 13.5% of employees with *Attrition 'Yes'*. The dependence of the *JobRole* with *Attrition*, in employees with a low *JobLevel*, is confirmed by the Chi2 test which indicates a p value of 0.000945.

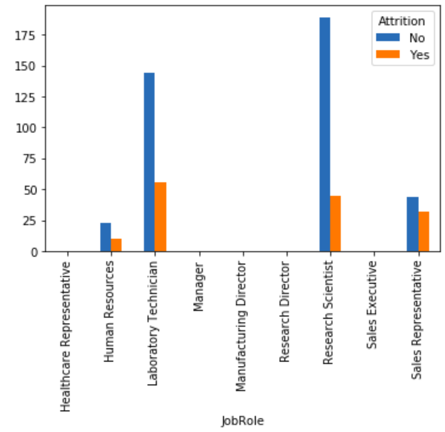


Figure 10. Number of employees of Attrition 'Yes' for Job Role and Job Level 1

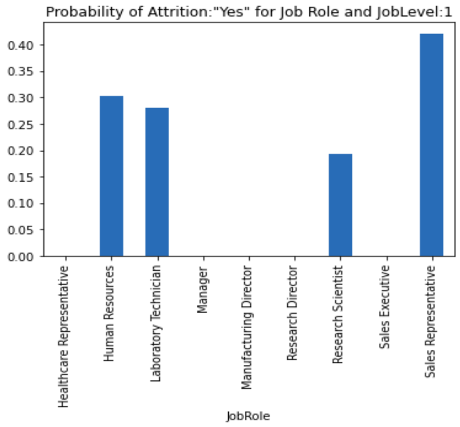


Figure 11. Probability of Attrition 'Yes' for Job Role and Job Level 1

The observed tendency is also confirmed by EducationField, the chi-square statistics for *Attrition* vs *EducationField* confirms that we may reject hypothesis that attrition is independent of field of education (16.02 against critical value of 9.48). In fact, the probability is higher for employees with Marketing degree (Sales Representatives), but it is also higher for those with Technical degree (Laboratory Technicians/Research Scientist) and slightly so for those with HR.

We've also noticed that not only position by itself affect attrition probability, but the change of it. While the variable *YearsInCurrentRole* is not correlated with *YearsAtCompany*, as shown in linear regression plot on Figure 14 (people of every level of experience got moved or promoted), there is high attrition among those who just change job role (0, 1 and 2 years) and (considering the correlation with *YearsWithCurrentManager* variable) the manager (Figure 15).

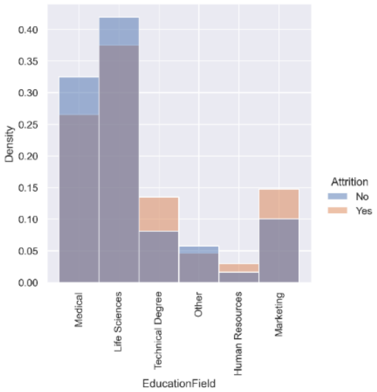


Figure 12. Normalized distribution of values for Education Field with respect to Attrition

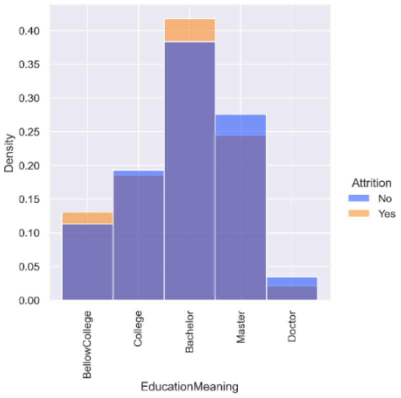


Figure 13. Normalized distribution of values for Education Level with respect to Attrition

The second pick is on 7 years and probably depicts those leavers who contrarily didn’t got promotion. We may conclude that employees are at risk of leaving company during the very first years in new role.

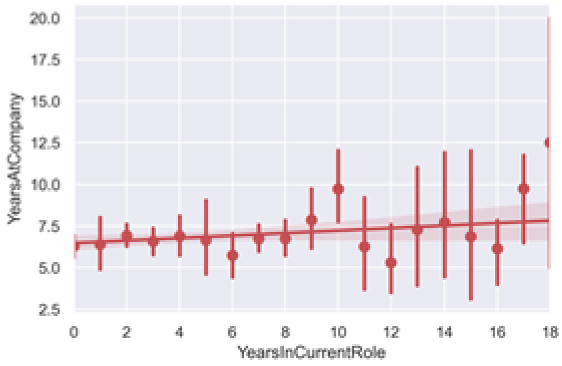


Figure 14. Linear regression plot showing no significant correlation between years in company and in current role

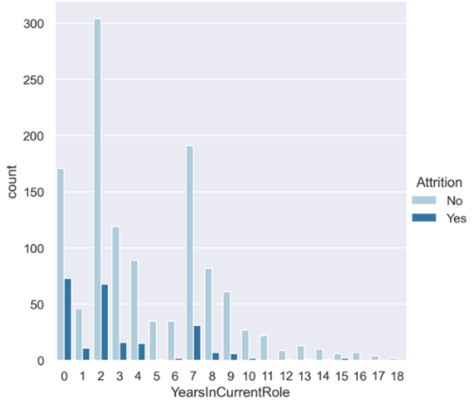


Figure 15. Distribution for value of YearsInCurrentPosition

The level of education may also affect attrition rate, for instance the plot of normalized distribution of ‘leavers’ and ‘stayers’ across levels of education (Figure 13) suggests that employees with **bachelor's degree** and **without tertiary education** leave more frequently, however Cramer’s coefficient does not show significant correlation and chi-square test permits us to say that distribution is equal.

Attrition vs Personal treats and conditions

There are several variables that describe personal characteristics of employees and their condition like Age, Gender, MaritalStatus, DistanceFromHome and OverTime, we may say that the first three describe directly or indirectly responsibilities and the remaining two working pressure.

The distribution of the **age** of employees with higher rate of attrition is approximately the same of those with a lower rate (the median is respectively 36.0 and 37.0). From the regression analysis of *Age* and *Attrition* we know that the attrition coefficient is not statistically significant.

However, the visual inspection of the chart that distinguishes the two groups based on the kernel density (KDE) shows that people dropping out most frequently tend to be in the **prime of their working lives** (in the age group between 30 and 40), while people approaching retirement less often leave their employment. The two-sample Kolmogorov-Smirnov test confirms the hypothesis that the difference between the two empirical distributions is statistically significant (t statistic:1.0 > critical value: 0.1136).

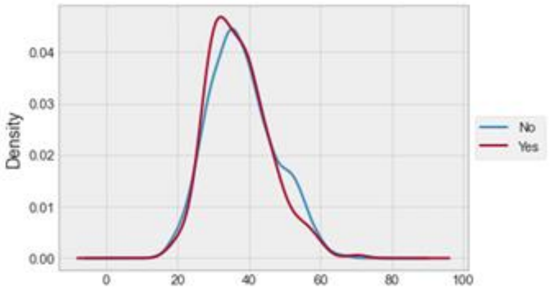


Figure 16. Kernel Density plot of Age

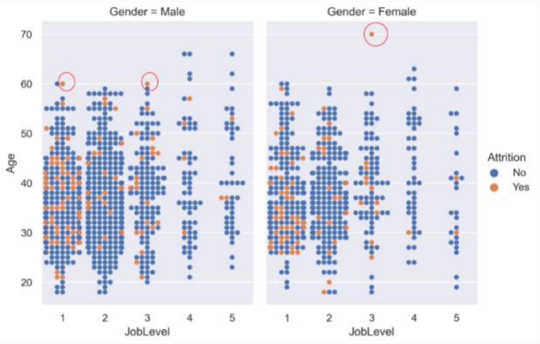


Figure 17. Scatterplot of Age values with respect to JobLevel for Male and Female employees.

The attrition rate among **men** and **women** is similar, however women generally start to work later, reach higher positions later, quit the job at a lower age in comparison with men and retire earlier. In fact, the median value for the two groups is respectively 37 for men and 34 for women. The two-sample t-test statistic confirm this intuition: the null hypothesis, according to which the age of people quitting their job is independent from gender, can be rejected (t-test: 1.339, p-value: 0.180). The scatter plot on Figure 17 shows also that young women (<30) on job levels 1 and 2 have higher level of attrition. Instead, women who reach level 3 and 4 tend to keep their job position independently of age. However, we don’t consider this analysis to be particularly accurate, since there were a lot of missing values for both *Age* and *Gender*.

Most of employees quitting the job are **single** (with a probability of 0.50), while married employees quit their job less frequently. Moreover, **single women** more often drop out (as shown on Figure 18).

