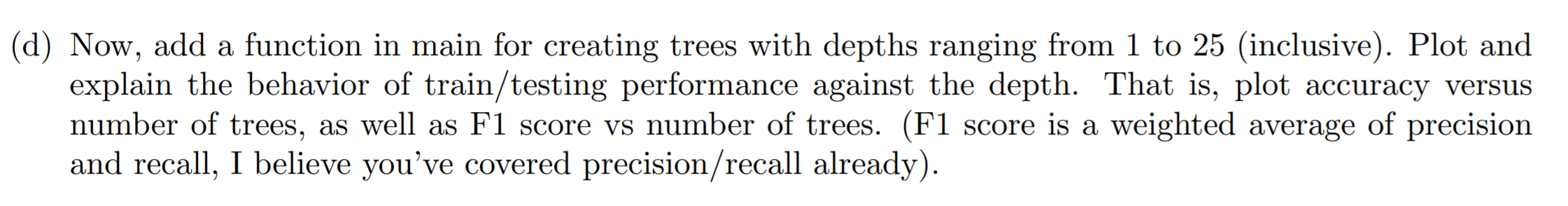
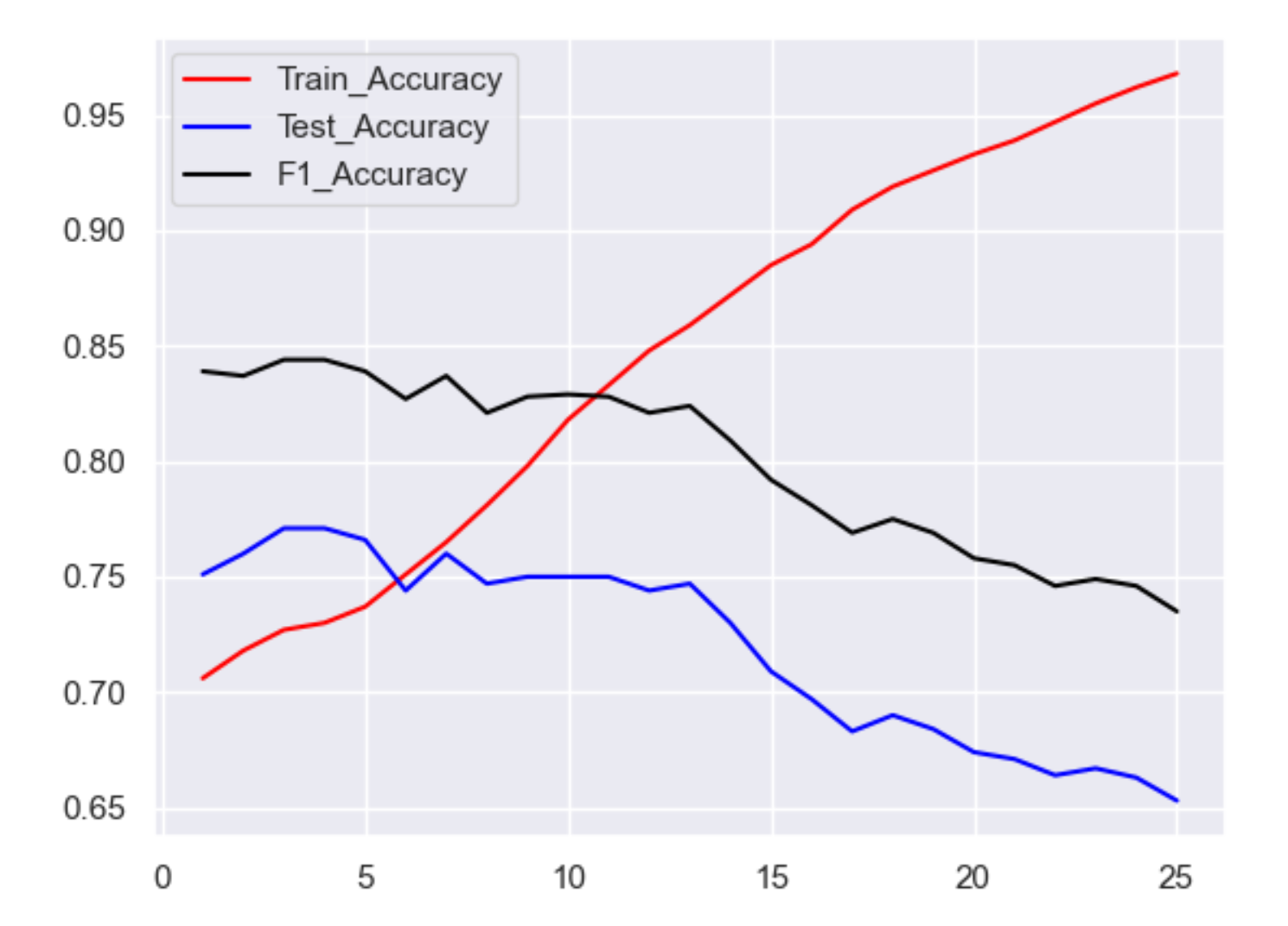
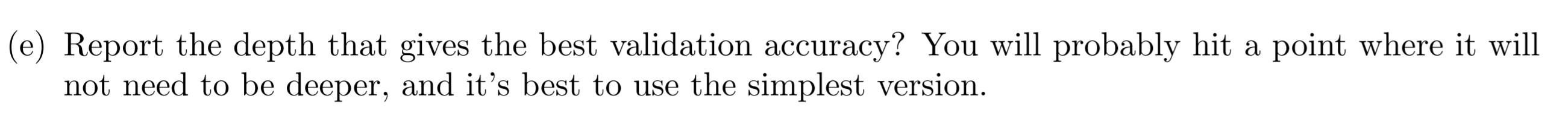
Assignment3 Report

