Ensemble Learning Project Proposal

Maria Isabel Vera Cabrera, Chiara Palma,
Shwetha Salimath, Lauren Yoshizuka, Cheng Wan

January 2022

1 Introduction

For any company, the hiring process requires many important decisions to be made. A hiring manager must focus on selecting the most motivated and capable candidates while mitigating potential turnover rate. Employee selection usually involves interviews that assess subject knowledge, skills, and personality to ensure the right fit within the company. However, there is always some degree of asymmetric information between the candidate and the hiring company. Even if the right candidate is offered the job, they may not decide to accept the offer, or if hired, there is uncertainty that they remain in the position long-term. This project focuses on reducing business costs and training time during the hiring process.

1.1 Motivation

The benefits of developing a model that accurately predicts whether a candidate will accept or reject a job offer, including the most likely rationale for their decision, are two-fold: 1. companies will be able to be more cost-efficient, and 2. employees will be more likely to remain at the company in the long-term.

2 Problem Definition

The project is to predict if a candidate would stay with the company after completion of the training process given information about demographics, education, experience submitted by the candidates during enrollment. Also to understand the factors affecting the candidates decision.

This project has been obtained from a kaggle challenge titled, "HR Analytics: Job Change of Data Scientists.Predict who will move to a new job"

3 Data-set Description

There are two datasets available, both in csv format, one for training and one for testing. The training dataset contains 19158 instances, while the test dataset contains 2129 rows. Each instance corresponds to one candidate for a data scientist position. They are uniquely identified by the key *enrollee_id*. There are 12 attributes for each person, which are the following explanatory variables:

- *city*: City code in the format "city_"+numerical code
- *city_development_index*: Development index for the city. Decimal value between 0 and 1.
- *gender*: Category Male or Female.
- *relevant_experience*: Two categories for professional background of the candidate in data science: Has relevant experience or No relevant experience.
- *enrolled_university*: Type of enrollement of the candidate with university studies. Categories: Full time course, Part time course, no_enrollement
- *education_level*: Highest education title obtained. Possible options are Primary School, High School, Graduate, Masters and Phd.
- *major_discipline*: Major field of studies of the candidate. It can be Arts, Business Degree, Humanities, STEM, No Major, Other.
- *experience*: Years of professional experience. It is categorical for the extreme values: <1 or >20 and integer for values in the middle: from 1 to 20, inclusive.

- *company_size*: Categorical range of the number of employees in the company where the candidate is currently working at.
- *company_type*: Type of company where the candidate is currently working. It can be Early Stage Startup, Funded Startup, NGO, Public Sector, Pvt Ltd, or Other.
- *last_new_job*: Years between the previous job and the current one. It could be 1, 2, 3, 4, >4 or never.
- *training_hours*: Number of training hours completed. Integer value.

Most of those independent variables contain missing values, especially the categorical features. This has to be taken into account during the pre-processing stage.

The target to predict is a binary variable called *target* with value 1 if the candidate is looking for another job and 0 if he/she is not.

4 Proposed Methodology

In order to complete this classification task, it is possible to implement and use several tree based methods. A non-exhaustive list of the latter is introduced in the following section. The performance of all these models will be then evaluated according to different accuracy scores.

4.1 Models

4.1.1 Decision Tree Classifier

A decision tree is a predictor, i.e. $h : X \to Y$, that predicts the label associated with an instance x by traveling from a root node of a tree to a leaf. The splitting rule for each node of the tree is based on threshold the value of a single feature. In other words, we move to the right or left of the child node based on $1_{[x_i<\theta]}$, where $i \in [0 : n]$ is the index of the relevant feature, n is the number of features and $\theta \in R$ is the threshold.

4.1.2 Bagging techniques

Bagging is an ensemble techniques which uses multiple base learned trained separately with a random sample from the training set. The algorithsm aims at reducing the error, using a voting or averaging approach across the different base learners to produce a more stable and accurate model. One of the key advantages of bagging is that is can be executed in parallel since there is no dependency between estimators.

