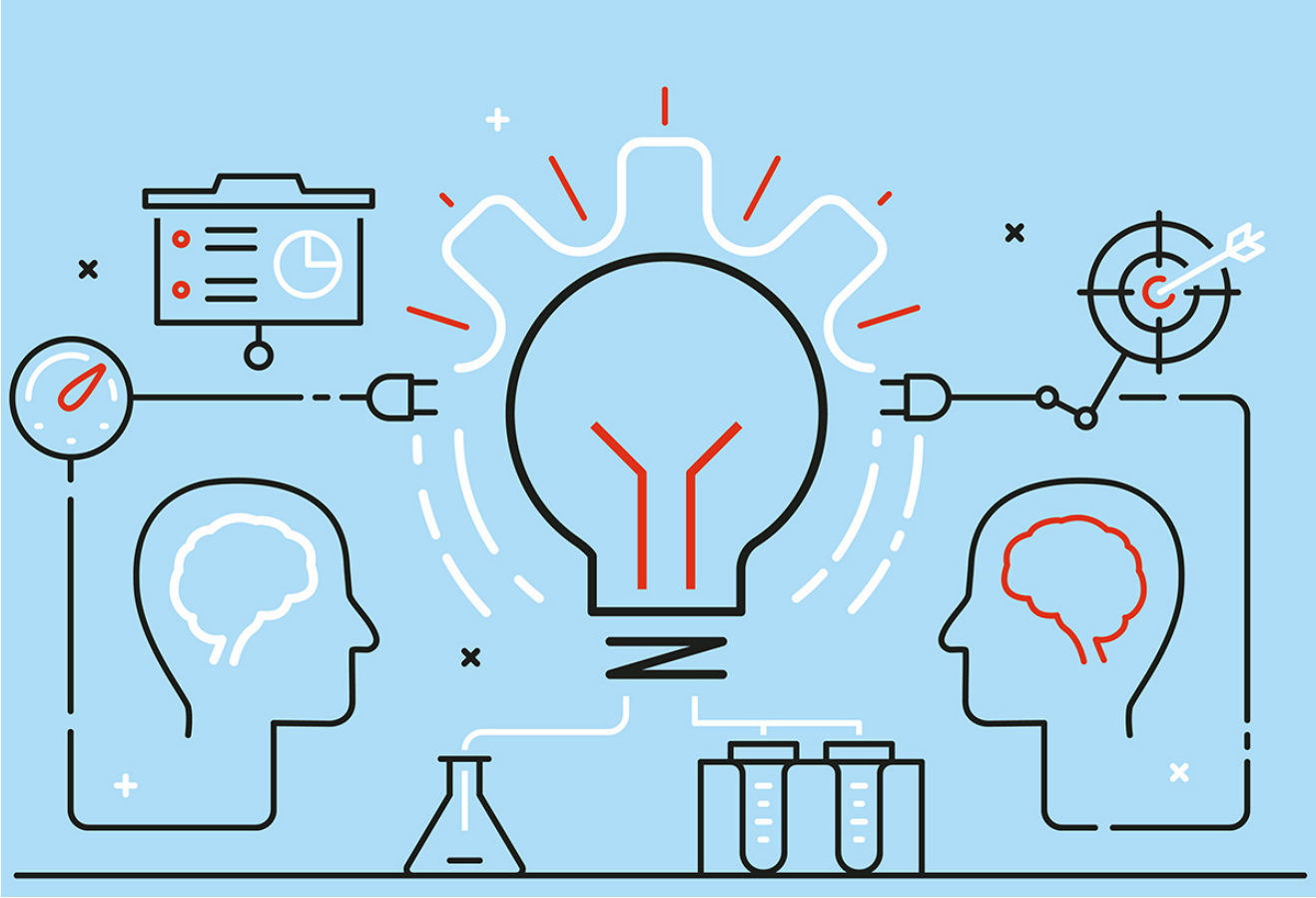
Machine Learning Report

CREATED BY GUO RUI _1630013011

2018/11/30



PART 1. The workflow.

