FLIP00 FINAL PRESETANTION REPORT

RONGXIN XU

Abstract. This report contains five parts. First, introduce the definition of the problem, describe the data and analyze the problem. Second, statistical the data information, visualize the data to find some potential relationships between the attribute values and process the data like dummy variables, feature engineering, feature selection etc. Third, explain the most important parameters of different algorithms, and the method of experiment. Fourth, experiment and analyze the performance of different algorithms based on experimental result. The last one is conclusion .

Contents

Date: 2019-11-14.
1991 *Mathematics Subject Classification.* Artificial Intelligence.
Key words and phrases. Machine Learning, Data Mining, ...

1. Introduction

1.1. **Problem Statement.** This is a problem with time-series prediction. After a month of making scientific observations and taking careful measurements, can predict total sales for every product and store in the next month. The raw dataset contains train set with 2935849 samples and 214200 unlabeled samples as test set. Through the train data, predict total sales for every product and store in the next month.

1.2. **Data List.**

There are six data sets with a total of 11 attributes, the flowings are the name and meaning of attributes

id: an Id that represents a (Shop, Item) tuple within the test set.
shop_id: unique identifier of a shop.
item_id: unique identifier of a product.
item_category_id: unique identifier of item category.
item_cnt_day: percentage of soul in the creature.
item_price: current price of an item.
date: date in format dd/mm/yyyy.
date_block_num: unique identifier of item category.
item_name: name of item.
shop_name: name of shop.
item_category_name: name of item category.

1.3. **Problem Analysis.**

1.3.1. *Train Data and Test Data.*

Because this game is over, it can't submit the predictive values for testing. So I divide the raw train data into train data and test data, and the ratio is 8:2. Another vexed problem is the data too small, so I use ten-fold cross-validation to train the models.

1.3.2. *Problem Possible Solutions.*

There are many machine learning algorithms can solve the Time series prediction problem, such as xgboost, random forest and so on. Use CV to find the best parameters of the algorithms and then validate with testing data. But the most important thing is do feature engineering to improve accuracy.

1.3.3. *Evaluation Methods.* Before experiment, determine the evaluation methods to assess the model performance is very important, usually it has the following methods for classification problem:

- F1 Score/AUC
- Class Accuracy
- Recall
- Precision

2. Exploratory Data Analysis

2.1. **Data Information.**

The following table 1 is the statistical result of each attribute in sales_train.csv. There are 6 numerical variables, and no missing values. The data is very clean and complete, So let's start visual analysis.

TABLE 1. Add caption

	date_block_num	**shop_id**	**item_id**	**item_price**	**item_cnt_day**	**item_category_id**
count	2935849	2935849	2935849	2935849	2935849	2935849
mean	14.57	33	10197.23	890.62	1.24	40
std	9.42	16.23	6324.3	1726.44	2.62	17.1
min	0	0	0	-1	-22	0
25%	7	22	4476	249	1	28
50%	14	31	9343	399	1	40
75%	23	47	15684	999	1	55
max	33	59	22169	307980	2169	83

2.2. **Data Visualization.**

Use EDA to plot the distribution of the data, can observate the data intuitively and find the relation between the attribute values. For example boxplot can visually observe the distribution of numerical variables, scatterplot can show their distribution trends and whether exists outliers. For classification problems, the data with the same label is drawn in same color, which is very helpful for the construction of the Feature.

2.2.1. *Histogram.*

The figure Figure 1 shows the mean of the four numerical variables and the figure Figure 2 is the number of different color about types of ghastly creatures. It seems that all numerical features may be useful, but many colors are evenly distributes among the monsters, which means they maybe have little effect on classification.

2.2.2. *Boxplot.*

When analyzing the data, the boxplot can effectively help us identify the characteristics of the data: visually identify outliers in the dataset or determine the data dispersion and bias of the data set. Through the figure Figure 3, we know that the two types of Ghost and Ghoul in the monster have higher discrimination on the four variables, while Goblin is in the middle position, which intersects with the other two types of features. Based on the above observation, we guess that the predictive accuracy of Ghost and Ghoul will be better than Goblin. And the outliers are very small, which can be ignored.