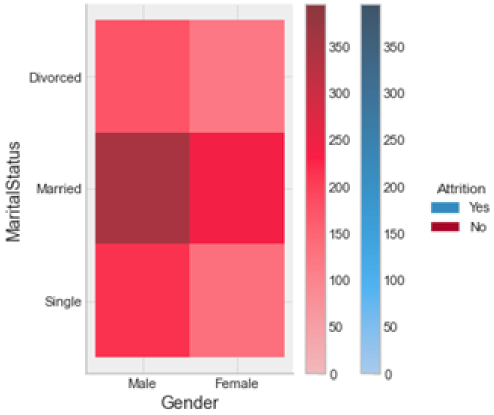


Figure 18. Attrition rate of Gender and Marital Status

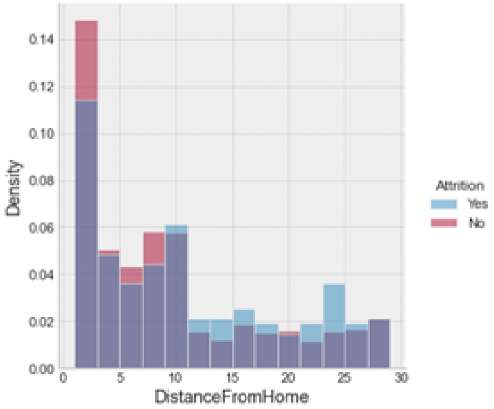


Figure 19. Density histogram plot of DistanceFromHome

The employees who **live more near** to their job place have lower attrition (Figure 19), in fact, the medium of *DistanceFromHome* is slightly higher, 9.0 for *Attrition=Yes* and 7.0 with *for Attrition=No.* From the regression analysis of *DistanceFromHome* and *Attrition* we know that the attrition coefficient (p value =0.003) is statistically significant, so the relation between the two variables is significant, maintaining a constant value for the other variables. R squared is too low (0,006): this means that the variance of the dependent variable (attrition) explained by the R-squared, is minimal.

The employees, who **work extra hours**, leave more often (Figure 20 and Table 3) and those who work overtime, and travel rarely, are the most represented group between leavers.

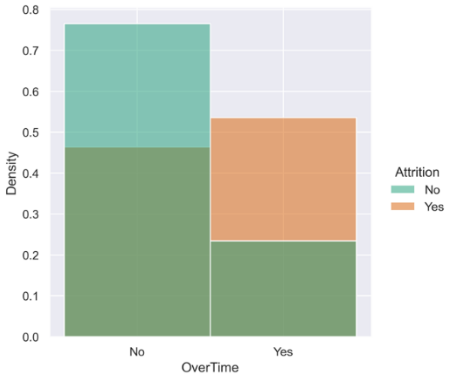


Figure 20. Normalized distribution of leavers and stayers for OverTime variable

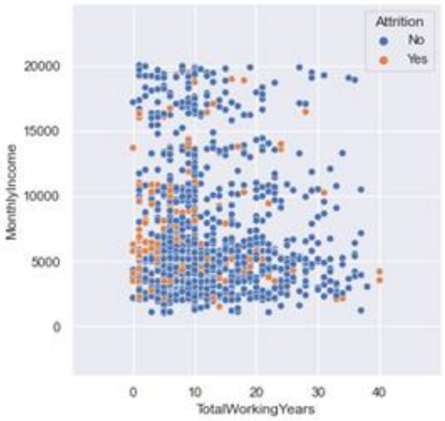
	OverTime		Standard	
	Abs	Rel	Abs	Rel
Leavers	127	54%	110	46%
Stayers	289	23%	944	77%

Table 3. Frequency distribution (relative and absolute) for OverTime for employees who left and whose who remained.

Attrition vs Employee compensations

The dataset provides us with some attributes describing financial compensation (*MonthlyIncome, Rates*) and other incentives (*StockOptionLevel*) that affect attrition rate.

Analyzing the scatterplot (Figure 21) across three dimensions (*Attrition*, *TotalWorkingYears* and *MonthlyIncome*), the two points of employees with *Attrition=Yes* and *TotalWorkingYears* '40' are more evident, but we can also notice there is consistent group of 'leavers' with values of *MontlhyIncome* between 3.500 and 10.000 and with *TotalWorkingYears* up to 5 years. Ecdf shows that 0.8 of all employees earn less than 8000 and after this number the probability of Attrition is higher.



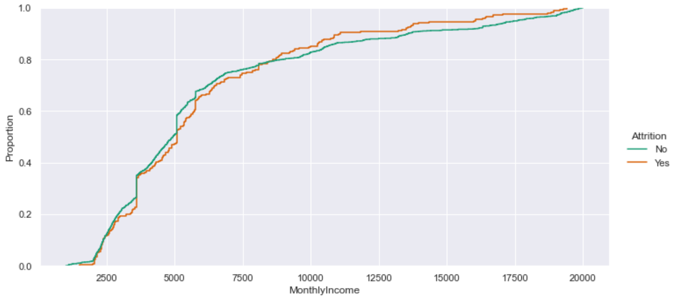


Figure 22. Empirical cumulative distribution for MonthlyIncome

Figure 21. Scatterplot WorkingYears wirh respect to MonthlyIncome

By comparing the three *DailyRate*, *HourlyRate* and *MonthlyRate* variables with YearsAtCompany and *TotalWorkingYears*, we can see that the employees with *Attrition=No* are more concentrated in the lower part of the jointplot with the YearsAtCompany variable on the y-axis. On the other side, the 'Yes' values of Attrition seem more grouped between 0 and 5 years of TotalWorkingYears. For example, we can notice a group of employees with limited experience and MonthlyRate between 2500 and 3000, unlike YearsAtCompany (Figure 21).

The Figure 23 also demonstrates that the people leaving with more work experience don't correspond to those with more years of work in the company. In fact, even though there are not many employees with more than 25 years in the company, all of them have *Attrition 'No'.*

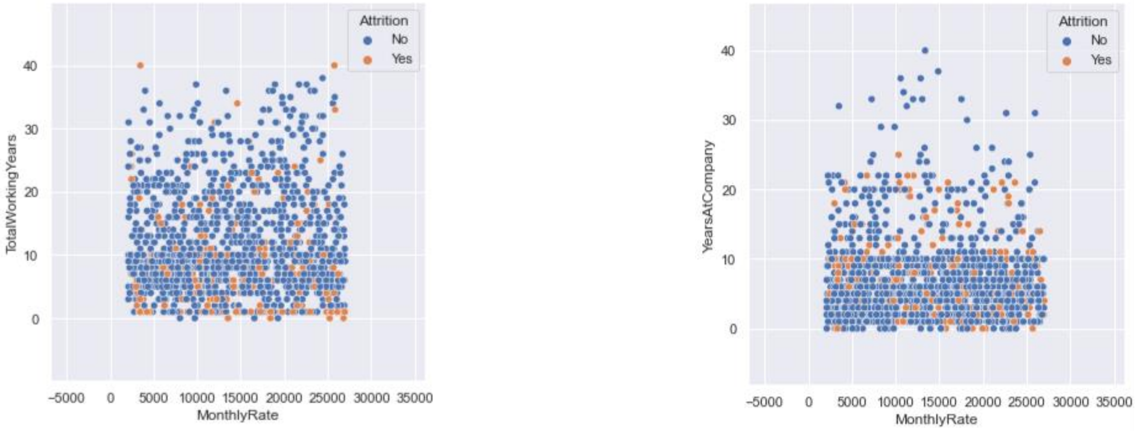


Figure 23. Scatterplot of MonthlyRate respect to TotalWorkingYears and YearsAtCompany

StockOptionLevel shows the possibility of employee to participate in the company or buy shares at special price. The **Figure** 24 shows clearly that the proportion of "leavers" is greater among employees who has no stock option level with respect of whose who has some, while two sets are almost equal by size (57% with stock option and 43% without)

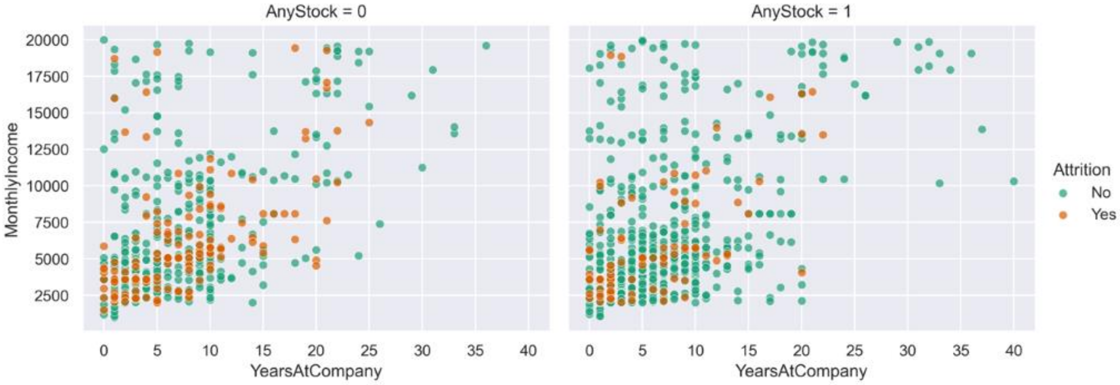


Figure **24**. Scatter plot for monthly income against working experience for employees with and without stock option level.

Moreover, we noticed that the correlation between stock option level and marital status exists and it is statistically significant (Spearman Rho: -0.74). This correlation is probably due to the fact that incentive compensations are granted to the employees at certain moment in their career that coincide with a marriage.

Attrition: Employee evaluation and satisfaction

The values of level of satisfaction of the employees (*EnvironmentSatisfaction, RelationshipSatisfaction, WorkLifeBalance* and *JobSatisfaction*) and the performance feedback of the manager (*PerformanceRating* and *JobInvolvement*) in the dataset are high. We've discovered that the attrition rate doesn't depend on **performance rating**, but most of the people quitting the job belong to the category of employees with a **low level** of working life balance, environment satisfaction, job involvement and job satisfaction and the dependence of the attrition on each of these variables is confirmed by the chi square statistics. To have more general picture of influence of rating variables on attrition we computed aggregated rating of two types (satisfaction and feedback). The first rating has minimum value of 4 and maximum value of 16. While distribution by itself is not meaningful, we can see that employees with rating less than 11 tend to leave company a lot more.

