Name: Nooreldean Koteb

Miner Username: IcyEagle

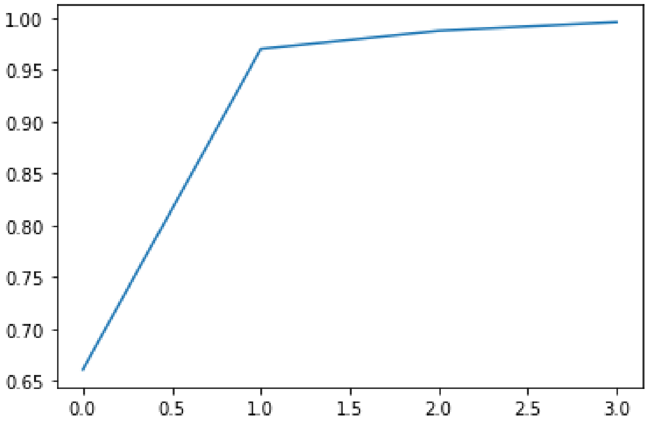
Rank: 16 **Accuracy:** 82%

Approach

Approaching this problem, I began by trying to understand the data and how it was imbalanced. Using pandas, I created a data frame and put the data into bins based on their final column value: [7083, 5296, 894, 513, 435, 238, 69].

Feature Selection:

To deal with the imbalance I began with resampling the data. In my preprocessing function I gave the ability to resample all labels based on the min, max, or a specified number. I then scaled the data using the Standard Scaler function and gave the option for data reduction.

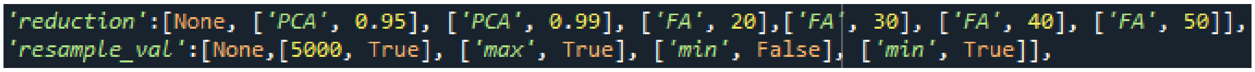


The first data reduction option was PCA. The values I generally used while experimenting was 0.95 and 0.99 of the explained variance. This significantly cut down my features to 1 - 4 features.

Another data reduction option I used was Factor Analysis. This data reduction function is different as it extracts from all features the maximum common variance and makes a common score. For this reduction method I generally entered 20, 40, and 50 as my n_components.

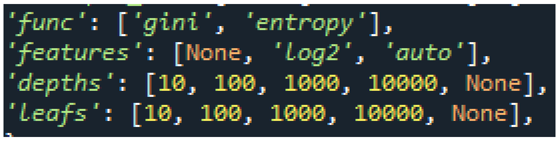
Methodology:

Decision Trees and Naïve Bayes classifiers cross validation were run only once. While KNN and Random Forest were run multiple times with more focused parameters every time. All classifiers were tested using various feature selection techniques.



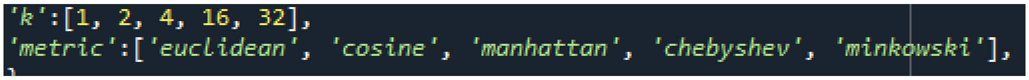
```
'reduction':[None, ['PCA', 0.95], ['PCA', 0.99], ['FA', 20],['FA', 30], ['FA', 40], ['FA', 50]],
'resample_val':[None,[5000, True], ['max', True], ['min', False], ['min', True]],
```

Naïve Bayes did not have any additional parameters past these feature selection parameters. Decision Tree had further tuning with the parameters below:



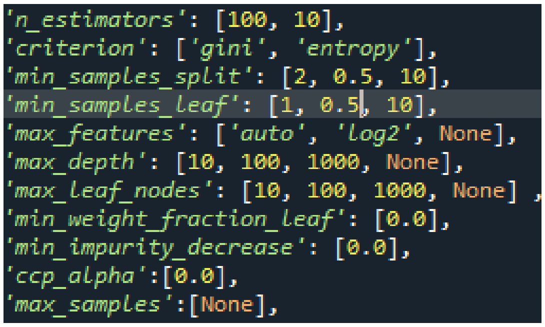
```
'func': ['gini', 'entropy'],
'features': [None, 'log2', 'auto'],
'depths': [10, 100, 1000, 10000, None],
'leafs': [10, 100, 1000, 10000, None],
```