PART 2. The model.

PART 3. The error analysis and optimization.

Part one

The Workflow

1.1. Prepare data.

I must have access to a large set of training data that includes the feature that I want to be able to infer (predict) based on the other features. But, fortunately my teacher did this step for us, we can go directly to the next step.

1.2. Data analysis.

When I have sourced your data, I analyze and understand the data and prepare it to be the input to the training process.

1.3 Visualize the data.

```
In [74]: #Get the attribute infomation
         data.head(10)
```

Out[74]:

	Attribute A	Attribute B	Attribute C	Attribute D	Attribute E	Attribute F	Attribute G	Attribute H	Attribute I	Attribute J	Attribute K	Ranking
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
5	7.4	0.66	0.00	1.8	0.075	13.0	40.0	0.9978	3.51	0.56	9.4	5
6	7.9	0.60	0.06	1.6	0.069	15.0	59.0	0.9964	3.30	0.46	9.4	5
7	7.3	0.65	0.00	1.2	0.065	15.0	21.0	0.9946	3.39	0.47	10.0	7
8	7.8	0.58	0.02	2.0	0.073	9.0	18.0	0.9968	3.36	0.57	9.5	7
9	7.5	0.50	0.36	6.1	0.071	17.0	102.0	0.9978	3.35	0.80	10.5	5

figure 1.1: Show all features of data

```
In [75]: #Get all data analysis
         data.describe(include='all')
```

Out[75]:

	Attribute A	Attribute B	Attribute C	Attribute D	Attribute E	Attribute F	Attribute G	Attribute H	Attribute I	Attribute J	Attribute K	Rar
count	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.000000	1499.00
mean	8.367111	0.523793	0.274890	2.556204	0.086815	15.915277	45.253502	0.996764	3.309473	0.658659	10.457205	5.66
std	1.770966	0.177666	0.194764	1.429167	0.044412	10.600786	32.132602	0.001922	0.154219	0.165347	1.063330	0.81
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.860000	0.370000	8.400000	3.00
25%	7.100000	0.390000	0.100000	1.900000	0.070000	7.000000	21.500000	0.995570	3.210000	0.550000	9.500000	5.00
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	37.000000	0.996750	3.310000	0.620000	10.200000	6.00
75%	9.300000	0.640000	0.430000	2.600000	0.090500	22.000000	60.000000	0.997900	3.400000	0.730000	11.100000	6.00
max	15.900000	1.580000	0.790000	15.500000	0.611000	72.000000	289.000000	1.003690	4.010000	1.980000	14.900000	8.00

figure 1.2: Show all data analysis

It is clearly to see that all features' information, such as the number of train data, mean, 25% score, and max value.