2.2.3. *Pairwise Plot.*

Pairwise plot is a favorite in exploratory analysis to understand the relationship between all possible pairs of numeric variables. This pairplot Figure 4 shows that data is distributed normally. And while most pairs are widely scattered (in relationship to the type), some of them show clusters: hair_length and has_soul, hair_length and bone_length. So it may need to reassemble the data.

2.2.4. *Correllogram.*

Correlogram is used to visually see the correlation metric between all possible pairs of numeric variables in a given dataframe. This figure Figure 5 make it convenient for us to analyze features, especially their impact on the 'type' column. As we can see the 'type' column has a high value of negative correlation with

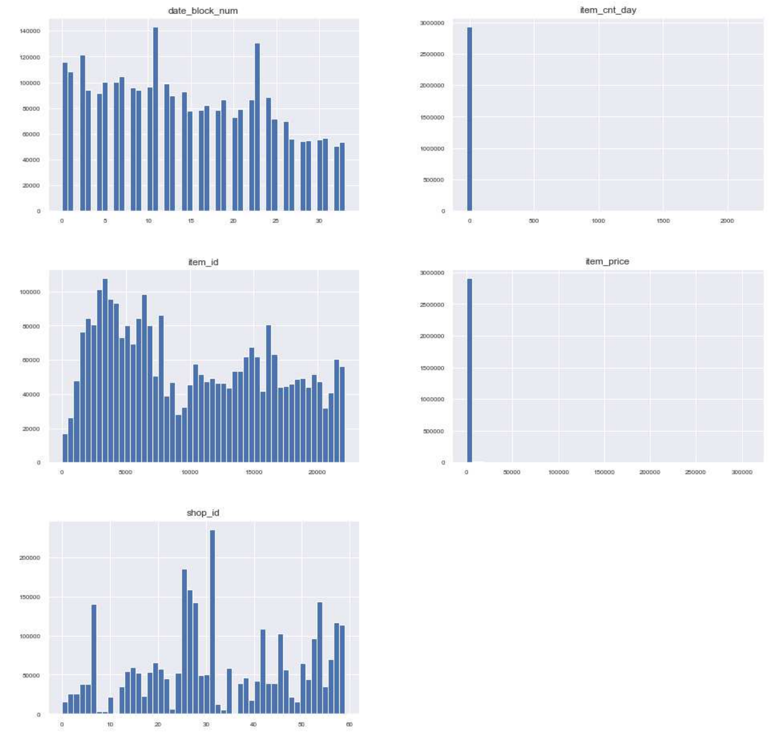

Figure 1. The Mean of Four Numerical Variables

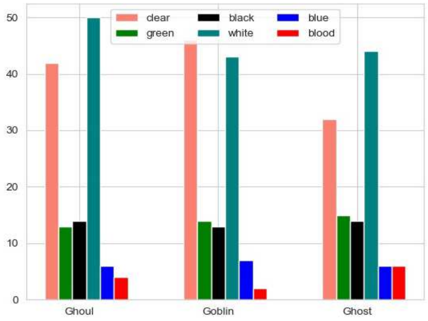

Figure 2. Color distribution Grouped by Type

columns 'has_soul' and 'rotting_flesh', but the correlation with the 'hair_length' is not very big. There is no obvious linear relationship between these variables.

2.2.5. *Other Figures.* The following pictures are independent of the choice of algorithm, Because they look great, so I want to share with you.