Part1  


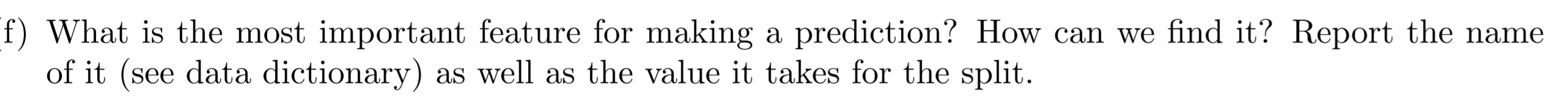


Base on the graph, the more depth we have, the more training accuracy we get, it means that we have an overfitting problem. Our model may fit the training set well, but it will be bad for the testing set.

The trend of testing accuracy and F1 accuracy are similar, the more depth it has, the less accurate it is. It makes sense that the trend are similar since F1 and testing accuracy are just different measurement for the same testing set. If the depth is lower, it will not have overfitting problem and it gives higher accuracy for F1.

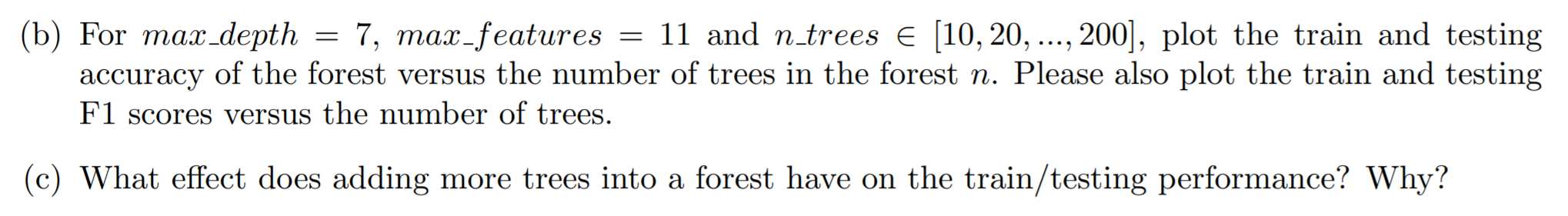


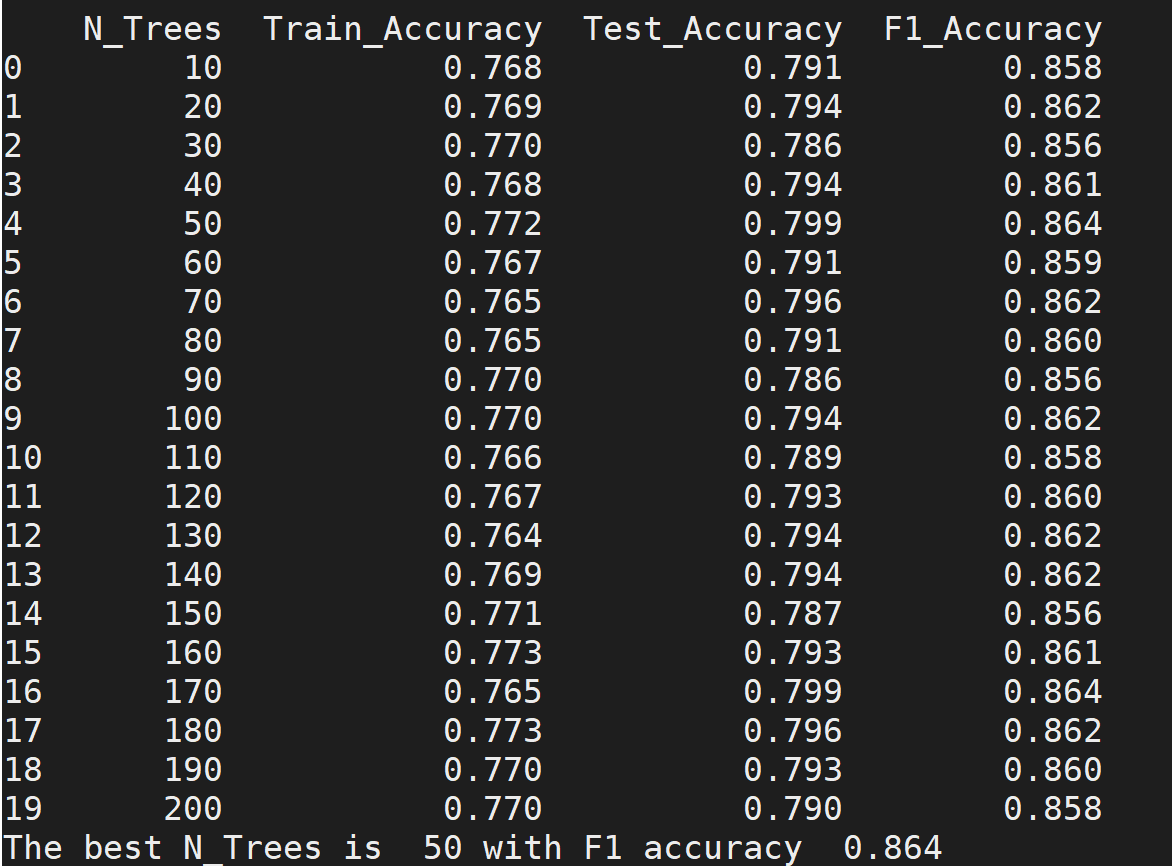


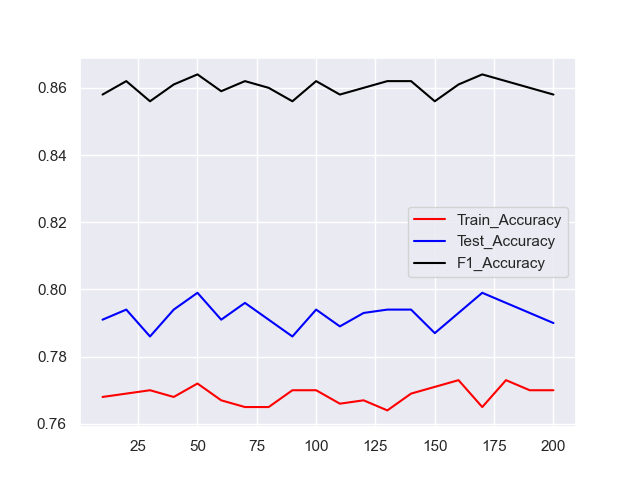




The most information gain should be the most important feature. It should be the first feature to split data into 2 groups, which is the feature for the root of tree.