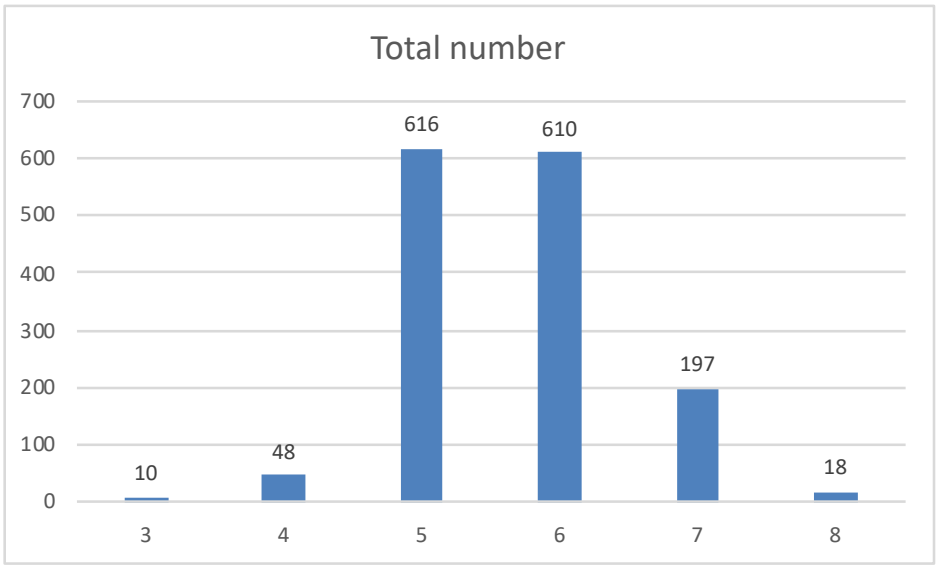
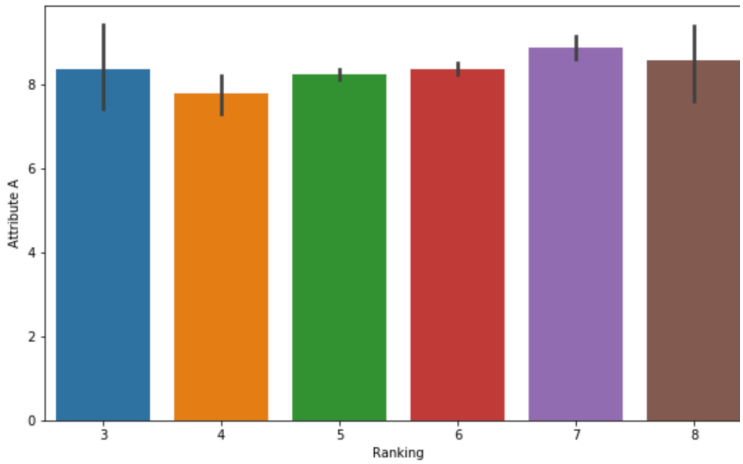


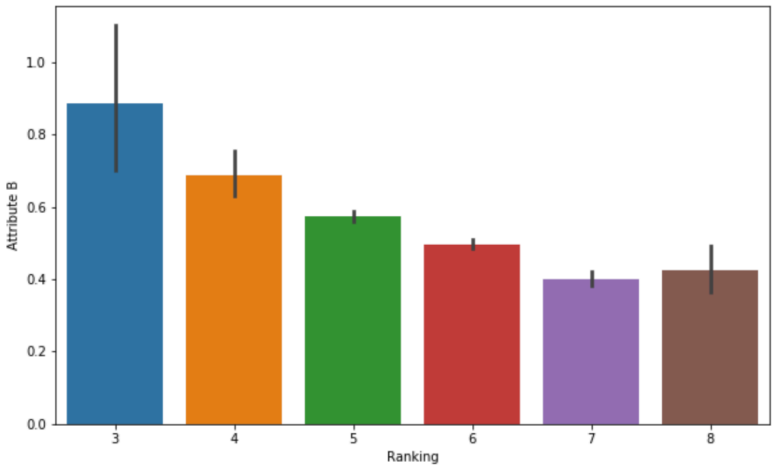
figure 1.3: Show the distribution of Ranking

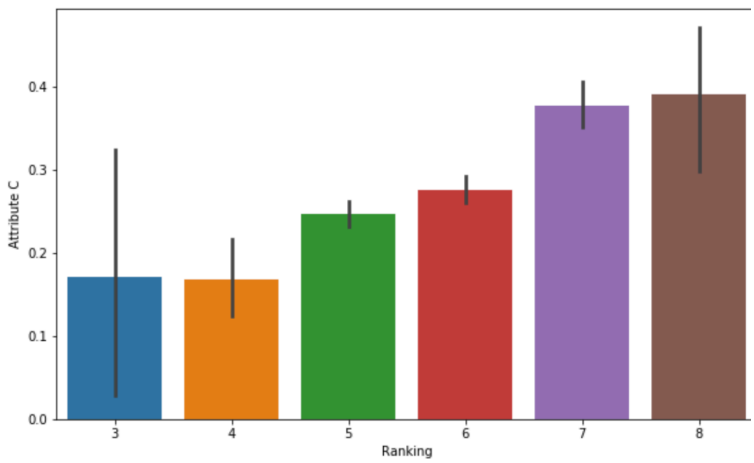
According to this bar chart, we can see most of Ranking distribution are 5 or 6. What I mean that we should pay attention to the accuracy of 5 and 6.

