**CMPT 459 Data Mining Milestone 3**

Group Name: **hello\_CKZ**

Github Link: <https://github.com/llinvincible9/hello_CKZ/tree/master/milestone3>

|  |  |
| --- | --- |
| Bagging | **Yu Ke**, keyuk |
| Random Forest |  |
| Neural Network |  |

4.

**Bagging:**

**Performance on validation dataset and testing dataset:**Bagging Classifier is an ensemble classifier, so there are many parameters. And the first parameter is to choose a **base\_estimator, which is to pick a normal classifier. At first, I used the defult parameters, which was decision tree, however, the result is not good. The accuracy** of the 10 fold validation **is about 0.75. And the score I submitted to the website was 2.12, which was a bad result.**

**10 Fold Cross-Validation**

|  |
| --- |
| **scores.Mean()= 0.7091983122362869**  **0.7181434599156118, 0.7139240506329114, 0.7052742616033755, 0.6985232067510548, 0.7135021097046413, 0.7008438818565401, 0.7183544303797469, 0.7151898734177216, 0.7059071729957805, 0.7023206751054852** |

**Comments:**The reason why the first version is not quite good is because in Bagging Classifier, I did not change the defult parameters. The code is like this:

|  |
| --- |
| Clf=BaggingClassifier().fit(X, Y) |

5.

Bagging:

In order to improve my bagging classifier, I tested and changed several parameters in function BaggingClassifier().

**Step1:** In stead of using the defult base\_estimator, I picked another classifier: K-Nearest Neighbors. The code is like this:

|  |
| --- |
| clf = BaggingClassifier(KNeighborsClassifier(n\_neighbors=5).fit(X, Y) |

When using the KN classifier in Bagging classifier, we need to choose the n\_neighbors in function *KNeighborsClassifier(),* so I used a loop to check which was better for n\_neighbors=5, 7, 9, 11, 13.

From the testing, I found n\_neighbors=9 came to the best result.

**Step2:**

The next step is to modify parameters like ***n\_estimators, max\_samples and max\_features***. From my testing, I found that these 3 parameters did huge significance to the result.

For ***n\_estimators,*** when it became larger, the result became better.

For ***max\_samples and max\_features,*** 0.5 is better than defult value 1.0.

By doing the above modification, the score changed from **2.12 to 0.67.**

The final code is like this:

|  |
| --- |
| best\_clf=BaggingClassifier( KNeighborsClassifier(n\_neighbors=9) ,n\_estimators=100,max\_samples=0.5, max\_features=0.5) |

7. Performance

Bagging:

Kaggle score: 0.67

In Bagging Classifier, I reducd overfitting contributes most to the gains and added more features also makes better accuracy. Besides, by using 10-Flod cross validation, I can pick the best classifier, which also reduced the overfitting.

8. Comparison

SVM vs Bagging:

Compared to SVM, Bagging Classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions to form a final prediction.

We can use different base\_estimator such as decision tree, K Nearest Neighbors and so on to perform to classifier.

Besides, when using the SVM in milestone2, it is really time consuming, and every fit() would take a long time. However, when using the Bagging Classifier, the running speed is a lot faster. And by using proper parameters, the result is better than SVM.

Gains:

Kaggle score: Milestone2-SVM: **0.77** to Milestone2- Bagging Classifier: **0.67**