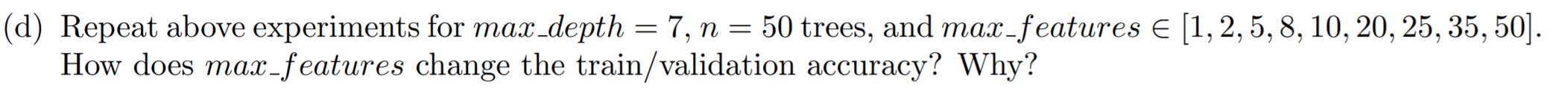
Par2  


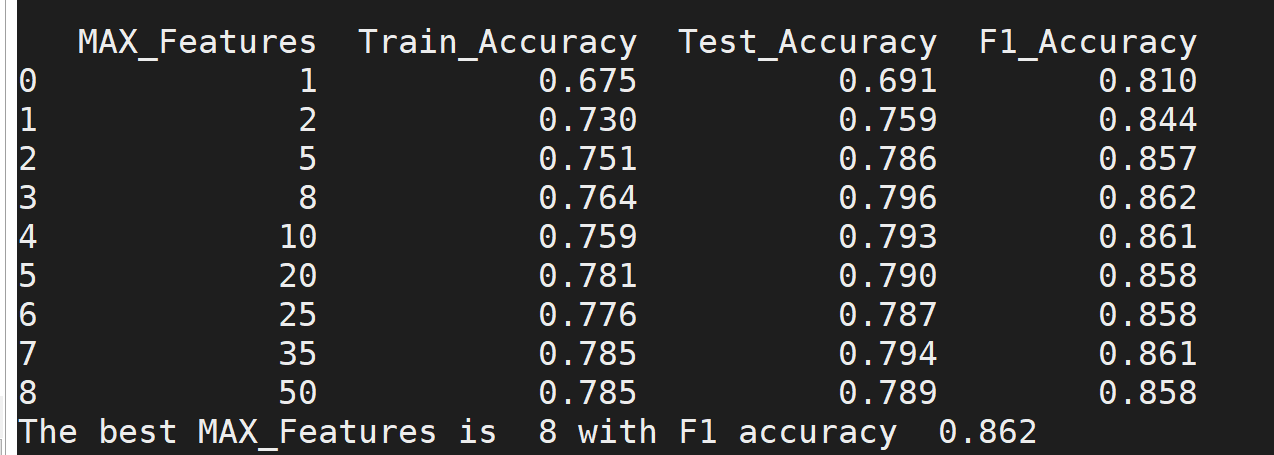


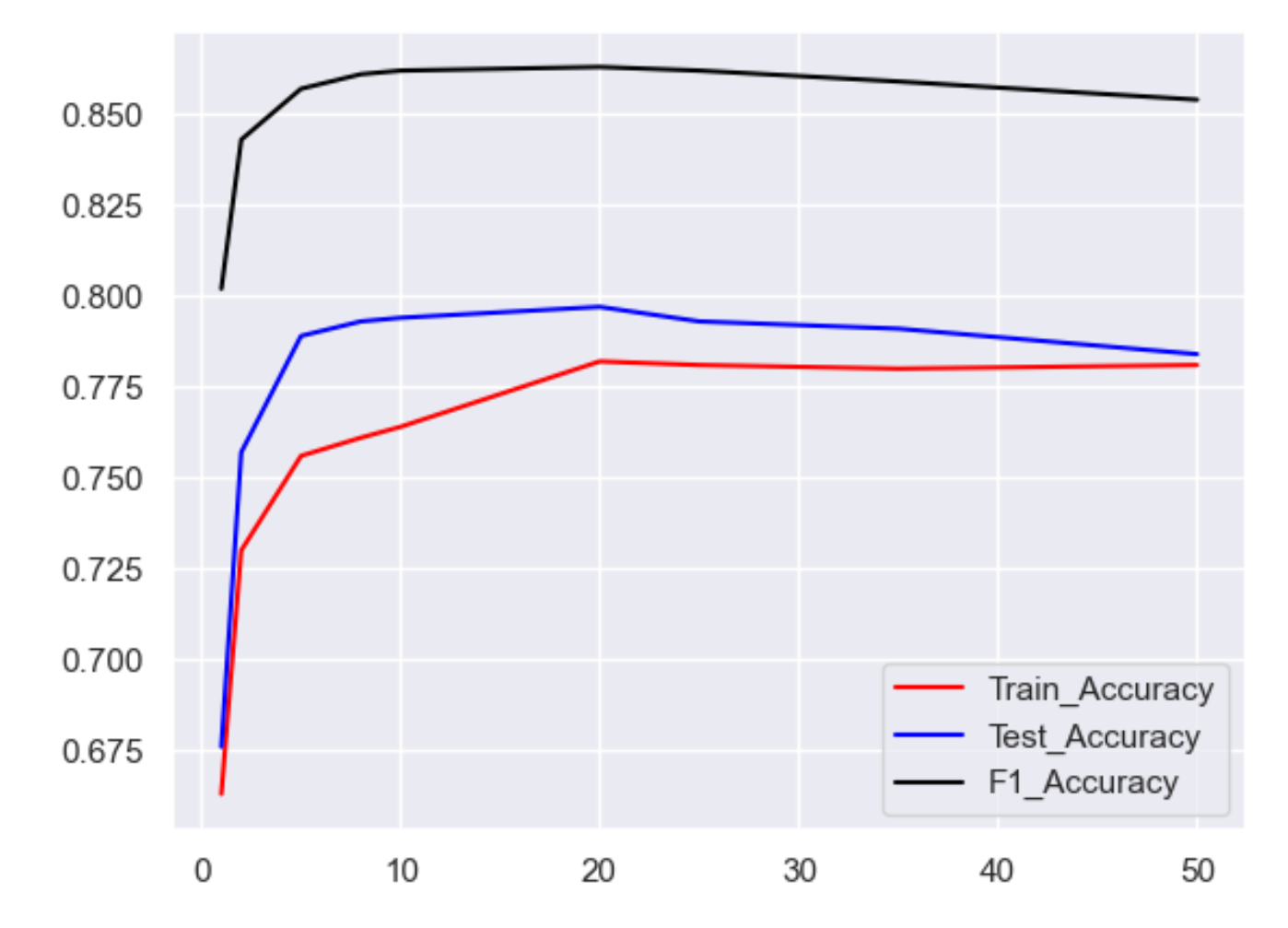


Base on the graph, the accuracy may go up or down by the number of trees increase. The more trees we have, the bigger the voting group it is, which means every decision tree’s vote weight less. The more trees we have, the correlation between trees will be higher. The less tress we have, the correlation between tress will be lower. We want to have high variances in random forest and then average the variances will help our model to be closer to the target.

As a result, our best ‘N\_trees’ is 50 and its F1 accuracy is 86.4%. It makes sense that our number of trees should not be too big in case the correlation of trees is too high, the number of trees should not be too low in case we don’t have enough variances from the dataset.