2.3. **Data Preparation.**

2.3.1. *Great New Features.*

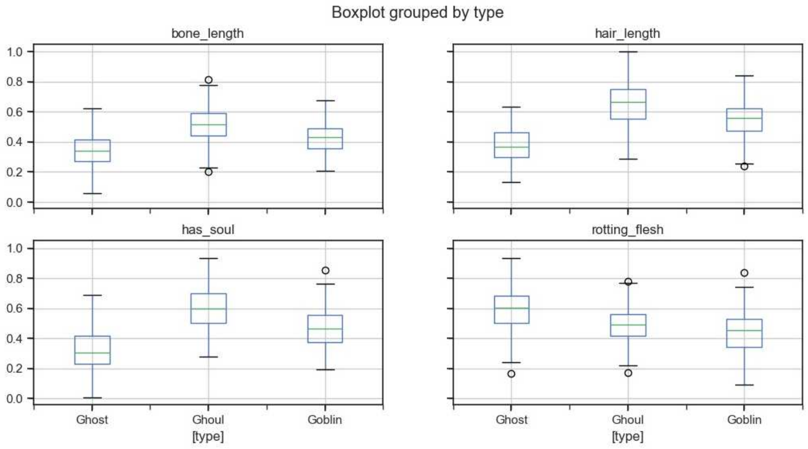

FIGURE 3. Boxplot Grouped by Type

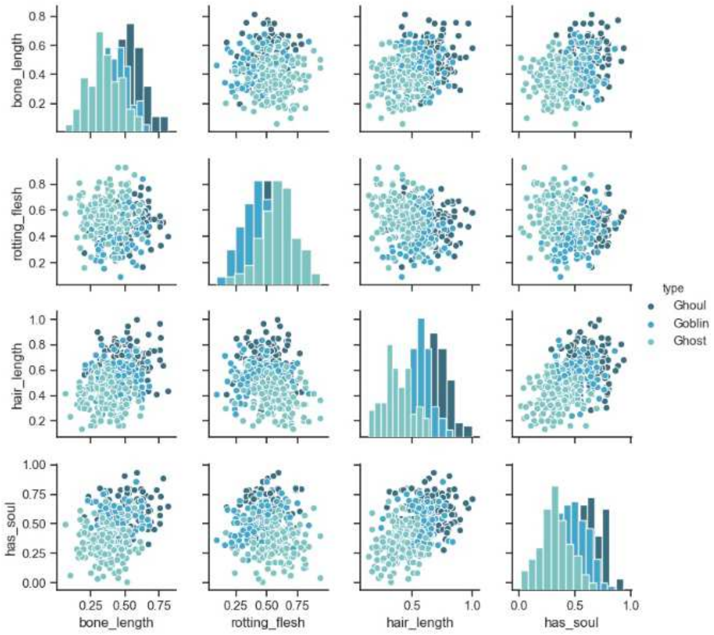

FIGURE 4. Feature Scatterplot

As it can be seen from the pictures above the data is distributed normally. But some of them show clusters: hair_length and has_soul, hair_length and bone_length. So I create new variables with multiplication of these columns:

hair_soul: row[hair_length]*row[has_soul]
hair_bone: row[hair_length]*row[bone_length]
bone_soul: row[bone_length]*row[has_soul]
hair_soul_bone: row[hair_length]*row[has_soul]*row[bone_length]