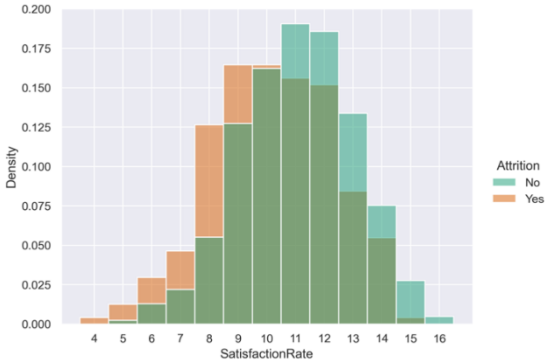


Figure 25. Normalized distribution of leavers and stayers for Employee satisfaction aggregated rate.

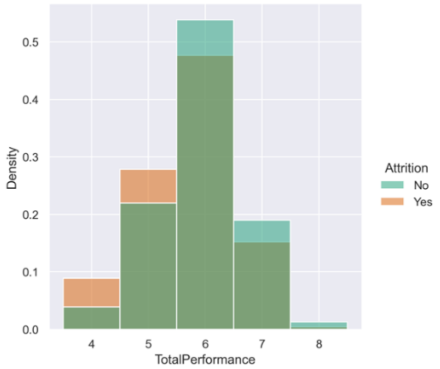


Figure 26. Normalized distribution of leavers and stayers for Employee performance aggregated rate.

The second rating has min. value of 4 (there are no records with rating 1 for this type) and max. value 8. Again, the distribution shows that employees with low rating have higher attrition rate. However, we do not take this observation too seriously as there were a lot of missing values for *PerformanceRating*.

Clustering

Statistical analysis suggested us that in our dataset there are at least two types of employees that leave the company: young professionals after 1-3 years of work, mostly in Sales department and mature specialists with high job level and salary among Research and Development, there are also people that retire. We also know that numerical continuous variables (required by clustering algorithms) are very few in the dataset and only 2 have somewhat significant correlation with attrition. In fact, clustering algorithms detect clusters in the dataset, their characteristics are distinct, but as for our understanding none of the methods predict the attrition well (the best result is the cluster with 50% of attrition), the clusters are noisy. Clustering captures either large subset with even lower attrition rate or very small cluster with higher rate, but it describes very small part of attrition cases. From our results we may conclude that clustering is not very suitable for our data.

K-means

We first applied the k-means clustering algorithm to the subset of our dataset with only numerical continuous and ordinal data (total of 20 variables). The data was normalized to the range (0,1) with MinMaxScaler.

The best silhouette score for the data among [2, 20] clusters is for 2 clusters and it's only 0,1 (the SSE is 1544). For k=2 there were no clusters with a maximum silhouette score less than the average score. The predicted two clusters didn't give us any significant information about "leavers": cluster 0 had an attrition of 0.19 and cluster 1 - of 0.09. The bigger k didn't produce significant improvements.

We've added to the data only one discrete dimension *OverTime*, with higher Theil's U coefficient for Attrition; the resulting data set has the following attributes:

[*'Age', 'DistanceFromHome', 'Education','JobLevel', 'StockOptionLevel',TotalWorkingYears', 'YearsAtCompany', 'YearsSinceLastPromotion', 'MonthlyIncome','Satisfaction', 'Performance', 'Department', 'OverTime'*]

On the visualization with reduced dimensions (PCA) the data separated into two visible clusters. Elbow graph (Figure 27) suggested that the best value for k could be between 5 and 6. The average silhouette score for this clustering was also low (0.087).

However, one of the resulting five clusters has an attrition of 0.37, it captures 41% of all workers that left the company, so far, our best result with k-means. The Figure 29 shows the clusters plotted in two dimensions against working experience and monthly income.

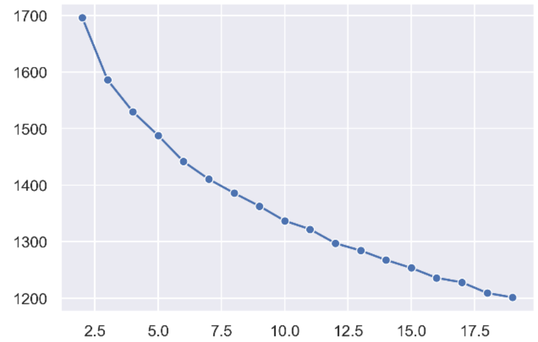


Figure 27. Elbow graph for clustering of data with OverTime

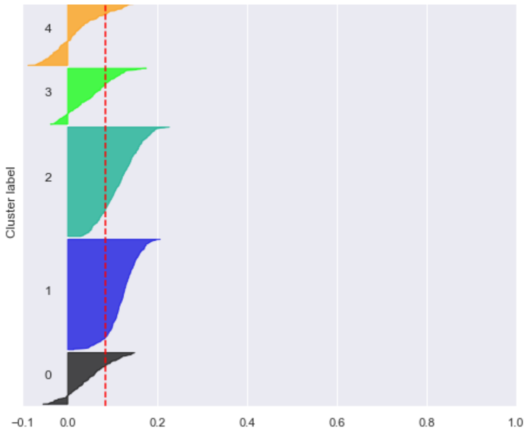
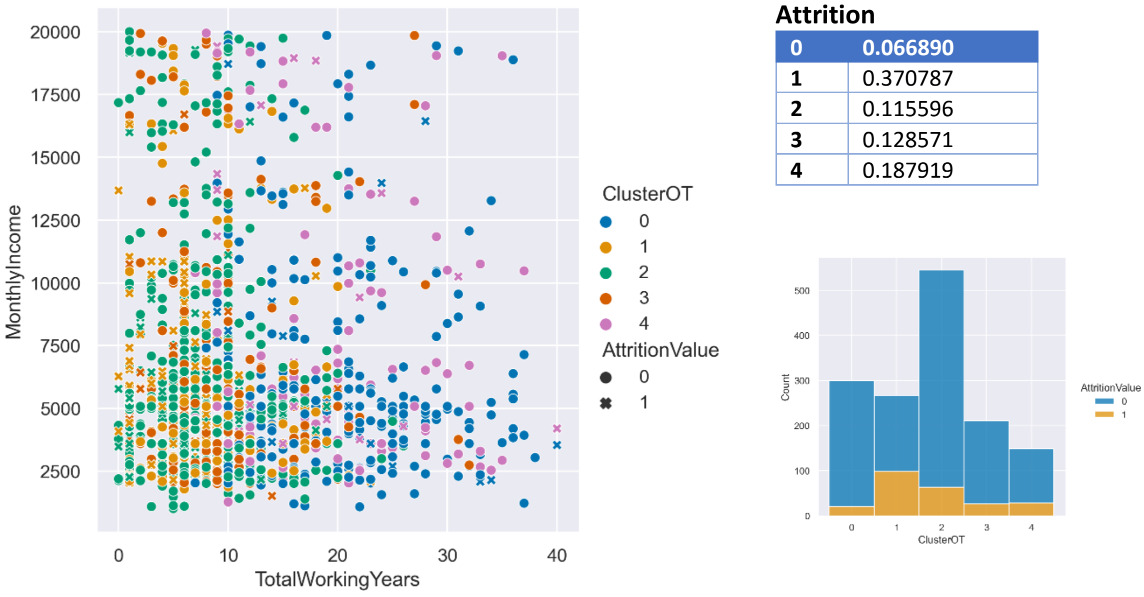


Figure 28. Silhouette analysis result for 5 clusters

Average age (35), income (5 926), *JobLevel* (1.5) and *YearsInCurrentPosition*(2.46) in cluster 1 are the lowest between all clusters and the whole dataset, *WorkingExperience* is second low. Cluster 0 captures the employees with highest working experience (19) and higher *JobLevel*.



Attrition

0	**0.066890**
1	0.370787
2	0.115596
3	0.128571
4	0.187919

Figure 29. Visualization of clusters in two dimensions and histogram for the attrition

Clustering on the whole dataset with one-hot encoded discrete and binary variables didn't produce significant results from point of view of prediction of the attrition value.

Hierarchical Clustering

For the choice of variables to be used in hierarchical clustering method we have adopted an experimental approach. We kept a fixed core of continuous variables with a good influence on the Attrition variable ([*Age, DistanceFromHome, TotalWorkingYears, YearsAtCompany, YearsInCurrentRole, Satisfaction, MonthlyIncome*]) and we added and replaced other features evaluating how they interfered in clustering. Eventually, the best combination we found consists in this list of features: [*Age, DistanceFromHome, TotalWorkingYears, YearsAtCompany, YearsInCurrentRole, Satisfaction, JobLevel, Education, OverTimeN, StockOptionLevel*]. The data was normalized to the range (0,1) with MinMaxScaler.

Regarding the distance function used we applied the four main methods and evaluated their performance: 'complete', 'single', 'average', 'ward'. In general, we observed that the 'complete' and 'ward' methods are the best performing ones to create clusters with a high percentage of *Attrition*, maintaining the most constant performance even during the testing phase of the different variables.