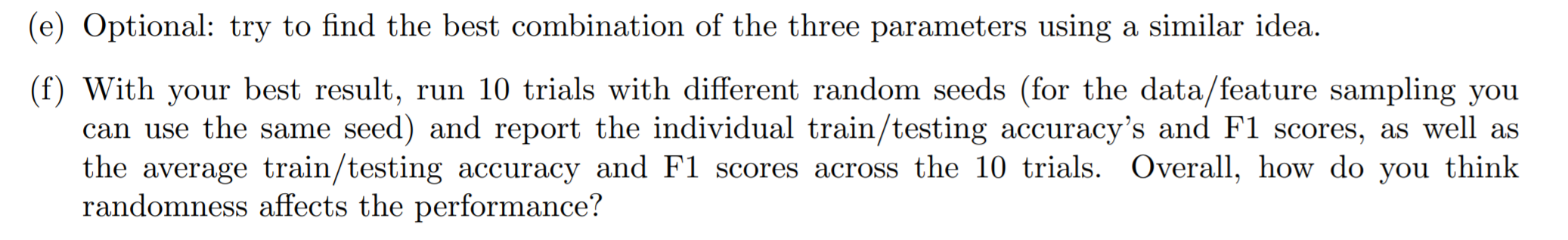


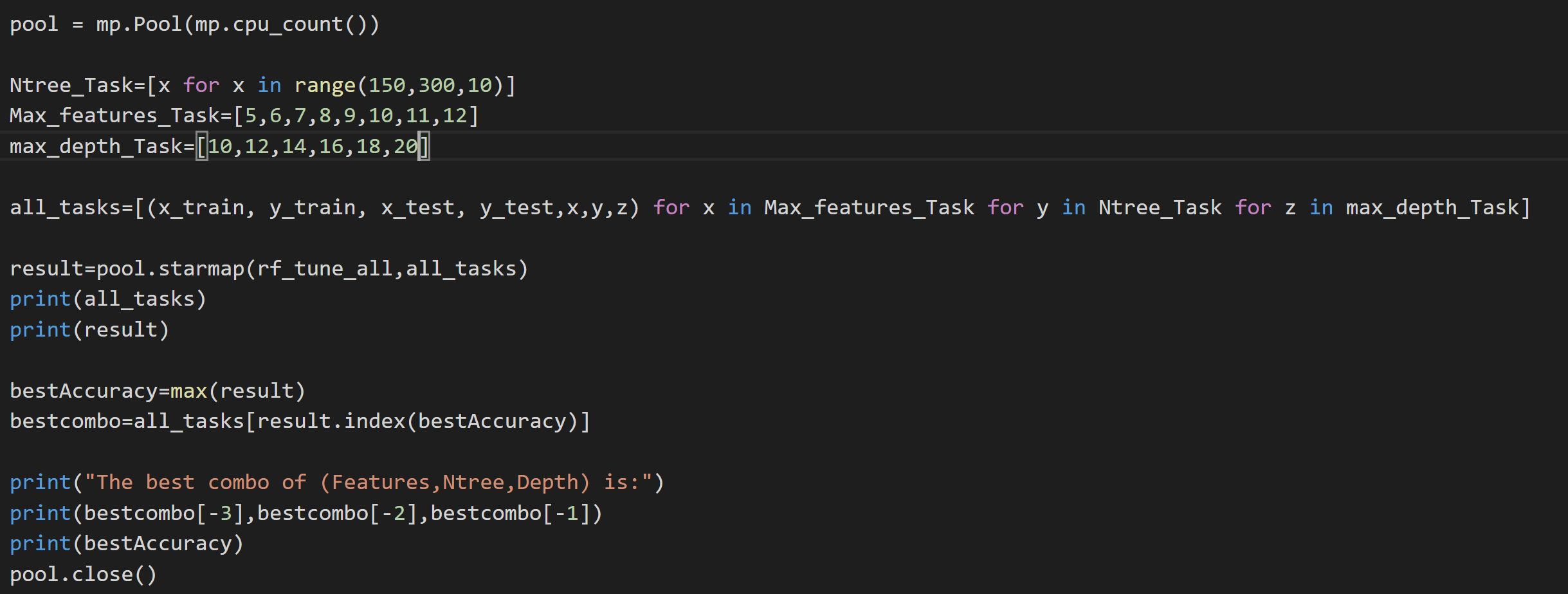


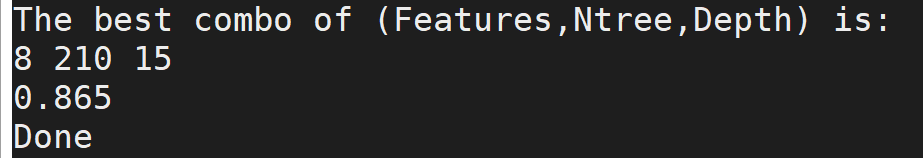
If max\_features is too high, we will not make good random decision trees with good variances, every tree will be similar. If max\_features is too low, we will not find the good feature to split the dataset, and the accuracy will be bad.

In this graph, by max\_features increase, the accuracy will increase and then be stable and slowly decrease.

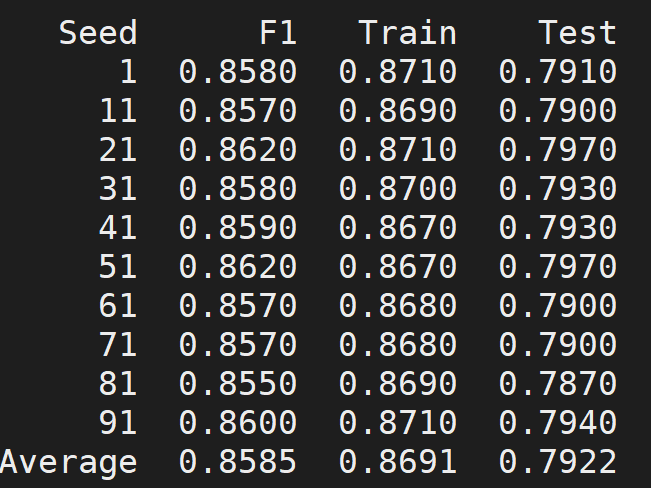
Our best MAX\_Features is 8 and its accuracy is 86.2%