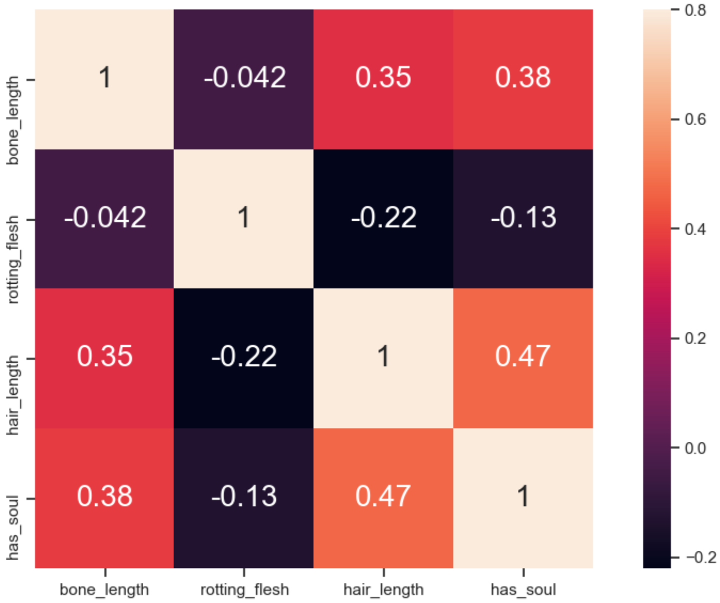

FIGURE 5. Correllogram

FIGURE 6. Marginal Histogram

Then analyse the new features in a pairplot, showing the picture Figure 9 below. It can be seen from the picture that there is a clear linear relationship between the variables.

2.3.2. *Feature Selection.*

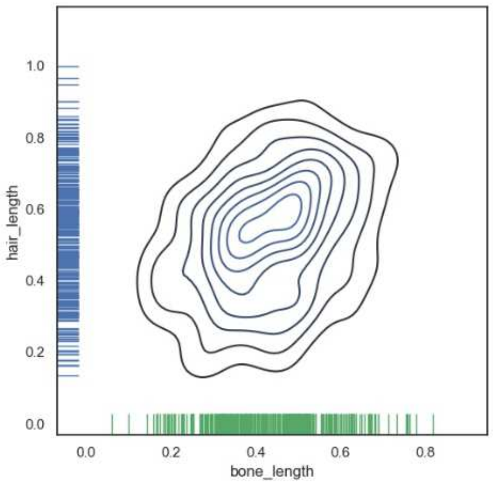

FIGURE 7. Binary Density Function

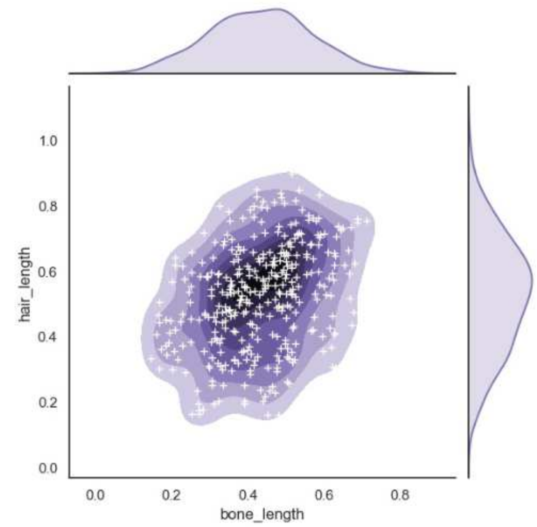

FIGURE 8. Marginal Binary Density Function

Use the algorithm below to calculate the importance of features. The following figure Figure 10 is a histogram ordered by feature importance.

We take the top seven features with higher importance to form a new train data, the rest are discarded.

3. METHODS

There are many machine learning algorithms for classification problem. Choose the following algorithms as the base models of ensemble model,show the most important parameters.