In order to evaluate the performance of the methods used, based on the variables chosen, we chose the longest branch cutting technique: we cut the dendrogram in the middle of the longest branch. In this way we obtain a number of clusters that theoretically should be more defined and refined.

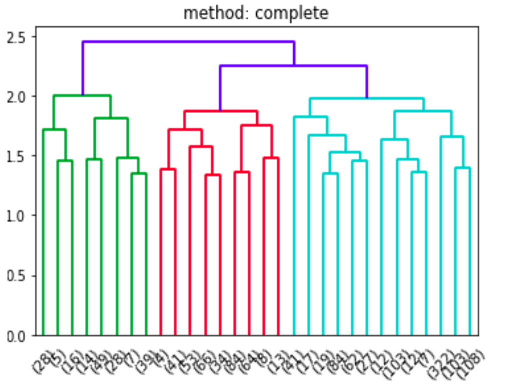


Figure **30**. Dendrogram with method: complete

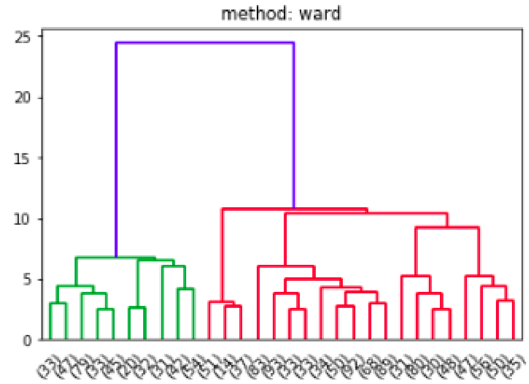


Figure **31**. Dendrogram with method: ward

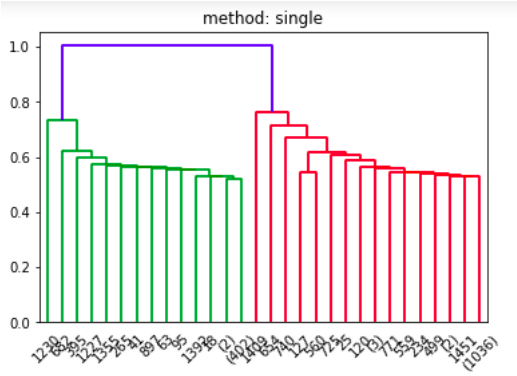


Figure **32**. Dendrogram with method: single

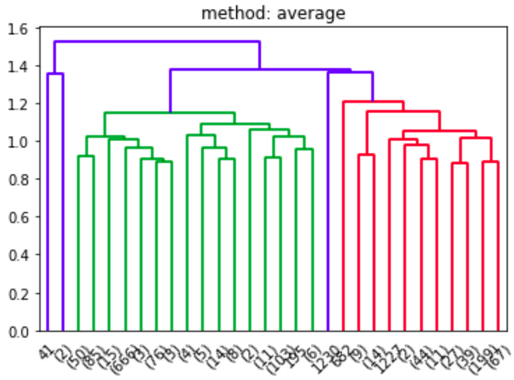


Figure **33**. Dendrogram with method: average

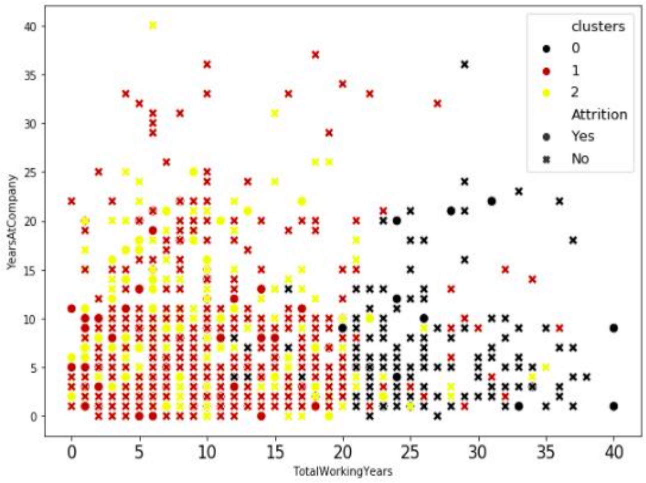


Figure 34. Visualization in two dimensions of clustering

Cluster	% Attr.	n Attr.	numerosity
0	0.080	15	186
1	0.111	102	917
2	0.326	120	367

In all four cases, clusters with an Attrition value above 30% are obtained, but the complete method obtained a percentage of 32.7% with a cluster with an acceptable number of elements.

This cluster (number 2, as it is shown in the table) capture 120 elements with positive *Attrition* and that is the 50.6% of the total positive *Attrition.* We think this result must be considered satisfactory.

Cluster number 2 captures the group of elements with the lowest average age (36.72), the lowest *Education* level (2.83) and the lowest *PerformanceRating* level (3.144), but with a higher *MonthlyIncome* than the other two clusters (6388).

DBSCAN

We applied the DBSCAN clustering algorithm to two subsets of the initial dataset, normalized with MinMaxScaler. The first subset is composed of 12 variables selected on the basis of the statistical analysis, including one discrete variable (Department) and other ordinal and continuous variables ([*Age*, *DistanceFromHome*, *Education*, *JobLevel*, *StockOptionLevel*, *TotalWorkingYears*, *MonthlyIncome]*) and the two aggregated variables Satisfaction and Performance. This choice is due to the fact that by selecting only continuous and ordinal variables our clusters didn't show a satisfactory outcome.

In order to choose the optimal value for the epsilon parameter (eps=0.66), we used the metric of Euclidean distance and we calculated the distance from each point to its closest neighbor using the *NearestNeighbors* method. We calculated the minimum number of neighbors a core point should have (min_samples=16) with the Silhouette score applied to the normalized data points.

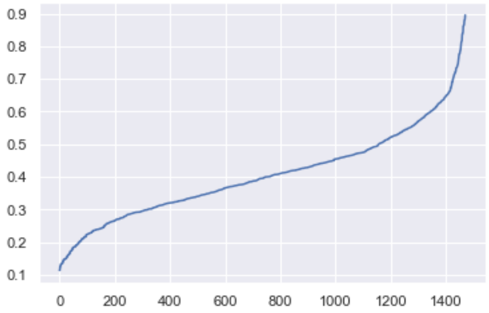


Figure 35. Nearest Neighbors graph to determine the optimal value for epsilon

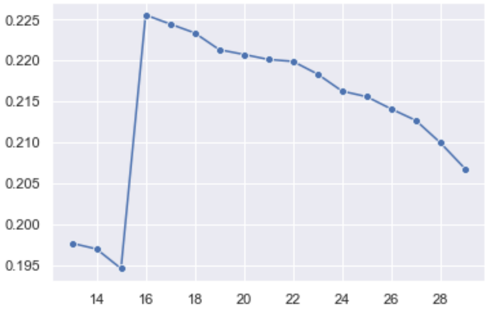


Figure 36. Silhouette graph results

We obtained two clusters, one of which (cluster 1) has got an attrition of 0.32 and captures the 44% of employees quitting the company (Table in Fig. 47). The cluster 1 captures employees with lower average *Age* (36), *MonthlyIncome* (5762.44) and *TotalWorkingYears* (9) and which show overtime job conditions. In the following scatterplot we can observe that the cluster 1 presents some groups of employees who leave the company, considering their Age and their number of working years (for example, *Age*=25/30, *TotalWorkingYears* = 0/1).

We compared the result obtained in the previous case with that obtained, including another subset of 9 variables selected according to their correlation with the attrition rate: *OverTime, MaritalStatus, JobLevel, StockOptionLevel, YearsInCurrentRole, TotalWorkingYears, MonthlyIncome, NumCompaniesWorked, YearsAtCompany.* We used the same method adopted for the previous dataset to establish the parameters and this time the optimal eps is 0.42 with 16 min_samples. We obtained six clusters and the cluster 1 and 4 captured nearly the same number of values (53 vs 57), but only the second one has an attrition of 0.53, which presents 24% of all employees with *Attrition 'Yes'* (Fig. 48). In the following scatterplot there are the same values of Fig 47 (*Age* and *TotalWorkingYears* by Attrition) and we can notice that cluster 4 in *'Attrition = 1'* (means *Attrition 'Yes'*) captures approximately the same points as the previous cluster 1.

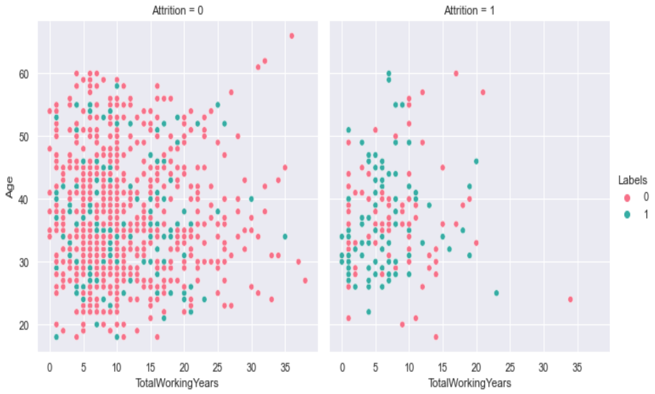


Figure 37. Scatterplot of the values of Age and TotalWorkingYears in the two clusters by attrition rate.