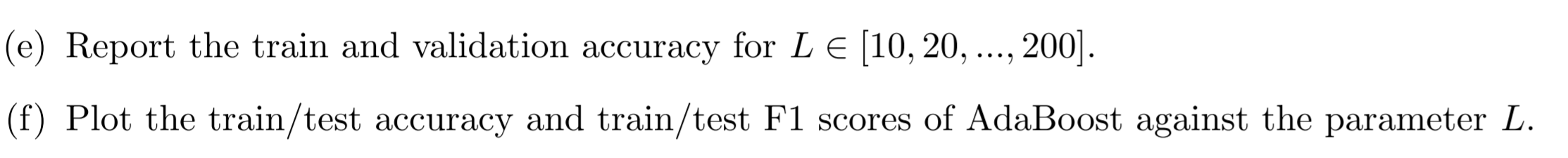


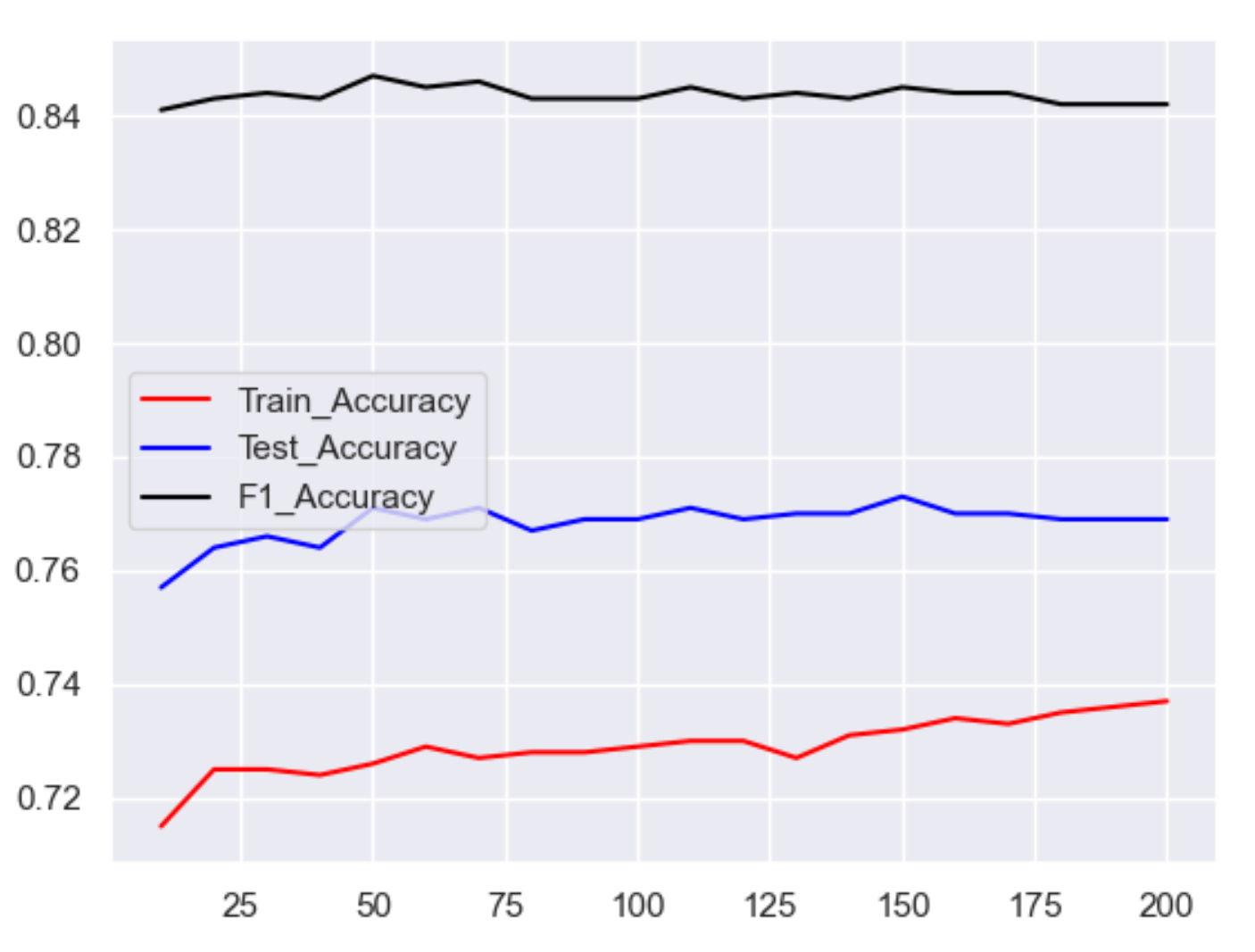