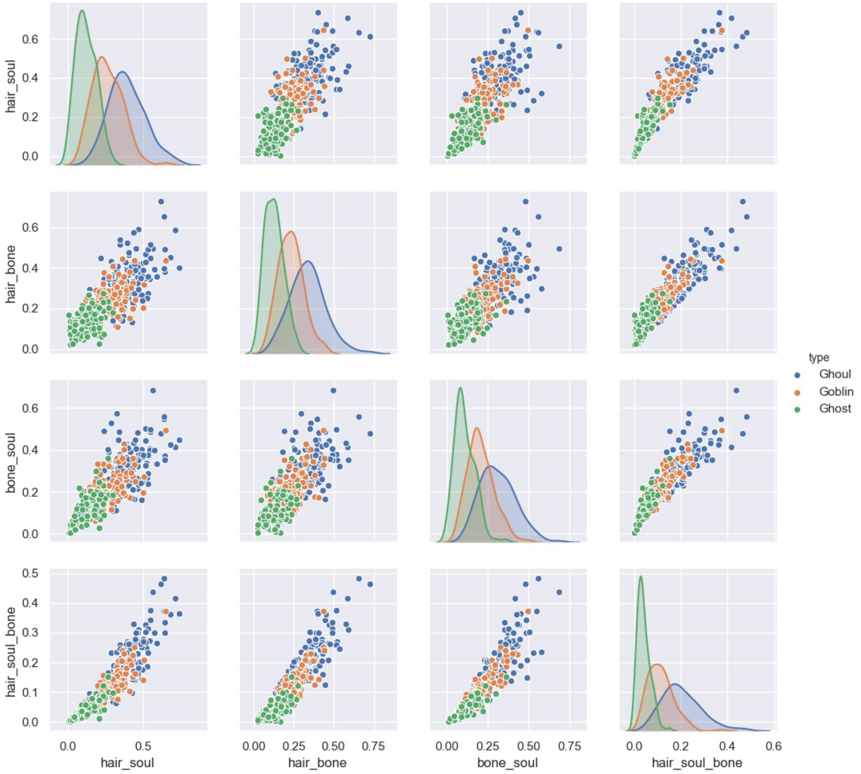

FIGURE 9. New Features Pairplot

Algorithm 1 Features Selection

```
Require: Features $X = \{X_1, X_2, ..., X_n\}$, The number of tree node $M$, $GI_m$ Gini index
    of node $m$, $K$ the number of target, $p_m k$ proportion of target $k$ in node $m$, $VIM_{jm}^{(Gini)}$
    the importance of feature $X_j$ in node $m$ , $n$ the tree number of RF.
Ensure: Variable Importance Measures $VIM_j^{(Gini)}$.
Initialize $GI_m$ , $VIM_j^{(Gini)}$;
for $m \leftarrow 1...M$ do
    for $k \leftarrow 1...K$ do
        Compute the Gini index of node $m$ $GI_m = \sum_{k=1}^{|K|} \sum_{k' \neq k} p_{mk} p_{mk'} = 1 - \sum_{k=1}^{|K|} p_{mk}^2$
    end for
    Divide node m into node r and node l
    Compute the importance of feature $X_j$ in node $m$ $VIM_{jm}^{(Gini)} = GI_m - GI_l - GI_r$
end for
for $i \leftarrow 1...N$ do
    Compute variable importance measures $VIM_j^{(Gini)} = VIM_j^{(Gini)} + VIM_{ij}^{(Gini)}$
end for
return $VIM_j^{(Gini)}$
```

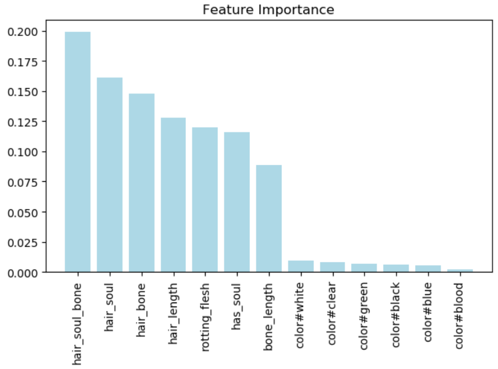

FIGURE 10. Feature Importance

- RandomForest
- LogisticRegression
- SVC
- KNeighbors
- XGBoost
- Netual Network

3.1. **Base Models.**

The base models have many parameters, select the some parameters that have a larger impact on the forecast results, the use Grid Search to find the optimal paratemers set. The following is training result.

3.1.1. *RandomForest.*

Random forest is a classifier with multiple decision trees, and the output is determined by the mode of the individual tree output.

n˙estimators: the number of decision trees
criteriom: criterion of choosing the most appropriate node
max˙depth: The maximum depth of the tree, the default is None
max˙features: The feature that is divided when selecting the optimal attribute cannot exceed this value.