	Attrition	
Cluster	Abs	Rel
0	99	0.10237
1	53	0.32000

Table 4. Table of absolute and relative value of attrition for each cluster

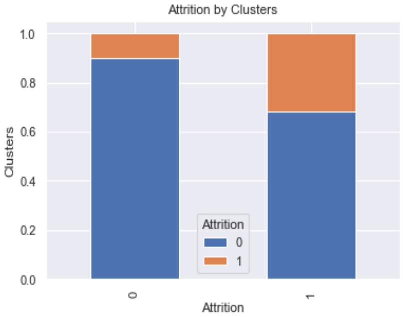


Figure 38. Barplot of the attrition in the two clusters

In the cluster 4 we can also see that the attrition rate is higher between employees who don't have any salary incentives (*StockOptionLevel* = 0), are young and single (average *Age* = 35), have been working overtime and for a few years (average *TotalWorkingYears* = 8), don't have an important role in the company (*JobLevel* =1, 2) and *YearsInCurrentRole* (2.62).

	Attrition	
Cluster	Abs	Rel
0	9	0.200000
1	53	0.168254
2	11	0.065868
3	34	0.083130
4	57	0.527778
5	29	0.237705

Table 5. Analysis of six clusters

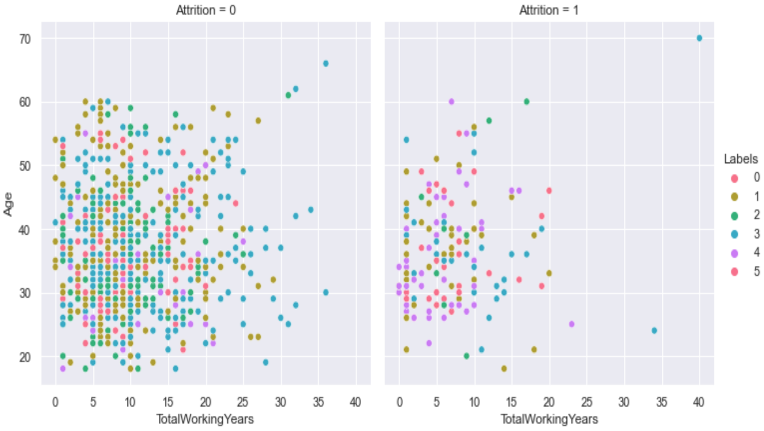


Figure 39. Scatterplot for the Attrition

Classification

Decision Tree

Decision models classifier can compute only numeric values. Therefore, the categorical variables were turned into numerical ones (*BusinessTravel, Department, JobRole, MaritalStatus, EducationField, OverTime*), excluding the target variable. Moreover, we genereted the variable *Satisfaction* as an aggregation variable as we did during clustering, to reduce the dimensionality of our dataset.

All variables were used for classification, using Attrition as the target variable. Experiments have also been carried out on the use of other subsets of variables, trying also various types of associations, but, finally, the use of all variables seems to be the most convenient way to produce nodes with lower gini and entropy values. Furthermore, we tried to remove attributes that were strongly correlated with each other to see if the classifier was affected (YearsSinceLastPromotion, YearsWithCurrentManager, YearsInCurrentRole), but this did not happen and is also confirmed by the feature importance of the classifier which assigns very low values to these variables.

Regarding, the construction of the decision tree, we started by searching for best parameters using the RandomizedSearchCV, looking for a model with a range of depths between 2 and 11 nodes, while min_sample_split was tested in a range between 30 and 100 records, and min_sample_leaf in a range between 20 and 80 records and the criterion of the classifier was set as 'gini'. The report ranked as first this model Parameters: {'min_samples_split': 30,'min_samples_leaf': 38, 'max_depth': 8} with a Mean validation score: 0.739 (std: 0.034). Trying the classification with the parameters from the report on our test set the results were not satisfactory. The tree stops classifying the label 'Yes' just on the fourth node, producing only two leaves that classify a 'Yes' output. One leaf with a gini value of 0.488 (far too high), another leaf produced by a third level node predicts 'Yes' output with a gini value of 0.361 (still not optimal). Analysing the scores of this model on the test set, the accuracy is 0.859 and F1 score is 0.92 however the confusion matrix detects the very high number of false negatives in the model (35 FN out of 45 'Yes' samples). Clearly this model in very weak.

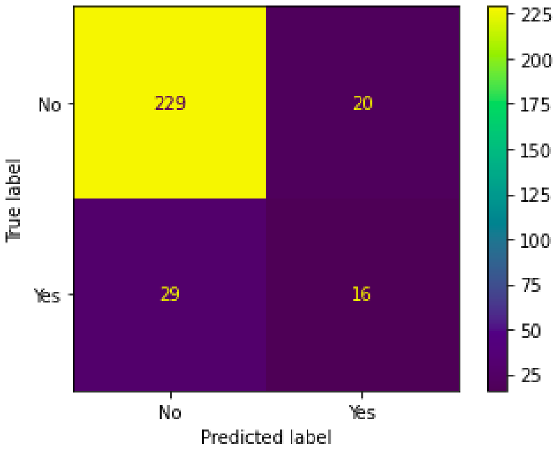
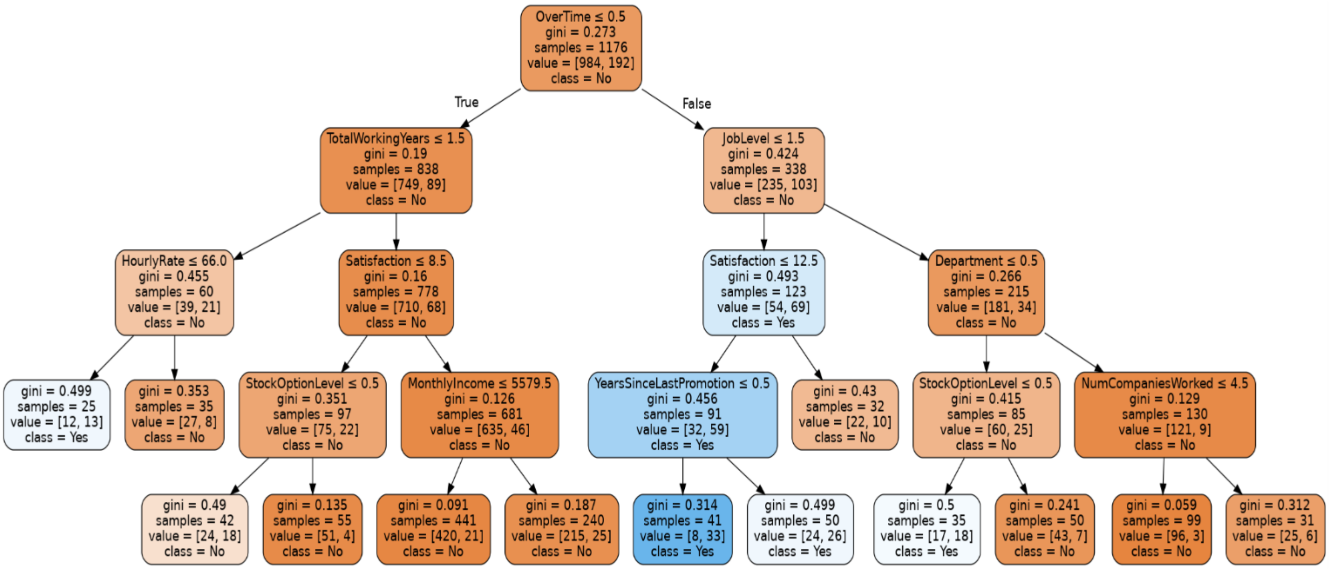


Figure 40. Test Set Conf. Matrix, best 'gini' model

We tried to tune the parameters, looking for better solutions that could mainly reduce the number of false negatives in the model. To do this, we observed that through an early termination of the splitting of the nodes it is possible to reduce the overfitting problems of the tree, without, however, resolving them completely. So we reduced the max depth of the tree from eight nodes to four. The best model obtained with the 'gini' criterion has these parameters: max_depth=4, min_samples_leaf=25, min_samples_split=30. This produces a matrix in the test set whose False Negatives number 29 and True Positives have 16 samples. The ROC curve confirms the deductions made so far: of all the models tested, the area never significantly exceeds the value obtained by this model (0.64), which, however, remains unsatisfactory.