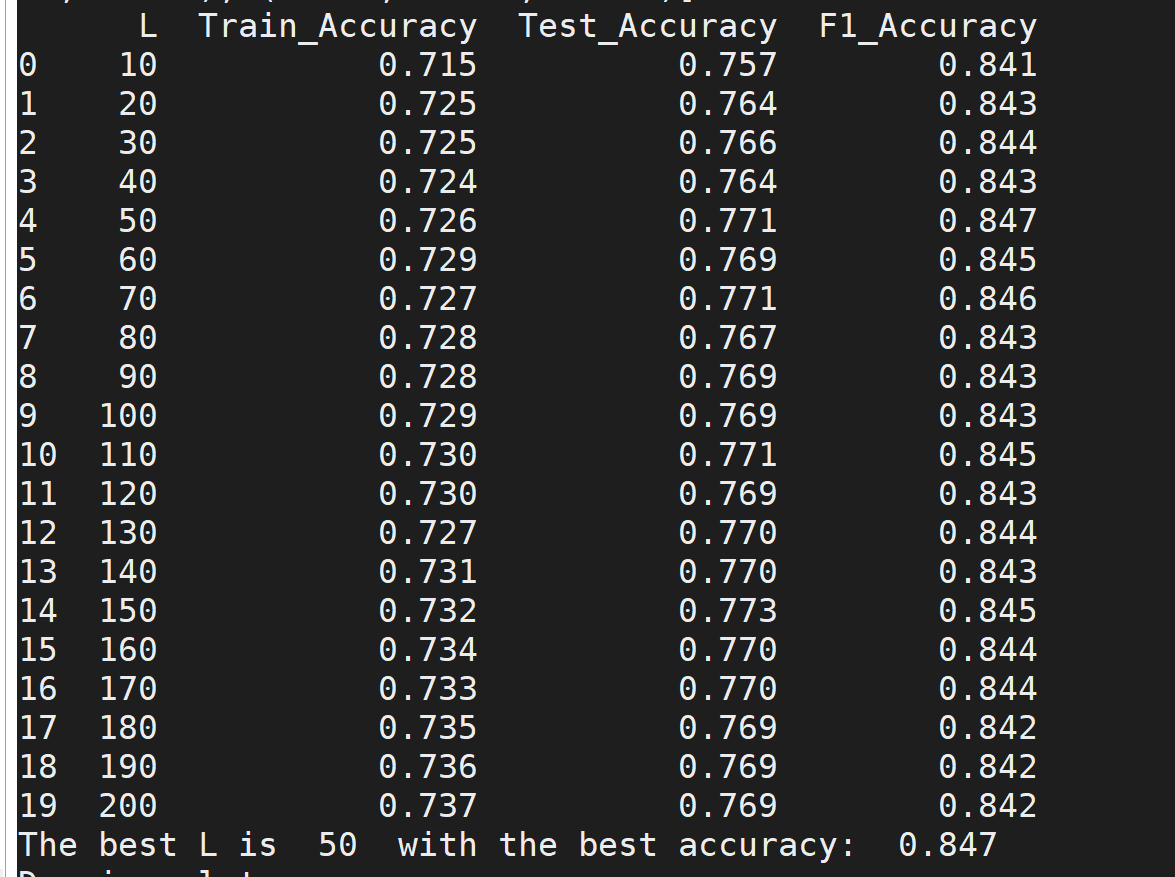


