**Description**

This competition's dataset provides nearly 12 years of crime reports from across all of San Francisco's neighborhoods. Given time and location and we will predict the category of crime that occurred noting that Multi-class log loss is used for evaluation.

**Results**

In the below table we will state the logloss value for each classifier noting that code for each classifier is provided in **San Fransicso Crime Classification** **notebook** without redundant coding related to the same classifier. For example the notebook includes Random Forest classification with number of trees = 50 but numTrees = 100 isn’t included as it is just slight code change in RandomForestClassifier command. The goal of avoiding redundancy is to provide readable and concise notebook.

|  |  |
| --- | --- |
| Classifier | Logloss |
| Logistic Regression(RegParam=0.01) | 2.64309 |
| Logistic Regression(RegParam=0.1) | 2.65954 |
| Logistic Regression(RegParam=0.001) | 2.64929 |
| Random Forest(numTrees=50) | **2.59053** |
| Random Forest(numTrees=100) | 2.59121 |
| Random Forest(numTrees=200) | 2.59105 |
| Decision Tree(Max Depth=5) | 2.60944 |
| Decision Tree(Max Depth=10) | 2.61326 |
| Decision Tree(Max Depth=15) | 4.79418 |
| Naïve Bayes(smoothing=1) | 2.61186 |

**Conclusion**

Random Forest with 50 trees is the best classifier with logloss of **2.59053**. Usually Bagging algorithms like Random Forest reduce variance and logloss of test data. However; increasing number of trees may cause overfitting and that’s why when we increased number of random forest trees logloss somehow increased.

Also it seems that data isn’t linearly separable so Logistic regression doesn’t perform very well. 