OverTime ≤ 0.5 gini = 0.273 samples = 1176 value = [984, 192] class = No
True
False
TotalWorkingYears ≤ 1.5 gini = 0.19 samples = 838 value = [749, 89] class = No
JobLevel ≤ 1.5 gini = 0.424 samples = 338 value = [235, 103] class = No
HourlyRate ≤ 66.0 gini = 0.455 samples = 60 value = [39, 21] class = No
Satisfaction ≤ 8.5 gini = 0.16 samples = 778 value = [710, 68] class = No
Satisfaction ≤ 12.5 gini = 0.493 samples = 123 value = [54, 69] class = Yes
Department ≤ 0.5 gini = 0.266 samples = 215 value = [181, 34] class = No
gini = 0.499 samples = 25 value = [12, 13] class = Yes
gini = 0.353 samples = 35 value = [27, 8] class = No
StockOptionLevel ≤ 0.5 gini = 0.351 samples = 97 value = [75, 22] class = No
MonthlyIncome ≤ 5579.5 gini = 0.126 samples = 681 value = [635, 46] class = No
YearsSinceLastPromotion ≤ 0.5 gini = 0.456 samples = 91 value = [32, 59] class = Yes
gini = 0.43 samples = 32 value = [22, 10] class = No
StockOptionLevel ≤ 0.5 gini = 0.415 samples = 85 value = [60, 25] class = No
NumCompaniesWorked ≤ 4.5 gini = 0.129 samples = 130 value = [121, 9] class = No
gini = 0.49 samples = 42 value = [24, 18] class = No
gini = 0.135 samples = 55 value = [51, 4] class = No
gini = 0.091 samples = 441 value = [420, 21] class = No
gini = 0.187 samples = 240 value = [215, 25] class = No
gini = 0.314 samples = 41 value = [8, 33] class = Yes
gini = 0.499 samples = 50 value = [24, 26] class = Yes
gini = 0.5 samples = 35 value = [17, 18] class = Yes
gini = 0.241 samples = 50 value = [43, 7] class = No
gini = 0.059 samples = 99 value = [96, 3] class = No
gini = 0.312 samples = 31 value = [25, 6] class = No

Figure 41. Decision Tree Visualization, best 'gini' model

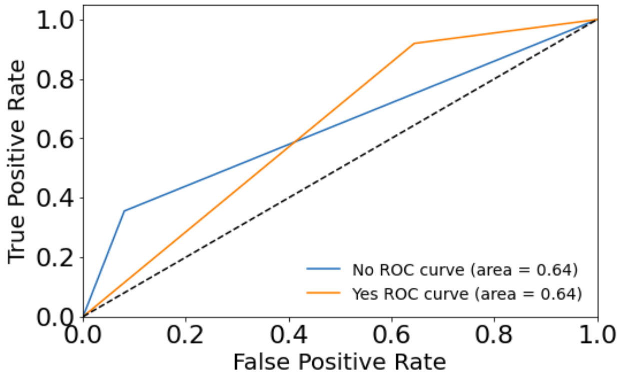


Figure 42. ROC Curve, best 'gini' model

Despite of the weak classification ability for the label 'Yes', the model has a good classification capacity for the label 'No', a fact which is further confirmed by the Cross Validation which assigns rather high Accuracy and F1_score (Accuracy: 0.8347 (+/- 0.03) F1-score: 0.6073 (+/- 0.08)).

The considerations about the effectiveness of the classifier remain similar even when changing the model criterion from 'gini' to 'entropy' the final considerations remain similar. The best model has the following parameters: max_depth=5, min_samples_leaf=30, min_samples_split=35 and the results in terms of confusion matrix are the same as the best classifier with 'gini' criterion. The only difference is one less TN and one more FP. Consequently, the ROC curve produced by this classifier also has an area of 0.64, while the Cross Validation assigns the following scores: Accuracy: 0.8361 (+/- 0.03) F1-score: 0.6081 (+/- 0.09).

Eventually, the analysis shows that the Decision Tree classification technique for this dataset is not effective for the prediction of the label 'Yes' in the target variable 'Attrition'.

KNN Classification

Finally, we analyzed neighbors-based classification, searching the optimal number of neighbors and the best weights. Also, in this case, the removing of attributes that were highly correlated with each other (*YearsSinceLastPromotion, YearsInCurrentRole*) didn't improve the classification. After converting all categorical variables to numeric ones and normalizing the dataset (minus the target column *Attrition*), we created and trained the KNN model with the default parameters (n_neighbors=5 and weights= 'uniform'). The results of confusion matrix of test set were good for Attrition_No, but not for "Yes": precision (No=0.86, Yes=0.50), recall (No=0.98, Yes=0.11), f1-score (No=0.92, Yes=0.18). In fact, also the ROC AUC had a value very close to 0.50 (roc_auc=0.53). So, we tried to evaluate alternative n_neighbors for better predictions. The Figure 43 shows two graphs about the accuracy and the error rate, using the average where our predictions were not equal to the test values.

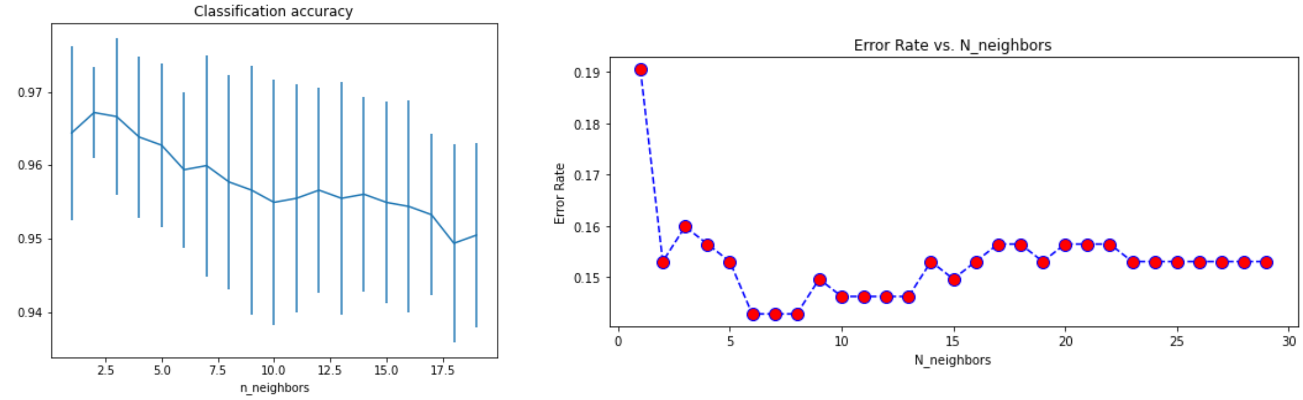


Figure 43 Accuracy and error rate of n_neighbors

We noticed that the accuracy decreased, and the error rate increased from n_neighbors=10, while values 6,7,8 should have been optimal. Comparing the confusion matrix, the accuracy and the Roc curve, we saw that the values didn't change much. The precision of Attrition_Yes of test set were perfect (precision=1.00) with n_neighbors=6 and 8, but the recall, f1-score and Roc curve slightly decreased respect to n_neighobors=7. In this last case, the value of Roc_auc was the best (0.56) with accuracy 0.857. In the Figure 44 we can see the matrix of the test set whose False Negatives number 39 and True Positives have 6 samples. So we tried to change the weights to improve our neighbors-based classification.

Setting KNeighborsClassifier with n_neighbors=7 (the same as before) and weights= 'distance', we noticed that the confusion matrix, accuracy and Roc curve didn't change, in contrast to n_neighbor=6 or 8. In fact the precision is significantly reduced respect to weights= 'uniform' ('distance': n_neighbor=6 precision=0.55, n_neighbor=8 precision=0.62, vs 'uniform': n_neighbor=6/8 precision=1.00). Moreover, while the other values of confusion matrix (recall and f1-score) and the Roc curve slightly increased, the accuracy decreased: 0.850 with n_neighbor=6 and 0.853 in the other case. So the best neighbors-based classification obtained was with n_neighbor=7, which had the best Roc curve (0.56), accuracy (0.857), recall (0.13) and f1-score (0.22).

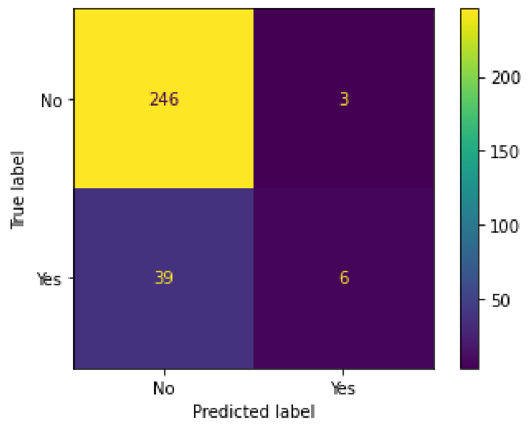


Figure 44. Test Set Conf. Matrix, best KNN model

In conclusion, if we observe the No of *Attrition* label*,* the KNN classification results better than the Decision Tree one, but for *Attrition_Yes* both the models aren't satisfactory, even if the Decision Tree classification obtains better results.

Association rules mining

Association rules mining gave us some interesting insights in our dataset, but to extract meaningful rules about "leavers" we had to use very low support and confidence value, therefore the computation was expensive. The prediction of target variable and substitution of missing values are quite precise, but the best rules cover small number of records and it would be interesting to explore the way to increase it.

Our dataset has mostly multi-dimensional attributes and continuous variables, therefore we performed some additional data manipulation. 6 categorical and ordinal attributes were left intact *(Attrition, BusinessTravel, EducationField, Gender, JobLevel, Department)*, for 5 categorical attributes we reduced granularity by aggregation:

- *Education* to *Secondary*, *Tertiary* and *Phd*
- *MaritalStatus* to *Alone* and *inCouple*
- *StockOptionLevel* to *StockOptionYes* and *StockOptionNo*
- *Performance* aggregated rating to <=5 and >5
- *Satisfaction* aggregated variable to *<=10* and *>10*

5 continuous attributes were discretized by a similar number of bins.

- *Age* to 3 groups: 18-35, 36-50 and 51-70
- *TotalWorkingYears* to 4 groups: less than 5, 6-10, 11-20, 21-40
- *YearsAtCompany* to 4 groups: less than 2, 3-5, 6-10 and 11-20
- *MonthlyIncome* to 3 distinct groups <3500, 3500-8500, <8500
- *YearsSinceLastPromotion* to 2 groups: <2 and >2

We performed discretization based on number of observations in each group and on statistical data analysis, for instance the threshold of 8500 for *MonthlyIncome* were chosen based on cumulative probability function, 80% of employees in the company earn less than 8500 and the probability of attrition beyond this threshold is slightly higher.

