CS412 HW4 Report

Yiwei Zhuang

Brief introduction of the classification methods

For load data, I write a LoadData.py to handle all the data. It is a dictionary which has two keys, one is label and another is feature. I use these two keys because I do not want train the test data with its label so it is easier for me to distinguish between label and feature.

For my decision tree functions, I write 2 separate file. DecisionTree.py is used for auto grader and it has some scripts. In DecisionTreeFunctions.py, it has three parts. The first part is to calculate the gini index and gini index for some specific attribute. The second part is some data selection functions that can prune the original data and select specific data with specific feature value or label value. The third part is the decision tree creation and classification. For creation, it will output a dictionary to represent a tree and the classification will input a tree with some data and output its prediction.

For my random forest functions. Still two files. In RandomForestFunction.py, it will first bootstrap the input data and create input value of number of trees, default = 50, then do the majority vote classify. For its decision tree in forest, it will also choose a subset of features randomly to spilt the node.

All model evaluation

I write a script evaluation.py to calculate all the evaluation data.

Note: If a value is -1, then it is null value because it divides 0

For test data:

Decision Tree Evaluation

	Accuracy	Sensitivity	Specificity	Precision	F-1 Score	F beta 0.5	F beta 2
balance-scale	0.587	[0 , 0.637, 0.663]	[6.94, 2.094, 1.547]	[0, 0.67, 0.614]	[0, 0.65, 0.638]	[0, 0.663, 0.624]	[0, 0.644, 0.653]
led	0.858	0.78	0.892	[0.765, 0.9]	[0.773, 0.897]	[0.768, 0.899]	[0.778, 0.894]
nursery	0.972	[0.962, 0.692, 0.976, 1.0, -1]	[44.3, 107.9, 190., -1, 946.0]	[0.955, 0.677, 0.988, 1, 0]	[0.959, 0.684, 0.982, 1, 0.]	[0.957, 0.68l, 0.987, 1, 0.]	[0.961, 0.690, 0.978, 1. 0.]
poker	0.614	0.756	0.315	[0.698, 0.381]	[0.726, 0.345]	[0.709, 0.366]	[0.744, 0.326]

Random Forest Evaluation

	Accuracy	Sensitivity	Specificity	Precision	F-1 Score	F beta 0.5	F beta 2
balance-scale	0.79	['0.00', '0.88', '0.87']	['-1.00', '3.14', '4.74']	[0. 0.76271 0.82242]	[0. 0.79262673 0.79047619]	[0. 0.76512456 0.77281192]	[0. 0.82217973 0.80896686]
led	0.86	0.589	0.947	[0.7, 0.9]	[0.7744, 0.898]	[0.77, 0.899]	[0.77 0.897]
nursery	0.96	['0.95', '0.15', '0.86', '0.62']	['3.75', '961.50', '7.51', '81.91']	[0.701, 0.826, 0.797, 0.96]	[0.81, 0.248, 0.828, 0.756]	[0.747, 0.428, 0.81, 0.87]	[0.89, 0.175, 0.849, 0.669]
poker	0.68	0.995642701525	0.0182	[0.68005952 0.66666667]	[0.80813439 0.03555556]	[0.72608834 0.08230453]	[0.91108453 0.02267574]

For training data:
Decision tree:

```
train file is:    balance-scale.train
test    file is:    balance-scale.train

Accuracy 1.0

Sensitivity [1.0, 1.0, 1.0]

Specificity [-1, -1, -1]

Precision [ 1.    1.    1.]

F-1 Score [ 1.    1.    1.]

F beta    0.5 [ 1.    1.    1.]

F beta    2 [ 1.    1.    1.]



train file is:    led.train.new
test    file is:    led.train.new

Accuracy 0.859607091519

Sensitivity 0.777429467085

Specificity 0.895790200138

Precision [ 0.76661515    0.90138889]

F-1 Score [ 0.77198444    0.89858082]

F beta    0.5 [ 0.76875387    0.90026356]

F beta    2 [ 0.77524226    0.89690437]



train file is:    nursery.data.train
test    file is:    nursery.data.train

Accuracy 1.0
```

```
Sensitivity [1.0, 1.0, 1.0, 1.0, 1.0]

Specificity [-1, -1, -1, -1, -1]

Precision [ 1.    1.    1.    1.    1.]

F-1 Score [ 1.    1.    1.    1.    1.]

F beta    0.5 [ 1.    1.    1.    1.    1.]

F beta    2 [ 1.    1.    1.    1.    1.]


train file is:    poker.train
test    file is:    poker.train

Accuracy 1.0

Sensitivity 1.0

Specificity 1.0

Precision [ 1.    1.]

F-1 Score [ 1.    1.]

F beta    0.5 [ 1.    1.]

F beta    2 [ 1.    1.]


random forest:
train file is:    balance-scale.train
test    file is:    balance-scale.train


Accuracy 0.94

Sensitivity ['0.07', '1.00', '0.99']

Specificity ['-1.00', '15.67', '13.43']
```

```
Precision [ 1.          0.93939394  0.93      ]

F-1 Score [ 0.13793103  0.96875     0.96124031]

F beta  0.5 [ 0.28571429  0.95092025  0.94224924]

F beta  2 [ 0.09090909  0.98726115  0.98101266]


train file is:  led.train.new
test  file is:  led.train.new

Accuracy 0.82

Sensitivity 0.526645768025

Specificity 0.954451345756

Precision [ 0.8358209   0.82077151]

F-1 Score [ 0.64615385  0.88257817]

F beta  0.5 [ 0.74799644  0.84442545]

F beta  2 [ 0.56872038  0.92434167]


train file is:  nursery.data.train
test  file is:  nursery.data.train

Accuracy 0.95

Sensitivity ['0.99', '0.00', '0.92', '1.00', '0.00']

Specificity ['12.59', '-1.00', '144.54', '-1.00', '-1.00']

Precision [ 0.86840369  0.          0.98422847  1.          0.        ]

F-1 Score [ 0.9235356   0.          0.9513803   1.          0.        ]

F beta  0.5 [ 0.88964725  0.          0.97082072  1.          0.        ]
```

```
F beta  2 [ 0.96010793  0.            0.93270318  1.            0.          ]


train file is:  poker.train
test  file is:  poker.train
RandomForest

Accuracy 0.97

Sensitivity 1.0

Specificity 0.891156462585

Precision [ 0.95892169  1.          ]

F-1 Score [ 0.97903014  0.94244604]

F beta  0.5 [ 0.96686513  0.97615499]

F beta  2 [ 0.99150518  0.91098748]
```

Parameters tuning and reasons

For decision tree part, I set a input for the limitation of the tree depth to prevent over fitting. Usually, I let the tree grow the full depth. I tested some parameters and find if I always let the tree grow full size then the training data accuracy will 100. For test data, still a full size tree will be better.

For random forest, the first parameter I choose is the size of random forest, which is the number of decision tree in the forest. Originally, I only choose the size 5 and I find the accuracy is not stable and often changes. Sometime it is better than a single tree sometimes not. Then I choose size 200 but it is a little bit slow and I think we have a time limit for auto grader. So I choose the size 50 and I find the accuracy is stable. Another parameter for selecting the input data for each tree, I use a random function to generate some random number to choose each row as the input data. For the number of selecting attribute when we have to randomize the feature in a single tree, still I use the random number between 1 and original size and then do the gini calculation.

Conclusion

In general, if we can choose a proper size of random forest, we can improve the accuracy especially for those accuracy below 70 when we only use a single decision tree. For an accuracy above 90 when only using a single tree, the effect of random forest is not very obvious.