Battery Risk Prediction Challenge

Topcoder Handle: aukintux

March 31, 2018

Abstract

The following document describes the various learning algorithms fitted to the dataset and the results obtained in the quest of optimizing the score metric. The various tasks were performed in a 8GB machine on Jupyter Notebooks.

1 Score Metric

Let us analyze the score metric for this challenge. The final score F_s is defined as follows:

$$F_s = F_1 + 1 - MRAE_c \tag{1}$$

Where $MRAE_c$ stands for custom mean relative absolute error and is defined as follows:

$$MRAE_c = \frac{1}{N}\sum_{i}^{N} min\left(1, \left|\frac{P-G}{G}\right|\right) : G > 0 \tag{2}$$

This previous formula sums over all values whose ground truth value G is greater than zero. Let us see that if the predicted value P is -1, then, the component inside the sum will be 1 since G is greater than zero.

This takes into account the requirement that: "*If P is -1, the relative absolute error is defined as 1*".

The metric is then composed of two parts: $0 < F_1 < 1$ and $0 < MRAE_c < 1$. It is important to note that since $MRAE_c$ only sums through the samples whose ground truth label is strictly positive and in addition each component inside the sum is bounded from above by 1. Then, it is never a good idea for the algorithm to predict $P = -1$ since the results will hit the upper bound instantaneously.

2 Algorithm Selection

In both cases the algorithms were fitted on the processed data and a 5-fold *GridSearchCV* was used in order to find the *best_params_*.

2.1 F_1 Optimization

In order to optimize the F_1 score where $risk = 0$ is considered to be the positive class and $risk! = 0$ is considered to be the negative class different classification algorithms were applied:

1. SVC
2. SVC with PCA
3. RandomForestClassifier
4. RandomForestClassifier with PCA
5. DecisionTreeClassifier
6. DecisionTreeClassifier with PCA

The best performing one was found to be the DecisionTreeClassifier. Even tough PCA can reduce the number of features to 50 with a cummulative variance of 99.37% the algorithm without PCA is not computationally expensive. It takes less than 5 minutes to train and gives a light improvement.

2.2 $MRAE_c$ Optimization

In this case only the samples where $risk > 0$ where used in order to train the model. A few regression algorithms were fitted to the data:

1. ElasticNet
2. ElasticNet with PCA
3. SVR
4. SVR with PCA
5. RandomForestRegressor
6. RandomForestRegressor with PCA
7. DecisionTreeRegressor
8. DecisionTreeRegressor with PCA

The algorithm that best performed was the DecisionTreeRegressor.

3 Results

3.1 F_1 Score

The result of using the 5-fold *GridSearchCV* with F_1 as scoring showed that the best parameters for the *DecisionTreeClassifier* are **max_depth: 20** and **min_samples_split: 7**. Subsequently, the F_1 score for such set of parameters is $\mathbf{F_1 = 0.9981072555205047}$ and $\sigma_{\mathbf{F_1}} = \mathbf{0.0018936}$. The following are visual representations of the previous results.

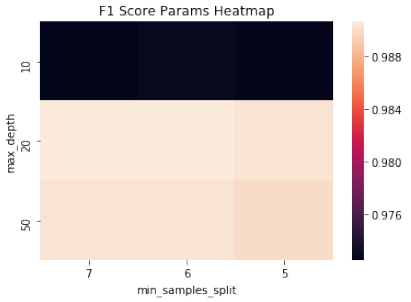

Figure 1: Heatmap around the optimal parameters for the DecisionTreeRegressor.

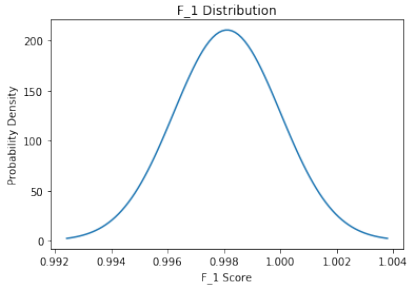

Figure 2: Distribution of the F_1 score for the best set of parameters.

3.2 $MRAE_c$ Score

The 5-fold *GridSearchCV* using the custom scorer $MRAE_c$ as scoring parameter showed that the best set of parameters for the *DecisionTreeRegressor* are are **max_depth: 30** and **min_samples_split: 4**. Subsequently, the $MRAE_c$ score for such set of parameters is $MRAE_c = \mathbf{0.018023652156838658}$ with standard deviation $\sigma_{MRAE_c} = \mathbf{0.00115203}$. The following are visual representations of the previous results.

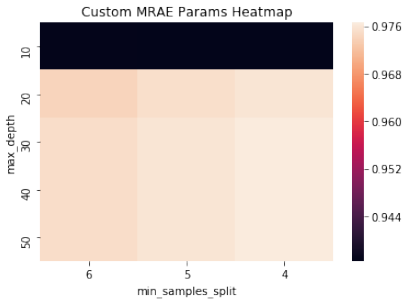

Figure 3: Heatmap around the optimal parameters for the DecisionTreeClassifier.

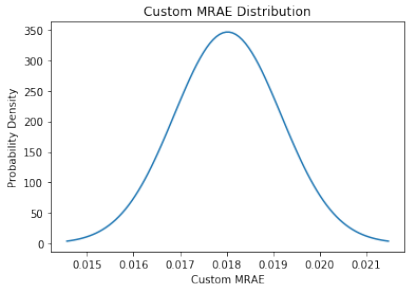

Figure 4: Distribution of the $MRAE_c$ score for the best set of parameters.

3.3 Final Score

The final score obtained with this particular design is: $F_s = 1.9800836033636662$. The practical routine of *sklearn* for the decision tree include a *random_state* internal variable. This was set randomly to 42 in order to get reproducible results.