IBM HR Analytics Employee Attrition & Performance ML Project Observations

Author: Ömer Faruk Merey
Dataset: https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset

Employee attrition occurs when the size of your workforce diminishes over time due to unavoidable factors. Employee resignation is one of the most demolishing outcomes of this problem. This dataset was created by IBM to analyze the employees that are leaving the company. The dataset was created on personal and company related features of the employees as well as their attrition.

In the first part of my project, I used matplotlib to visualize data with same important statements. Such as gender gap, age, satisfaction level, work-life balance, and many more to attrition levels. In this phase, I tried to understand which features are the most important to analyze when an employee quits his/her job. After finishing my analysis, I checked out the correlation matrix of our dataset to understand which features are correlated and which are not. I found a quite number of features that are correlated similarly which was discarded in the code along with features that were not correlated at all. This correlation matrix was also a good measure to see which features were correlated with attrition. I used Label Encoder for non-numeric values. After all, I scaled age and the income attributes because of the range they had in their values which made a huge improvement on my model.

The dataset is very imbalanced in so ways. The attrition rate was %16 to %84. Working with a dataset like this was challenging. I used stratify to balance the data when cross-validating in grid search cv but even though stratifying the data, in some occasions it was impossible to balance the attrition rate which really effected my models while training.

I used cross-validation with grid search after splitting my test/train data. I used %15 of my data to test while %85 of my data to grid search with a cv of 5. Training my models, I have observed expected results because of our disparity in our dataset. Precision is more focused in the positive class than in the negative class, so since our data contains much more negative class our precision level are not really promising. I even used stratify which made a little improvement by balancing the attrition rates but after all it was still a problem because the amount of yes and no's were not enough.

When it comes to comparing my models, some of the evaluation metrics results were same and it had a few differences. I think it's because of the data disparity which caused some interesting results. Speaking of disparity, even though there is a difference between the train and test evaluation scores of our best model the difference is probably because of the data sample size. There is about 1500 records of employees and in my conclusion, this is not enough data for this problem. So, there was not a significant sign of overfitting when we observe the models results.