3.1.2. *LogisticRegression.*

Logistic regression is the algorithm that processing a large amount of observation data to obtain mathematical expressions that are in line with the internal laws of things

Penalty: Regular function
C: Regular coefficient

3.1.3. *SVC.*

The basic principle of the SVM algorithm is to find a hyper plane that can distinguish between two types, so that the margin is the largest.

kernel: kernel function
C: penalty factor for error terms
degree: order of polynomial kernel function

3.1.4. *KNeighbors.*

The meaning of the knn algorithm is that enter new data without tags, compare each feature of the new data with each feature in the training set, and select the classification tag with the most similar feature (nearest neighbor: k)

n˙neighbors: number of neighbors to use by default for kneighbors queries
leaf˙size: leaf size passed to BallTree or KDTree
p: power parameter for choosing the distance calculation formula
weights: used in prediction
algorithm: compute the nearest neighbors

3.1.5. *XGBoost.*

XGBoost is to establish K regression trees so that the predicted value of the tree group is as close as possible to the true value (accuracy) and has the greatest generalization ability. From a mathematical point of view, this is a functional optimization, multi-target.

learning˙rate: control the speed of each update
n˙estimators: number of iterations
max˙depth: the depth of tree
gamma: penalty factor
subsample: the proportion of data used in all training sets when training each tree
colsample˙bytree: the proportion of features used in all trees when training each tree

3.1.6. *Netual Network.*

Netual network is widely used in various fields. It generally consists of an Input Layer, a Hidden Layer, and an Output Layer. Each layer is composed of Units.

hidden˙layer˙sizes: The i-th element represents the number of neurons in the i-th hidden layer
activation: activation function
solver: weight optimization solver
learning˙rate: weight update speed

3.2. **Ensemble Model.**

There are many machine learning algorithms, use the above machine learning algorithms as Ensemble Model's base models. Through Grid Search and ten-fold cross-validation to find the optimal parameters.

4. Experiment and Analysis

In the Data Exploration, I has created some new feaures. The following experiment is divided into two parts, one is to use the original train data, the other is to use new features as train data. Grid Search make it easier to determine a set of optimal parameters for these model I adpot. Because the train data is relatively small, a ten-fold cross-validation is used. Then I can test the trained models using test data and analyze the experimental results. And use the trained models as the base models of ensemble model, then do experiment.

4.1. **Base Models Training Result.**

4.1.1. *Original Train Data.*

The following are the best parameters and the Best Score in training of the base models in original train data.

- Best Parameters of Models

  RandomForest: 'criterion': 'entropy', 'max˙depth': 5, 'max˙features': None, 'n˙estimators': 100
  LogisticRegression: 'C': 1, 'penalty': 'l1'
  SVC: 'C': 10, 'degree': 3, 'kernel': 'linear'
  KNeighbors: 'algorithm': 'auto', 'leaf˙size': 10, 'n˙neighbors': 20, 'p': 5, 'weights': 'uniform'
  XGBoost: 'learning˙rate':0.08,'n˙estimators':50, 'max˙depth':5, 'gamma':0,'subsample':0.9,'colsample˙bytree':0.5
  Netual Network: 'activation': 'relu', 'hidden˙layer˙sizes': 9, 'learning˙rate': 'adaptive', 'solver': 'adam'
- Best Score in Ten-Fold Cross-Validation

  From the Table 2, it shows that the accuracy of each model is not much different, except XGBoost.

Table 2. Best Score of the Base Models

	RF	LR	SVC	KNN	XGBoost	Netual Network
Best Score	0.7224	0.7547	0.7493	0.7251	0.9314	0.7466

4.1.2. *New Train Data.*

The following are the base models with the best parameters provided by Grid Search.

