ML Final Project Report

- Author : Chi-Chih Chang
- ID : 0811514

Introduction

In this work, we are trying to solve a prediction problem, given a tabular data. The dataset itself is from the kaggle contest that host in 2022 Aug. In the following section, I will go through the methodologies and summarize the cool idea that learned from the kaggle forum.

Methodology

Data Pre-processing

In this section, we will discuss the two main data-propressing that is introduced in our solution, including the data imputation and derived feature.

Data Imputation First, let's talk about the data imputation. Originally, some data entries will contain some null value at some features, and we need to find a way to impute it so that this data can be used. In the data imputation procedure, I first utilize the `k-nearest-neighbor` technique to infer the null value based on the others. Secondly, from the kaggle forum, I found some interesting feature correlation at feature `measurement_3~9` and `measurement_17`. The property is that, when we first group the data by the `product code` feature, we can observe some slightly higer correlation with `measurement_17`, as shown below :

```
For example within Product A correlations:
Meas_4 x Meas_17 = 0.14
Meas_5 x Meas_17 = 0.56
Meas_6 x Meas_17 = 0.28
Meas_7 x Meas_17 = 0.24
Meas_8 x Meas_17 = 0.74
```

Something more interesting is that, when we multiply each measurement by its correlation value with `measurement_17` and sum it all up, we will get a new feature which is perfectly correlated with `measurement_17` for example:

```
0.14 * Meas_4 + 0.56 * Meas_5 + 0.28 * Meas_6 + 0.24 * Meas_7 + 0.74 * Meas_8
And this new feature is correlated with measurement_17 by 0.97.
```

This information give us some insigth that, when we first group the data by the `product code` feature, we might use `measurement_3~9` to build a regression model to infer the missing value of `measurement_17`. Using the information, I add a linear regression and tried fillig the missing value of `measurement_17`.

Derived Feature Secondly, let's discuss on the derived feature. Inspired by this discussion, some people point out that the show up of the missing value at `measurement` feature might be an useful indicator that is highly correlated with the product failure rate. To verify this idea, we utilize the statistic methods to do so. First, we make a null hypothesis that the measurement is not correlated with the failure rate, and we calculate the conditional product failure rate given the measurement is missing E[product fails | measurement is missing] and compare it to the unconditional product failure rate, which is 0.212608. If the null hypothesis is correct, the deviation will be small. Say that the failure count is binomially distributed with p = 0.212608, we can calculate the z-score and p-value using the approximation and get the following result :

```
feature             fail   miss   failure rate       z     p-value
loading           :   44 /   250 = 0.176          -1.41      0.157
measurement_3     :   61 /   381 = 0.160          -2.50      0.012
measurement_4     :  128 /   538 = 0.238           1.43      0.151
measurement_5     :  172 /   676 = 0.254           2.66      0.008
measurement_6     :  171 /   796 = 0.215           0.15      0.879
measurement_7     :  197 /   937 = 0.210          -0.18      0.860
measurement_8     :  218 / 1048 = 0.208          -0.36      0.716
measurement_9     :  283 / 1227 = 0.231           1.54      0.123
measurement_10    :  277 / 1300 = 0.213           0.04      0.967
measurement_11    :  311 / 1468 = 0.212          -0.07      0.944
measurement_12    :  356 / 1601 = 0.222           0.95      0.340
measurement_13    :  373 / 1774 = 0.210          -0.24      0.809
measurement_14    :  413 / 1874 = 0.220           0.82      0.411
measurement_15    :  430 / 2009 = 0.214           0.16      0.876
measurement_16    :  436 / 2110 = 0.207          -0.67      0.502
measurement_17    :  499 / 2284 = 0.218           0.69      0.493
```

Here, we can clearly see that the show up of missing value at feature `mearsurement_3` and `measurement_5` has relatively low p-value (<0.01), thus, we can reject the null hypothesis.

Using this idea, I add the following two line to aument these to feature into the dataset :

```
X['m_3_missing'] = X.measurement_3.isna()
X['m_5_missing'] = X.measurement_5.isna()
```

Model Architecture

For the model architecture, I select the `LogisticRegression` from `sklearn` to build the base model. Then, following the insight from some EDA result done by prior work, I select the following feature to train the model :

- `m3_missing` , `m5_missing`
- `mearuement_4` , `measurement_5`
- `loading`
- `m_5_missing` , `m_3_missing`

To avoid overfitting, I randomly select part of these feature to train the model, and finally ensemble them to derive the final prediction. Here, I use four model to conduct the ensemble.

Hyper-Parameters

About the LogisticRegression

For the logistic regression model, I using the sklearn for implementation, where Maximum number of iterations taken for the solvers to converge is set to be 1000, inverse of regularization strength is set as 0.0001, and the `newton-cg` is set to be the solver. Besides, `l2` penalization is also been enable to avoid possible overfitting.

About ensemble

For the ensemble detail, here I use four small model to ensemble. All these four models use the `LogisticRegression` with same hyperparameter to serve as the base model, execpt the feature that select to join the training, in the following list, we provide the feature that we use in the training for each small model:

- model1: `'m3_missing', 'm5_missing', 'measurement_1', 'measurement_4', 'loading', 'measurement_17', 'attribute_3'`
- model2: `'measurement_1', 'measurement_4', 'loading', 'measurement_17'`
- model3 `'m3_missing', 'm5_missing', 'measurement_4', 'loading', 'measurement_17'`

- model 4 : `'measurement_4', 'loading', 'measurement_17'`

Finally, the final prediction is produced by linearly combining the prediction logits with the following equation : `result = 0.2 * model1(X) + 0.25 * model2 + 0.25 * model3 + 0.3 * model4` .

