IE BUSINESS SCHOOL

MACHINE LEARNING II

CELINE KHOURY
PROFESSOR JESUS SALVADOR RENERO QUINTERO
FEB 17 - 2019

Section 01 - MBD

In order to help the management team of a company, a thorough analysis of the HR analytics dataset was conducted which includes explanatory variables of around 15k employees of the company. A model measuring the probability of attrition of employees (which includes employees leaving on their own or employees who got fired) within the company was generated taking into account the most relevant and important variables. This model will in turn, help the management team of the company understand what changes they can make within their workplace to retain most of their employees.

1- Data Loading and Data Preparation (Exploratory Data Analysis):

During this phase, some insights on predictors and data structure were provided in order to help the feature engineering work. The dataset will be transformed at this stage for example removing errors, dumifying variables (unfolding or creating different categories of one variable), imputing missing values… and the most important step during this phase is to leave the most informative features which will be used later on to test the model. A detailed explanation of the steps taken to build the model is provided hereafter.

After loading the data, a thorough exploratory data analysis was done to provide insights on predictors. First of all, all the libraries used were imported and the *function (read_data)* to read the raw dataset and the prepared one was defined. Moreover, a function (*fix_types*) to fix the types of the following variables: *Work_accident, promotion_last_5years and left* was defined, converting them to categorical variables. Both variables *sales* and *salary* were dummified. The variable *sales* was divided according to the different number of departments that we have resulting in 8 new columns (*sales_management, sales_marketing*…) and the variable *salary* was divided into three, depending on the different levels of salary in the dataset (low, medium and high) resulting in three new columns (*salary_low, salary_medium and salary_high* - encoding categorical variables with the *function onehot_encode* resulting in a total of 21 columns). In addition, no missing values were detected.

The *function feature_skewness* was defined which returns the features that are skewed. These features will be fixed later on, by the *function fix_skewness* using boxcox. The *function feature_scaling* rescales the numerical variables using min-max scaler. In addition, while preparing the dataset, no outliers were detected. Using spearman function to check for correlation between the variables (refer to heatmap in notebook), no correlation was found.

2- Baseline Model:

A basic model was trained on 80% of the raw dataset (training set) and then scored on a test hold-out - 20% of the dataset, to understand what we want to achieve and what will allow the accuracy of the model to increase. In addition, this basic model will check for the model performance on the classification problem using logistic regression. And the accuracy score for the baseline model is 0.787.

3- Feature Engineering:

During this stage, the data is transformed, and new features are created to better represent the way they affect the target variable (*left*). Consequently, some variables were plotted to check for their effect on the churn rate in order to understand what can be done to increase the accuracy of the model.

A- Feature Creation:

To improve the accuracy of the model, the following features (*time_spend_company, number_project and sales*) were grouped. And according to the plots of these specific variables in relation to employee attrition, each one was grouped subsequently:

1) The *function group_time_spend*: 2, 7, 8 and 10 years of time spent within the company were grouped into one single category and then dummified because the employees who spent this specific time at the company tend to have a low level of churn rate. To be more specific, these four groups were combined because the employee that has spent less time in a company (2 years) is more likely to stay because he just started and the employee who has spent 7 to 10 years within the company is a loyal employee meaning he will be less likely to leave as well.

2) The *function group_number_projects*: the number of projects (2, 6 and 7) were merged into one single category and then dummified because the frequency of leaving the company for the employees who have participated in these number of projects is quite similar.

3) The *function group_depts:* The departments (*sales_product_mng, sales_RandD, sales_marketing, sales_accounting, sales_management, sales_hr*) were grouped into one single category and they were, as well, dummified because the employee attrition in these six departments has a rather similar behavior which is seen in the plot titled: "Frequency of leaving per department".

4) The *function bin_satisfaction_level:* the *satisfaction_level* variable seemed to be one of the most relevant features in the dataset to determine the churn rate that's why, after analyzing the graph, it was binned which will allow the HR analytics department to understand and probably determine the churn rate based on the satisfaction level of the employees.

5) The *function bin_average_monthly_hours:* binning the average monthly hours spent by the employee within the company and this was binned according to the plot whereby the employees who left the company are represented with a blue color whereas the ones who stayed within the company are represented with a yellow color (which is the case for all the plots).

6) The *function bin_last_evaluation:* after analyzing the graph, it seemed logical to put the performance (*last_evaluation*) into groups which behave the same towards the churn rate.