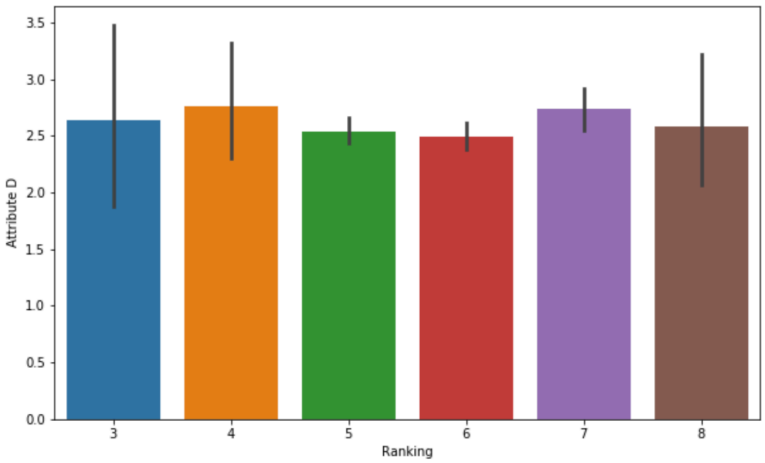
1.4 Draw graph.

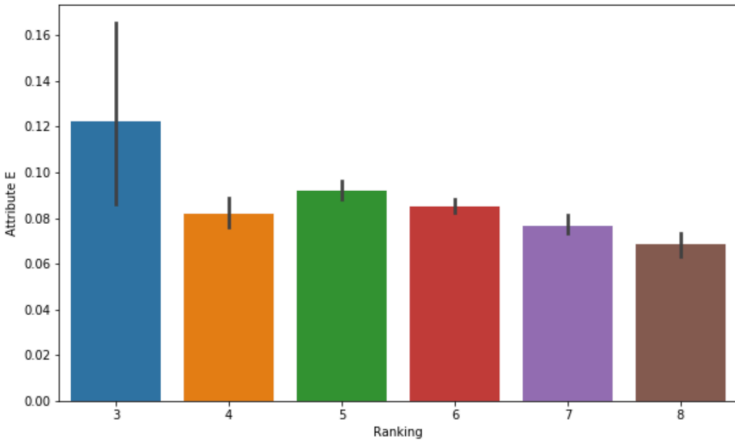
The following figures show the relationship of each variable with Ranking.