All attributes were then mapped to boolean values; the resulting item base has 53 items.

The quantity of frequent sets and their maximum length depends on the minimal support value, that is the parameter taken by apriori algorithm used for pattern extraction. Lower the value of minimum support, higher the number of sets generated and larger the size of the longest set: these relations are shown in Figure 45 and Figure 46, the number of sets is in log for clarity of representation. For instance, for min_support of 0.45 we will obtain 56 item sets with a maximum length of 3. By lowering the value of minimum support the number of frequent item sets and their maximum length increases. As a matter of fact, with min_support 0.4 we obtain 98 item sets with maximum length of 4 and with 0.05 for the minimum support, we obtain 61799 item sets and a maximum length of 9.

In the case of our dataset in most cases the number of closed and frequent item sets remains unchanged, except for a very low value of minimum support (with min_sup =0.05 frequent item sets = 61799, closed item sets=54394).

The number of maximal sets is generally lower than the number of frequent sets, but we can observe some variation accordingly with the change of the minimum support value. As a matter of fact, the difference between the number of frequent and maximal sets tends to decrease with the increase of minimum support value. For example, with minimum support of 0.2, the number of maximal sets is significantly lower (536) than the number of frequent item sets (1320). Instead, we have the same number of maximal and frequent sets with min_support of 0.75.

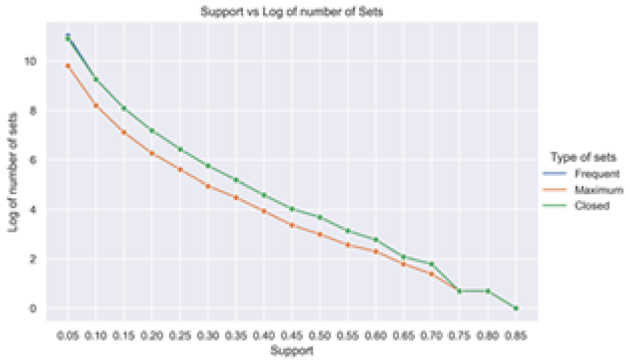


Figure 45. Support vs Log of number of item sets

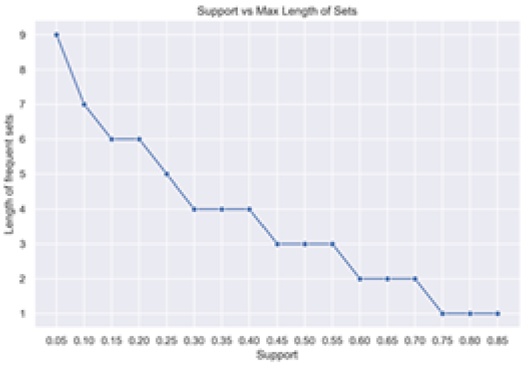


Figure 46. Support vs maximum length of item sets

Further we discuss the frequent sets obtained with min_support=0.4. The item sets formed by a unique item with a higher support value are *Education_Tertiary (0.85), Attrition_No (0.83), Performance>5 (0.72), OverTime_No(0.72).* By the results obtained from the selection of item sets consisting of two elements, we know that the following patterns co-occur with a higher frequency: (Attrition_No, Education_Tertiary; support = 0.71), (OverTime_No, Attrition_No; support = 0.64), (Attrition_No, Performance_>5; support = 0.62), (Attrition_No, DistanceFromHome_<=10; support =0.6). As it was expected the sets with high support value are those that combine very frequent attributes. The same is for the most frequent patterns consisting of 3 items. Finally, we get a single frequent set consisting of 4 items that identifies the co-occurrence of *Attrition_No, Performance> 5, Overtime_No, Education_Tertiary*, and its support is 0.40.

We can therefore assume that employees with low attrition rate constitute a group of people with intermediate education level and job stability, favourable working conditions and living at than of 10 km away from the office. Another interesting co-occurrence concerns job gratification: those who get job promotions coincide with those who have obtained a high-performance rate from their superiors *(YearsSinceLastPromotion_JustPromoted, Performance>=5; support = 0.46)*. In addition, most of those with intermediate education level also have an intermediate salary (MonthlyIncome_3500_8500, Education_Tertiary; support =0.465).

We've performed **rules generation from frequent sets** discussed above. The quantity of rules decreases with increase of confidence support and follow the distribution shown on Figure 47.

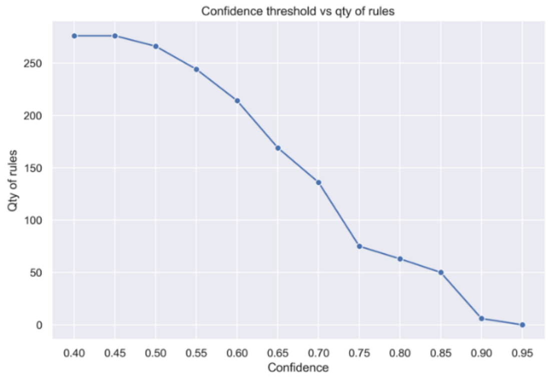


Figure 47. Quantity of rules for each confidence threshold

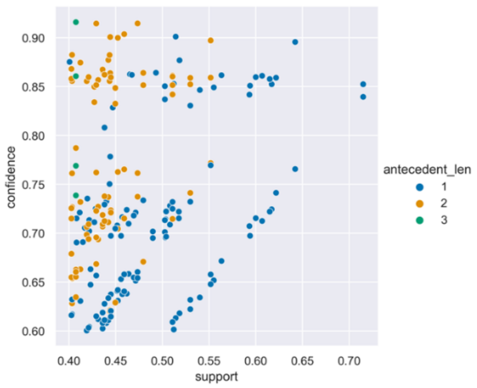


Figure 48. Scatter plot for antecedent length with respect to support and confidence thresholds

The four generated rules above 90% of confidence are:

(OverTime_No, TotalPerformance_6-8, Education_Tertiary) → *(Attrition_No)*
(Department_Research & Development, OverTime_No) → *(Attrition_No)*
(OverTime_No, DistanceFromHome_<=10) → *(Attrition_No)*

(StockOptionLevel_StockOptionYes) → *(Attrition_No)*
However, their lift is almost equal to 1 and *leverage* (support (X→ Y) - support(X)*support(Y)) is near to 0, that indicates statistical independence.

We explored the **distribution of confidence and lift values** for association rules generated with 0.6 threshold for confidence. They are shown on Figure 49 and Figure 50 respectively. It's interesting to observe that antecedents with only one item are positioned on the left (have slightly lower confidence) and those with 2 and 3 items are on the right. The distribution of lift values shows that most frequent value is 1, so the antecedent and consequent are independent and that rules with longer antecedents have higher lift.

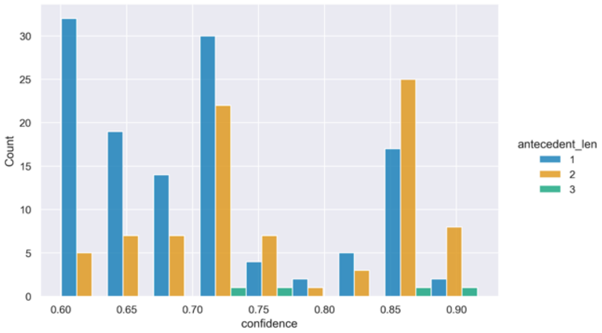


Figure 49. Count of rules for each confidence threshold with respect to antecedent length.

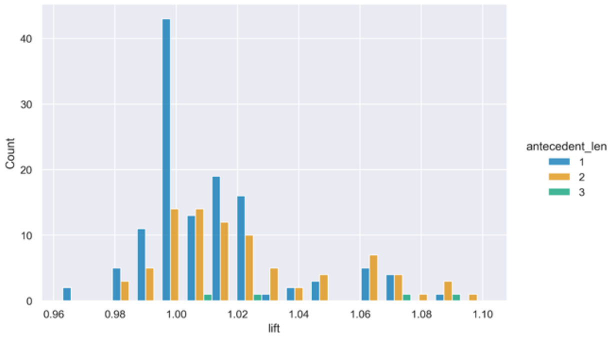


Figure 50. Count of rules for each lift value, taking in account antecedent length

Figure 48 shows the relation between support of the rule (that is defined as max(supp(X), supp(Y)) for (X→ Y)) and confidence. The rules with longer antecedents have smaller support but often higher confidence.

To explore the rules with our target variable Attrition_Yes, we verified that the variable starts to appear in the frequent item sets when the minimum_support threshold is set as low as 0.1 For min_support in [0.1, 0.09, 0.08, 0,07, 0.05] the number of the association rules with *'Attrition_Yes'* in consequents generated with min_lift=1 are respectively: [3, 9, 27, 60, 137, 480].

As our dataset is of reasonable size, we take a simple approach and examine the rules generated from the frequent item sets with very low min_support (0.08).

The rules with higher lift values are the following:

(OverTime_Yes) → *Attrition_Yes 1.89*
(StockOption_No) → *(MaritalStatus_Alone, Attrition_Yes) 1.87*
(Education_Tertiary, JobLevel_1)→ *Attrition_Yes 1.64*
The rules with highest lift values in absolute describe some trivial facts, for example: in sales department work people with marketing education:
(Department_Sales) → *(EducationField_Marketing, Education_Tertiary), lift: 3.29*

But also some more interesting observations: in this company most of the unmarried employees with low job position seems to be young scientists:

(MaritalStatus_Alone, JobLevel_1) → *(Department_R&D, TotalWYears_<=5), lift: 3.15*
And: those who occupy low level positions and male are starting their careers.