The following functions (*relative_satisfaction_salary, relative_working_hours_salary and relative_last_evaluation-salary*) create features by applying the following formula:

$$\frac{x_i - \bar{x}}{s}$$

where Xi: is one of the variables
$\bar{X}$: stands for the average of Xi for the employees **that have the same salary**
S: stands for the standard deviation of the average of Xi for the employees **that have the same salary**

7) The *function relative_satisfaction_salary:* in order to make more sense out of the dataset that we have, this feature was created, which was calculated by the formula provided above where:
Xi: stands for the *satisfaction_level*.
$\bar{X}$: stands for the average of the satisfaction level for the employees that have the same salary.
S: stands for the standard deviation of the satisfaction level for the employees that have the same salary.

Normally, employees compare their satisfaction level to other employees that have the same level of salary and according to that, the HR department would be able to get insights on the employee attrition according to the satisfaction level of the employees that are being paid similarly. After analyzing the plot, a trend was detected and hence the *function bin_relative_satisfaction* was defined in order to bin this new feature according to the similar behavior it has towards employee attrition.

8) The *function relative_working_hours_salary:* correspondingly, this feature was created and was calculated by the same formula but here:
Xi: stands for the *average_montly_hours.*
$\bar{X}$: stands for the average of the average monthly working hours for the employees that have the same salary.
S: stands for the standard deviation of the average monthly working hours for the employees that have the same salary.

Generally, employees compare their working hours with their colleagues that have the same position meaning the ones that have relatively a same salary, for example if an employee is

unhappy with his salary and his average monthly hours is very high compared to others that are paid the same, this employee might tend to leave the company more than an employee that has the same salary but works much less. After analyzing the plot, a trend was detected and hence the *function bin_relative_work_hours* was defined in order to bin this new feature according to the similar behavior it has towards employee attrition.

9) The *function relative_last_evaluation_salary:* equally, this feature was created which gives us insight on the employee performance depending on the level of salary that the employee has and the effect it has on the employee attrition. It was calculated by the same formula where:
   Xi: stands for the *last_evaluation*
   $\bar{X}$: stands for the average of the last evaluation for the employees that have the same salary.
   S: stands for the standard deviation of the last evaluation for the employees that have the same salary. After analyzing the plot, a trend was detected and hence the *function bin_relative_last_eval* was defined in order to bin this new feature according to the similar behavior it has towards employee attrition.

B- Feature Learning:

Genetic Programming is an algorithm inspired by the phenomenon of natural selection whereby the variables forming the current model are evaluated (fitness) and then the best variables, the most informative ones will be selected (crossover) and then independent random changes are performed (mutation). All of this is done using symbolic regression and after applying this algorithm, xx columns were generated.

Using a pipeline function in which a cross validation method was applied, the functions that increase the mean of the different scores obtained by the k-fold splits were outputted. A new dataset with the added features will be used from now on and the model that was trained through this function will be dropped.

Variable	New Mean Score
relative_satisfaction_salary	0.7897
relative_working_hours_salary	0.7909
relative_last_evaluation_salary	0.7911
bin_relative_last_eval	0.8413
bin_relative_work_hours	0.8715
bin_relative_satisfaction	0.9096
group_time_spend	0.9423
group_depts	0.9420
group_number_projects	0.9508

feature_normalizing	0.9512
bin_satisfaction_level	0.9614
bin_average_monthly_hours	0.9599
bin_last_evaluation	0.9595
Genetic_P	0.9693

C- Feature Selection:

RFECV which stands for Recursive Feature Elimination Cross Validated (using k-fold of 10) was applied, which will in turn get the most relevant features of the dataset that will allow the optimization of the accuracy score, accounting for 37 features out of the 80 features that we already had.

4- Final metric:

The dataset was reduced to the 37 columns resulting from the feature selection. In order to make sure that the trained model is not biased towards any portion of the dataset, the dataset will be sliced into a train set of 80% and a holdout test set of 20%. To avoid overfitting, a cross validation of 10 k-folds will be applied on the train set and the function (*get_best_fit*) will be used to iterate through each of the 10 k-folds and then generate the model trained which yielded the highest accuracy score (0.968 from k = 9). Predicting the target variable (*left*) from the test set and using the trained model, an accuracy score of **0.96** is obtained.