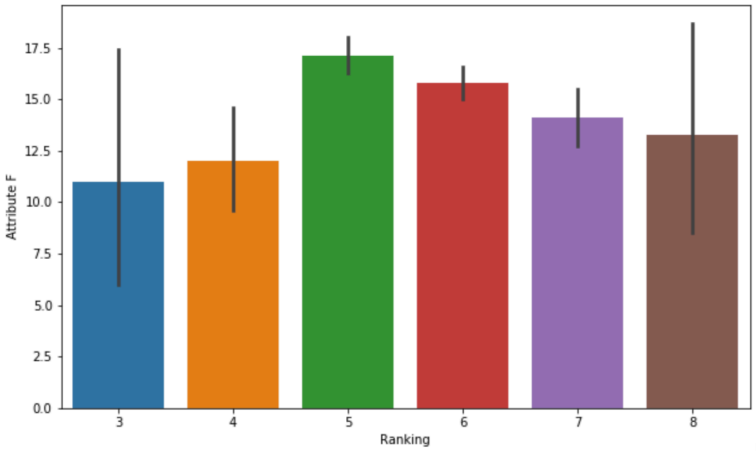


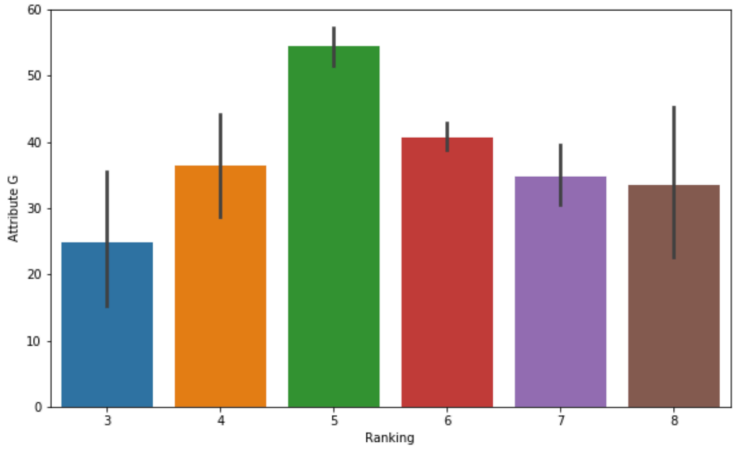


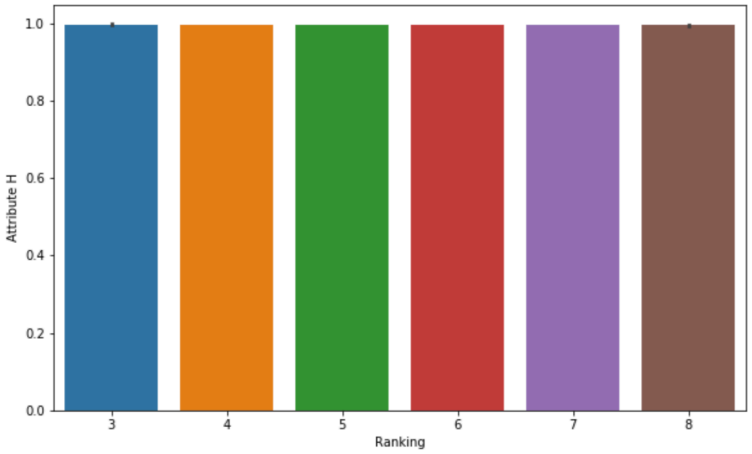


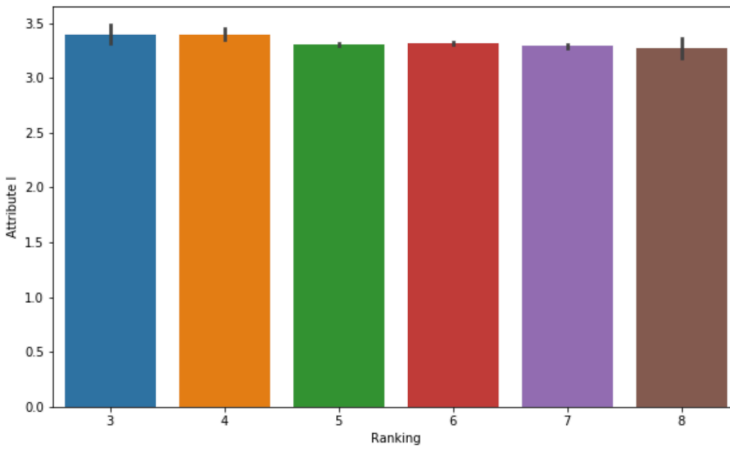


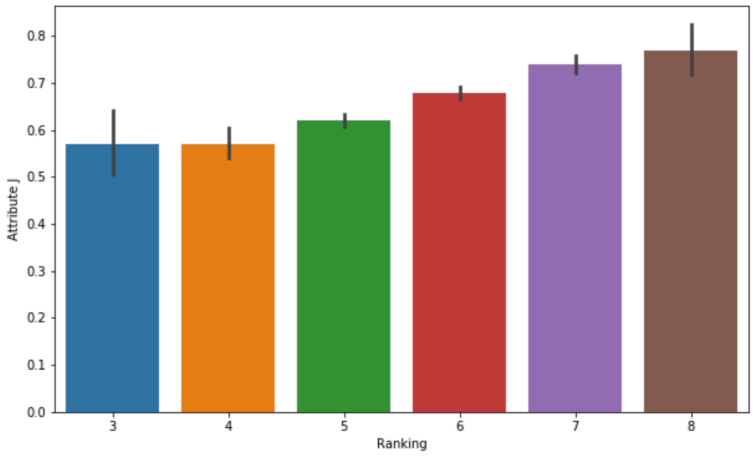


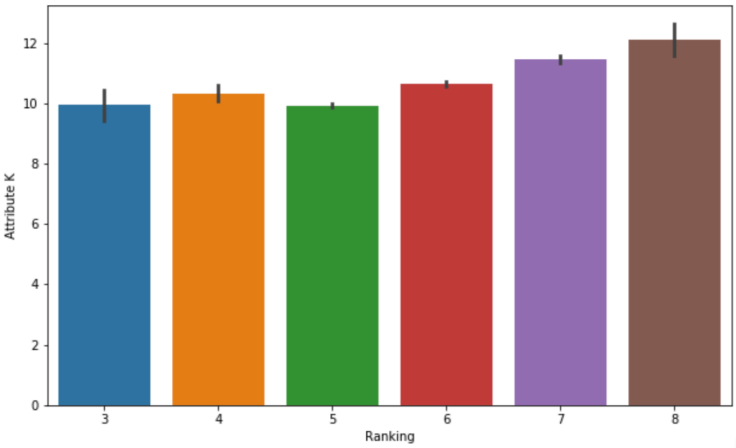












1.5 Data analysis.
According to these figures, it can clearly be seen that how each feature influence the ranking.
For example, B and C have the most impact for Ranking and have an opposite influence for Ranking. The attribute H and I have little impact for Ranking. I think these figures are very important to find relationships between each feature and Ranking, and also for finding the best learning model.

Part two

The model analysis

Focus our analysis, this is a classification problem, and also is a supervised learning. After training, I should predict the ranking from test set. Thus, we can consider the following model.

2.1. Linear regression algorithm.
The basic assumption is that the output variable can be expressed as a linear combination of a set of input variable. To avoid overfitting, regularization technique (L1 and L2) is used to penalize large value of w1, w2.
The strength of Linear model is that it has very high performance in both scoring and learning. The Stochastic gradient descent-based learning algorithm is highly scalable and can handle incremental learning.
The weakness of linear model is linear assumption of input features, which is often false. Therefore, I have to give up this model.

2.2 K-nearest neighbors algorithm.
KNN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. Both for classification and regression, a useful technique can be used to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. A peculiarity of the k-NN algorithm is that it is sensitive to the local structure of the data.

The follow chart is KNN classifier after training.

```
This is KNN Classifier result:
             precision    recall  f1-score   support

          3       0.00      0.00      0.00         1
          4       0.20      0.07      0.11        14
          5       0.80      0.67      0.73       140
          6       0.55      0.66      0.60       103
          7       0.52      0.63      0.57        41
          8       0.00      0.00      0.00         1

avg / total       0.65      0.63      0.63       300
```

The accuracy is about 0.65 for 300 test set.