(JobLevel_1, Gender_Male) → *(TotalWorkingYears_<=5) lift: 2.35*

Frequent item sets with low min_support produce huge number of rules (348503), where more interesting rules are repeated sets of items with different combinations.

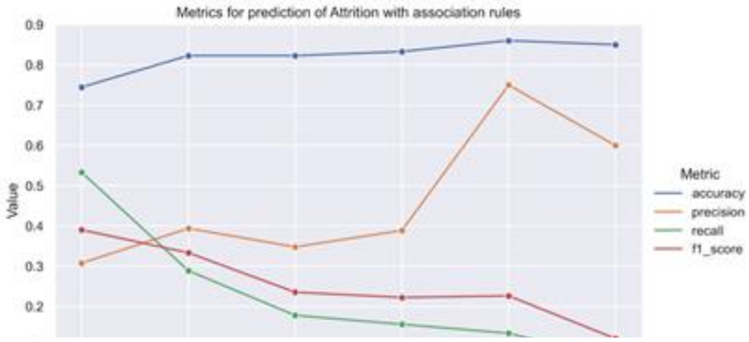


Figure 51. Evaluation metrics for prediction of target variable with 1 to 6 rules

To address **prediction of target variable Attrition_Yes** we divided our item base in training and test sets (same indexes as provided *hrtrain* and *hrtest*).

First, we simply examined the first 20 rules with the highest lift generated on the training set and identified 6 of them to apply for the test set. The rules that have only one consequent *Attrition_Yes* are shown in descending order of their lift value in **Figure** 52

We predicted the target variable *Attrition* with these rules and computed common metrics for model evaluation. While accuracy is quite high - **0.85**, precision is a lot lower **0.6** and recall is very low - **0.06**. The high accuracy is simply given by a large number of samples with *Attrition == No* value, in fact the accuracy is very near to guessing the most frequent class (0.846). The precision is quite high (very little 'stayers' were marked as 'leavers'), but in our case this is not so important, instead the recall is very low; in fact, the confusion matrix for this model (**Figure** 52) shows that only 3 from 45 leavers were identified correctly.

1. *{ ['OverTime'] == 'Yes' }*
2. *{ ['JobLevel']==1 }*
3. *{ ['YearsSinceLastPromotion']=='JustPromoted' }*
4. *{ ['Education'] == 'Tertiary' }*
5. *{ ['MaritalStatus']=='Alone' }*
6. *{ ['StockOptionLevel']=='StockOptionNo' }*

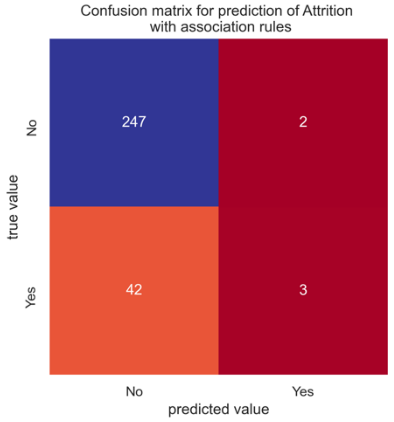


Figure **52**. Antecedents of the rules applied to predict Attrition and relative confusion matrix

We decided to check if the number of rules applied for prediction influences the accuracy of prediction, so we've calculated all the metrics for the same rules applied in an incremental manner. Indeed, as shown in **Figure 51**, recall is highest when only 1 rule is applied (*'OverTime==Yes'*): **0.53**, but precision is lower - **0.30** (54 'stayers' would be disturbed in vain by our suspect). The recall and precision values are inverted with 2 rules applied.

To replace **the missing values using the most meaningful rules**, we chose Department attribute, we randomly replaced the 20% of the values with null values and split the dataset in train and test set, using the record for which we knew the value of Department as training (1176 records) and the records with missing values as test set (294 records). We examined the first 20 rules with the highest lift respectively for the target values Research & Development, Sales and Human Resources, generated on the training set. We noted that there were no meaningful rules for the value Human Resources, as it has a very low support. Consequently, we decided to assign it the records that were not included in the other two classes. For the other two classes, we first selected the rules with a single antecedent and a single consequent with a higher lift value and support, then the rules with two, three, four items in the antecedent part of the rule. Since the accuracy did not improve adding more items, we chose the first four rules of one single antecedent with the highest lift value. The choice to order the rules by the number of antecedents is due to the fact that in this way the performances of the model improve considerably.

The first four rules with the highest lift value and one single antecedent for the class Research & Development are:

1) {Y['JobLevel'] ==1}-> Research & Development	*supp: 0.30, confidence: 0.80, lift: 1.227*
2) {['EducationField'] == 'Medical'} -> Research & Development	*supp: 0.24, confidence: 0.789, lift: 1.20*
3) {['EducationField'] == 'Life Sciences'} -> Research & Development	*supp: 0.305, confidence: 0.717, lift: 1.09*
4) {['MonthlyIncome]'== '>=8500'} -> Research & Development	*supp: 0.14 confidence: 0.68, lift:1.05*

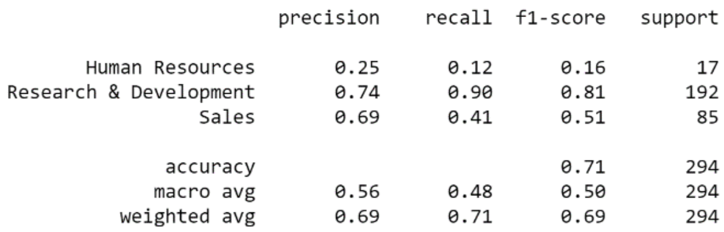
The first four rules with the highest lift value and one single antecedent for the class Sales are:

1) {['EducationField'] =='Marketing'} -> Sales *supp: 0.106,* ***confidence: 1.00, lift: 3.25***
2) {['JobLevel'] ==2)} -> Sales *supp: 0.165, confidence: 0.45, lift: 1.48*
3){['YearsAtCompany'] =='3-5'} ->Sales *supp: 0.11 confidence:0.32, lift: 1.068*
4) {['MonthlyIncome'] == '<=3500'} -> Sales *supp: 0.08, confidence: 0.33, lift: 1.08*

As we can see, the value of the support is low for all the rules, and also the lift is low. The lift is near to 1 (which means independence between the item sets) for all the rules except one. The rule with the highest confidence and lift is *{['EducationField'] =='Marketing'} -> Sales.* For the Sales class we can see that the difference between the lift of the first rule and the others is very high.

Overall, the reached accuracy on validation set is 0.75, which is not particularly high, but not bad either. In this case the three classes are imbalanced, and this negatively affects the accuracy. The best prediction is the one concerning the *Research & Development* class, while the prediction of the *Sales* class is much worse and the class *Human Resources* has been completely misclassified, as shown in the confusion matrix below. The results for the other metrics on the validation set are: Research & Development: precision = 0.75, recall = 0.95; Sales: precision = 0.80, recall =0.35; Human Resources: precision =0.33, recall = 0.22.

The results obtained in the test set are in line with those in the validation set, although the value of accuracy is a little lower and those of precision and recall for the Sales class are a little higher. As we can see from the recall measure, our model recognizes very well the elements of the Research & Development class (0.90), but very poorly those of the Sales class (0.41), for which we have many false negatives. The precision value shows that there are enough false positives in the prediction of the *Research & Development* class. This confirms the intuition that a large number of the records are categorized with the majority class label. The same is confirmed by the f1 score metric. In conclusion, we can say that our model has very low performance at this task.



```
                        precision    recall  f1-score   support

       Human Resources       0.25      0.12      0.16        17
Research & Development       0.74      0.90      0.81       192
                 Sales       0.69      0.41      0.51        85

              accuracy                           0.71       294
             macro avg       0.56      0.48      0.50       294
          weighted avg       0.69      0.71      0.69       294
```

Figure **53**. Evaluation metrics for missing values on the test set

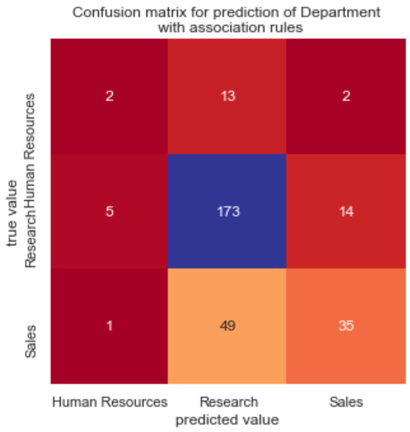


Figure **54**. Confusion matrix for missing values on the test set

References

Edwards M., Edwards K. Predictive HR analytics: mastering the HR metrics. Kogan Page, 2016

Berthold, M. R.; Borgelt, C.; Höppner, F. & Klawonn, F. Guide to Intelligent Data Analysis. Springer London, 2010

Selecting the number of clusters with silhouette analysis on KMeans clustering. https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html Retrieved 14/11/2020

Rakesh Agrawal. Fast algorithms for mining association rules. In Proceedings of the 20th International Conference on Very Large Data Bases (VLDB '94). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 487–499.