- Best Parameters of Models

  RandomForest: 'criterion': 'entropy', 'max˙depth': 5, 'max˙features': 'auto', 'n˙estimators': 100
  LogisticRegression: 'C': 10, 'penalty': 'l2'
  SVC: 'C': 5, 'degree': 3, 'kernel': 'rbf'
  KNeighbors: 'algorithm': 'auto', 'leaf˙size': 10, 'n˙neighbors': 10, 'p': 2, 'weights': 'uniform'
  XGBoost: 'learning˙rate':0.07,'n˙estimators':50, 'max˙depth':6, 'gamma':0,'subsample':0.9,'colsample˙bytree':0.5
  Netual Network: 'activation': 'identity', 'hidden˙layer˙sizes': 8, 'learning˙rate': 'adaptive', 'solver': 'adam'
- Best Score in Ten-Fold Cross-Validation

  Compare with the Table 2 above, you can find that the accuracy of the models increases, although it's not obvious.

Table 3. Best Score of the Base Models

	RF	LR	SVC	KNeighbors	XGBoost	Netual Network
Best Score	0.7224	0.7547	0.7493	0.7251	0.9326	0.7466

4.2. **Forecast Result of Base Models.**

From the table below shows the result of base models on test data. Some models have improved accuracy, but some models have reduced accuracy.

Table 4. Forecast Result of Base Models

	Acc˙old	Acc˙new	Prec˙old	Prec˙new	Rec˙old	Rec˙new	F1˙old	F1˙new
RF	0.81	0.83	0.84	0.84	0.81	0.83	0.82	0.83
LR	0.76	0.75	0.76	0.75	0.76	0.75	0.76	0.75
SVC	0.75	0.69	0.77	0.71	0.75	0.69	0.75	0.70
KNN	0.73	0.71	0.77	0.73	0.73	0.71	0.74	0.71
XGBoost	0.92	0.95	0.92	0.95	0.92	0.95	0.92	0.95
NN	0.73	0.68	0.75	0.71	0.73	0.68	0.74	0.69
mean ± var	.78±.005	.77±.009	.80±.004	.78±.008	.78±.005	.77±.009	.79±.004	.77±.009

4.3. **Forecast Result of Ensemble Model.**

Ensemble model means using more than 1 model to finish the prediction. Here just averaging the prediction results by using voting.

4.3.1. *Original Train Data.*

The table below is the metrics classification report of ensemble model in original train data.

Table 5. Metrics Classification Report of Ensemble Model

	precision	recall	f1-score	support
Ghost	0.80	0.83	0.82	24
Ghoul	0.88	0.79	0.84	29
Goblin	0.67	0.73	0.70	22
micro avg	0.79	0.79	0.79	75
macro avg	0.78	0.78	0.78	75
weighted avg	0.79	0.79	0.79	75

4.3.2. *New Train Data.*

The table below is the metrics classification report of ensemble model in new train data.

Table 6. Metrics Classification Report of Ensemble Model

	precision	recall	f1-score	support
Ghost	0.84	0.88	0.86	24
Ghoul	0.93	0.97	0.95	29
Goblin	0.80	0.73	0.76	22
micro avg	0.87	0.87	0.87	75
macro avg	0.86	0.86	0.86	75
weighted avg	0.86	0.87	0.86	75

5. Conclusion

- Exploratory data analysis is very important for the competition, that is an exploratory analysis of the data to provide the necessary conclusions for data processing and modeling.
- The data that we have, needed processed in many cases. Data preprocessing includes deal with missing data and outliers, change categorical variable into one-hot code and so on.
- The most important thing is feature engineering. We can create as more as poosible features, then select the most useful features.
- Model training is also very important. There are many algorithms, in my opinoin, if the time permits, we can We can try all the algorithms.
- The last thing is adjustment, for example, the models have many parameters, can use Grid Search to find the optimal paratemers.

References

List of Todos

(A. 1) School of Business Administration,, Hunan University, Changsha 410012, China
Email address, A. 1: rongxin_xu@163.com