Random Forest is a bagging techniques which uses tree as base learners. While it uses the sample principle of bagging, it provides an additional step consisting in the randomization of the selection of the attributes of the nodes in order to reduce the variance of the obtained estimator.

4.1.3 Boosting techniques

Boosting is a group of ensemble techniques which make prediction using different weak prediction models, typically decision trees. Differently from bagging, in boosting techniques weak learners are trained sequentially and the error made by the previous base learner is used to improve the stability of the model. The way in which the errors of each base learner is considered to be improved with the next base learner in the sequence, is the key differentiator between all variations of the boosting technique.

Gradient Boosting aims at minimizing the residuals of each learner base in a sequential way. This minimization is carried out through the calculation of the gradient applied to a specific loss function. Then each base learner added to the sequence will minimize the residuals determined by the previous base learner. This will be repeated until the error function is as close to zero or until a specified number of base learners is completed. Gradient boosting could be considered together with different implementations such as XGBoost and CatBoost can be considered as well.

AdaBoost, or adaptive boosting, algorithm is based on the idea of obtaining highly accurate predictions by combining many relatively inaccurate predictions. The final prediction is obtained by weighted majority vote. This method applies weights to each training sample and then for each iteration, then it re-assigns weights to each instance such that higher weights will be assigned to wrongly classified instances and the weights are decreased for those that were predicted correctly. By doing so, more difficult cases will be addressed by subsequent classifiers.

4.2 Evaluation Methodology

In the employee turnover analysis, we would like to explore which of the employees would like to make job changes (the positive class). We will include several evaluation metrics to assess five supervised machine learning algorithms. The performance of the classifier is validated based on five metrics; Accuracy (Train/Test), Precision, Recall or Sensitivity, Specificity, and F-1 score.

Accuracy is the ratio of numbers of correct predictions over the amount of total examples. In general, the higher the precision of the model, the better the model is.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Precision represents the percentage of samples identified by the model as positive that are true positive. The precision reflects the algorithm's ability to distinguish negative samples. The higher the precision, the stronger the algorithm's ability to distinguish negative samples. In the context of workforce attrition analysis, the precision score is expected to be maximized, because low precision means out of all potential candidates HR will contact, a lot of them actually don't plan to change jobs. Resources are wasted on contacting false positive candidates.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall (or Sensitivity) is the ratio of the number of samples correctly identified as positive divided by the total number of positive samples. It reflects the algorithm's ability to recognize positive samples. The higher the recall, the stronger the algorithm's ability to recognize positive samples. Maximizing the recall score in workforce attrition analysis signifies to capture as many individuals who are looking for job changes as possible.

$$Recall/Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

F1-score is a combination of the precision and recall. It is a measure of the classification model, defined as the harmonic mean of precision and Recall, with values ranging from 0 to 1, with 1 being the best and 0 being the worst. To reach the optimal balance of precision and recall and thus find the most optimal combination of resampling method and machine learning algorithm, the project aims to maximize the F1-score.

$$F1 - score = \frac{2 \times Precision \times recall}{Precision + Recall} \tag{4}$$

All the parameters obtained through the experiment will be displayed in evaluation metrics with respect to different learning models.

The ROC (Receiver Operating Characteristic) curve and AUC (Area Under the Curve) values will be used to evaluate the merit of a classification model. The ROC-AUC values and ROC curves could differentiate different classifiers with True Positive Rate (TPR) and False Positive Rate (FPR) and the under-curve area. The larger the TPR, the more actual positive classes are predicted in the positive class. The ideal target: TPR (the vertical axis) = 1, FPR (the horizontal class) = 0, i.e., the closer the ROC curve is to the (0,1) point, the more it deviates from the 45-degree diagonal. AUC measures the entire two-dimensional area underneath the entire ROC curve, which ranges in value between 0 and 1. AUC as a value can be used to evaluate the classifier intuitively; the larger the value, the better the prediction.