Experimental results

Main Result

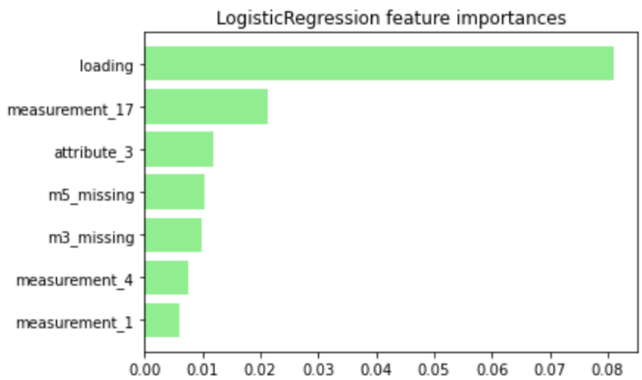
By using the setting describe above, we can get `0.59069` at the privat leader board.

Submission and Description	Private Score	Public Score	Selected
submission.csv Complete (after deadline) · now · Version1	**0.59069**	**0.58929**	

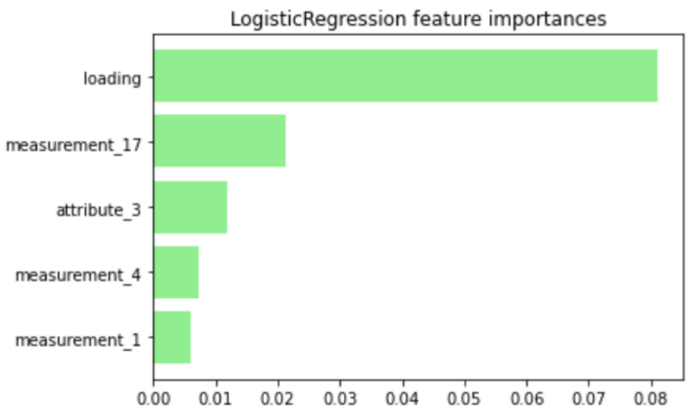
Ablation Study

Feature importance In the following section, let us analyze the feature importnace score of each small model train with different feature.

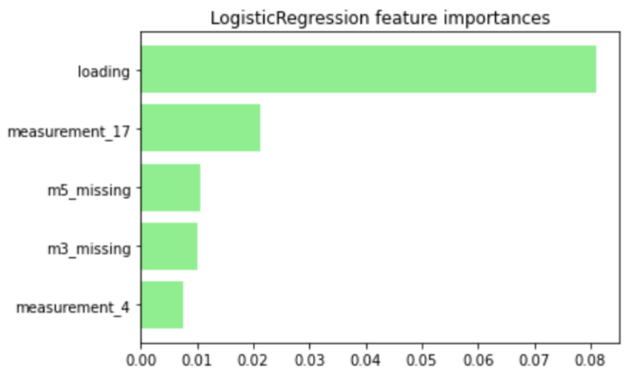
- Model 1



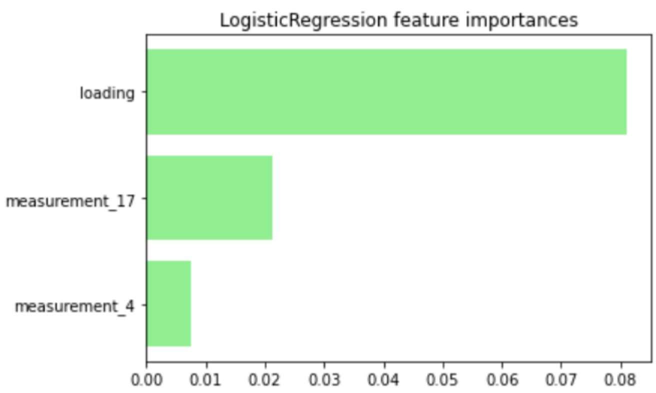
- Model 2



- Model 3



- Model 4



From the result, we can see that `loading` might be the most important feature. Besides, seeing from the result of model1, we can clearly see that `m_3_missing` and `m_5_missing` to actually play a role when doing the prediction.

Performance of model with different feature In the next section, we analyze the performance of each small model with different feature.

	model1	model2	model3	model4
Average auc	0.59116	0.59032	0.59103	0.5904

From the table showing we can see that the model1 and model3 with `m_3_missing` and `m_5_missing` feature, perform slightly better than the other one without this feature, and suprisingly, at model4 when using simply three feature, we can also gain the competive performance. We conjecture that it may beacause that the most importnace feature loading is still exist.

Summary

In this work, we go through the classic procedure to solve the ML problem, including the data-preprocessing and model building, also some ensemble idea is also added to boost the performance. Though the survey to the prior work, we sucessfully pass the baseline.

Reference

- https://www.kaggle.com/competitions/tabular-playground-series-aug-2022/discussion/349299
- https://www.kaggle.com/competitions/tabular-playground-series-aug-2022/discussion/343939
- https://www.kaggle.com/competitions/tabular-playground-series-aug-2022/discussion/342126
- https://www.kaggle.com/code/takanashihumbert/tps-aug22-9th-solution?fbclid=IwAR3SLn0XqsvYNUpkvHM0DCe8rflOW_82-39mlgAXNbPFu_wxf1l8MGoVGPA