The KNN classifer was run twice Once with metrics set to euclidean and cosine only. A deeper cross validation was then run with these parameters in addition to the feature selection parameters.



```
'k':[1, 2, 4, 16, 32],
'metric':['euclidean', 'cosine', 'manhattan', 'chebyshev', 'minkowski'],
```

Finally, the Random Forest classifier was run 4 times. In addition to the feature selection parameters, these parameters were tested during cross validation.



```
'n_estimators': [100, 10],
'criterion': ['gini', 'entropy'],
'min_samples_split': [2, 0.5, 10],
'min_samples_leaf': [1, 0.5, 10],
'max_features': ['auto', 'log2', None],
'max_depth': [10, 100, 1000, None],
'max_leaf_nodes': [10, 100, 1000, None] ,
'min_weight_fraction_leaf': [0.0],
'min_impurity_decrease': [0.0],
'ccp_alpha':[0.0],
'max_samples':[None],
```

Max_leaf_nodes did not seem to have much of an effect, while max_depth had very minimal effect. The model consistently did better with min_samples_leaf set to 1 and min_samples_split set to 2. Max_features did slightly better under log2 and criterion under gini. n_estimators at 100 performed the best. On the next 2 runs different n_estimators, max_depth, and max_leaf_nodes were attempted, however not much improvement occurred. On the 4th attempt max_samples was changed to try [0.1, 0.5, 0.75, None]. However, None remained the best. The best result achieved from Random Forest was 98.01% on the 4th attempt with these parameters.

Best Parameters: {'reduction': ['FA', 50], 'resample_val': ['max', True], 'n_estimators': 100, 'criterion': 'gini', 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 'log2', 'max_depth': None, 'max_leaf_nodes': None, 'min_weight_fraction_leaf': 0.0, 'min_impurity_decrease': 0.0, 'ccp_alpha': 0.0, 'max_samples': None}

In general, for all classifiers Factor Analysis at n_components= 50 and resampling set to max performed best.

Classifier Development:

The cross-validation function used for this project was imported and modified from the HW1 project. All classifiers were tested on a 70-30 train test split. KNN was the most successful classifier at first giving me 97.57%. Naïve Bayes did not have any parameters, however different data techniques were used during cross validation. Naïve Bayes highest result was 65.72%. Decision Tree did fairly well with an Accuracy of 96.54%. Random Forest Classifier was the best preforming with an Accuracy of 98.01%.

Random Forest Classifier:

Random Forest creates a collection of Decision Trees then averages their results/predictions using the outputs from every individual trees. This is much better than a normal Decision Tree because it cuts down on overfitting, however it is slower.

Classifier	Best Cross-validation	Miner submission
Decision Tree	96.54%	73%
Random Forest	98.01%	82%

Although Miner results are only 50% of the file, I think this is still a fair comparison between both classifiers. For more cross-validation results check out the folder “Cross_val Results”. This folder contains the output of all tests run.

ANN Classifier:

I had implemented part of an ANN and was planning on using it as a fifth classifier, however due to time I did not fully complete/integrate it with the cross-validation and final predict function. I believe an ANN might have preformed better with dropout since it may have done better with overfitting.

Results

Running Times:

Although Times varied from cross validation run to another due to parameter dimensions and available computational power at the time of the run. there was a general speed that all classifiers ran at. Random Forest took longer than KNN However it had many more parameters to go through, that is why it is ranked above KNN. Random Forest Would have been faster if given the same parameter dimensions as KNN.

Rank	Classifier	Time
1	Naïve Bayes	~2 minutes
2	Decision Trees	~5 minutes
3	Random Forest	~(60 – 120) minutes
4	KNN	~(30 – 60) minutes

Results:

These are the best results achieved during cross validation sessions for all 4 classifiers. This however does not reflect their results on the given final test set.

Rank	Classifier	Accuracy
1	Random Forest	98.01%
2	KNN	97.57%
3	Decision Tree	96.54%
4	Naïve Bayes	65.72%

Although Decision Tree got a pretty high score during cross validation it does not compare as well to KNN and Random Forest on the final results. Decision Trees gave me 71%-73% on Miner submissions, and although Miner results are a random 50% of the file, I believe overfitting was high on the Decision Tree classifier compared to Random Forest and KNN.

For more cross validation results please check out the “Cross_val Results” directory. Every text file there is a sorted list of tests containing the accuracy, F1_score, confusion matrix, and parameters for each model tested during cross validation.