2.3 Random forests algorithm.

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The follow chart is the result of KNN classifier after training.

```
             precision    recall  f1-score   support

          3       0.00      0.00      0.00         1
          4       0.00      0.00      0.00        14
          5       0.83      0.81      0.82       140
          6       0.63      0.80      0.70       103
          7       0.77      0.56      0.65        41
          8       0.00      0.00      0.00         1

avg / total       0.71      0.73      0.71       300
```

The accuracy is about 0.71 for 300 test set, but it is unstable.

2.4 Stochastic gradient descent

Stochastic gradient descent also known as incremental gradient descent, is an iterative method for optimizing a differentiable objective function, a stochastic approximation of gradient descent optimization.

The follow chart is the result of SGD classifier after training.

	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.50	0.07	0.12	14
5	0.69	0.79	0.74	140
6	0.48	0.60	0.54	103
7	0.11	0.02	0.04	41
8	0.00	0.00	0.00	1
avg / total	0.53	0.58	0.54	300

The accuracy is about 0.53 for 300 test set, but it is unstable.

2.5 Support Vector Machine algorithm

In machine learning, support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

The follow chart is the result of SVM classifier after training.

	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	14
5	0.77	0.74	0.76	140
6	0.55	0.79	0.65	103
7	0.83	0.37	0.51	41
8	0.00	0.00	0.00	1
avg / total	0.66	0.67	0.64	300

The accuracy is about 0.66 for 300 test set. I think it is the best model I found. I will try to optimize this model for higher accuracy.

Part three

The error analysis and optimization

3.1 The error analysis.

According to our data set, we have 11 features. Between these features, it is very difficult to distinguish their differences because they are too similar. And, this is a classification problem with 6 categories, which mean that it is difficult to get high prediction accuracy. Although I try my best to find the best parameters and increase accuracy, the result is not very good. As the result, I give up the previous models, and just focus on SVM.

3.2 Optimize SVM.

Grid search is the traditional way of performing hyperparameter optimization has been grid search, or a parameter sweep, which is simply an exhaustive searching through a manually specified subset of the hyperparameter space of a learning algorithm.

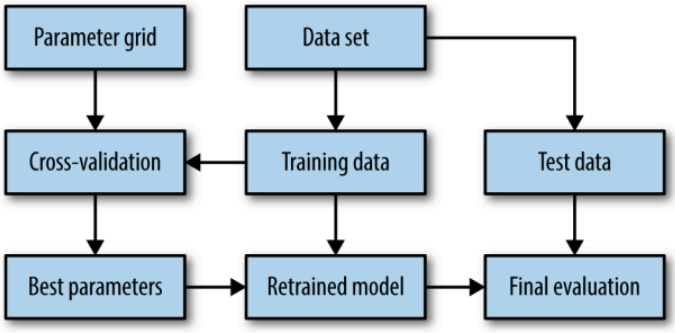


Figure 5-7. Overview of the process of parameter selection and model evaluation with GridSearchCV

In order to find best parameters to optimize the cost function of SVM, I set the following parameters.

```
param = {
    'C': [0.001,0.01,0.1,1,10,100],
    'kernel':['linear', 'rbf'],
    'gamma' :[0.001,0.01,0.1,1,10,100]
}
```

After using Grid Search and Cross-validation, I got the best parameters. Hence on given parameters tuning the SVM with rbf kernel, the accuracy increases from 0.66 to 0.72 using cross validation score. Obviously, this is a dramatic increase.

```
This is SVM2 Classifier result:
             precision    recall  f1-score   support

          3       0.00      0.00      0.00         1
          4       0.00      0.00      0.00        14
          5       0.92      0.42      0.58       140
          6       0.45      0.98      0.61       103
          7       1.00      0.24      0.39        41
          8       0.00      0.00      0.00         1

avg / total       0.72      0.57      0.53       300
```

In conclusion, we can see SVM(with best parameters) has the best accuracy for predicting our ranking. And also, I will use this model to predict your test set. I have filled result in your excel file.