CS 412 Homework 4 Report

Bo Wang

University of Illinois at Urbana-Champaign

December 9, 2018

1 Overall accuracy on the test dataset:

Accuracy		
	Decision Tree	Random Forest
balance.scale	0.725	0.719
nurserys	0.991	0.986
led	0.860	0.832
synthetic.social	0.477	0.590

2 Parameter settings:

For decision tree method, I chose to use binary split. Since all data sets contains only categorical attributes, so I set the split by "whether the value of this attribute equals my value". To achieve this goal, I create a class called Question, which has two variables, attribute and value. I did not set the maximum depth of decision tree. Instead, the tree brach will stop splitting if there is no further information gain.
For random forest, I set the number of tree to be 8. I also tried larger numbers, such as 20, 30, 50, 80. However, those number did not yield a better prediction for led, balance scale and nursery. (the result for those three data set was higher than the threshold, but the improvement are not significant). Moreover, 80 trees even lowered the performance. I chose 8 mainly because it can generate best prediction within 30 seconds.

3 Accuracy on the training dataset:

Accuracy		
	Decision Tree	Random Forest
balance.scale	1.0	0.948
nurserys	1.0	0.996
led	0.850	0.851
synthetic.social	1.0	0.929

4 F1 Sores of decision tree TESTS:

balance.scale	
Class label	F1 score
1	0.821
2	0.838
3	0

Led	
Class label	F1 score
1	0.802
2	0.876

Nurserys	
Class label	F1 score
1	0.964
2	0.932
3	0.864
4	0.890

Synthetic.social	
Class label	F1 score
1	0.489
2	0.473
3	0.506
4	0.498

5 F1 Sores of Random Forest TESTS:

balance.scale	
Class label	F1 score
1	0.817
2	0.829
3	0

Led	
Class label	F1 score
1	0.891
2	0.873

Nurserys	
Class label	F1 score
1	0.987
2	1.0
3	0.992
4	0.940

Synthetic.social	
Class label	F1 score
1	0.535
2	0.528
3	0.488
4	0.510

6 F1 Sores of decision tree TRAININGS:

balance.scale	
Class label	F1 score
1	1.0
2	1.0
3	1.0

Nurserys	
Class label	F1 score
1	1.0
2	1.0
3	1.0
4	1.0

Led	
Class label	F1 score
1	0.847
2	0.912

Synthetic.social	
Class label	F1 score
1	0.542
2	0.489
3	0.490
4	0.521

7 F1 Sores of Random Forest TRAININGS:

balance.scale	
Class label	F1 score
1	1.0
2	1.0
3	1.0

Nurserys	
Class label	F1 score
1	1.0
2	1.0
3	1.0
4	1.0

Led	
Class label	F1 score
1	0.834
2	0.925

Synthetic.social	
Class label	F1 score
1	0.658
2	0.628
3	0.672
4	0.632

8 Conclusion:

The ensemble method (random forest) improved the performance of my decision tree classification method. Especially for the last data set, synthetic social, the accuracy was boosted over 20 percent. However, the improvement for the first three data sets was not as good as I expected. To improve the performance of those three data sets, I changed the number of trees (i.e. divide the training sets into more parts), but my trial did not work well. I also implemented a function to print my decision tree (the tree is easy to printed and observed because it is binary). I found that most of my decision trees yielded same decisions as my basic single decision tree did. I think the performance of synthetic social was improved because it has very large number of attributes. Multiple decision trees can generate multiple answers

based on a particular data set, which can minimize the negative effects of noisy data. The cost of generating more robust prediction is time. It took more than 30 seconds to generate random forest for synthetic social, which was much slower than the basic decision tree.