I used parallel programming to run a bunch of combos, the best accuracy I can get is 86.5% with Max\_features=8, N\_tree=210, Depth=15

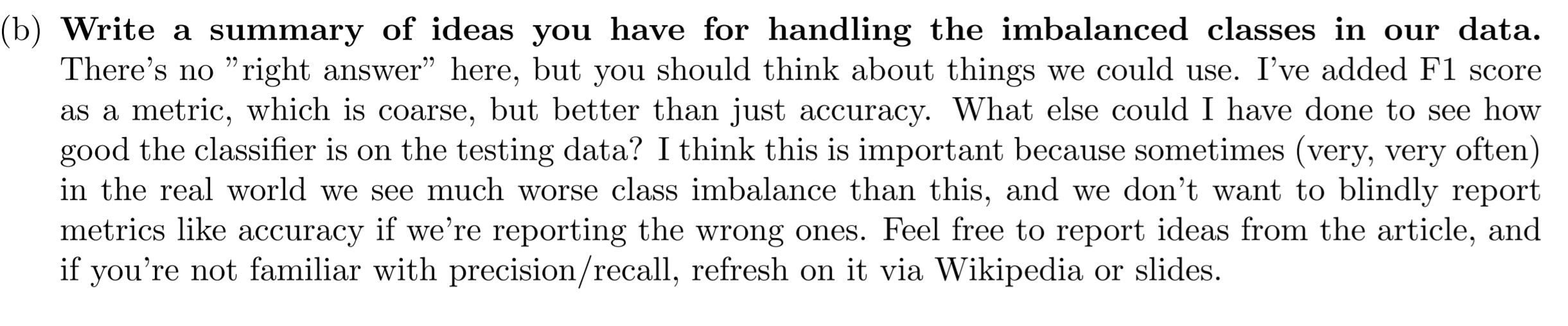


For running 10 different seeds, I don’t think random seeds will impact the result much. Random forest will be random, however, if you use the same features, it will not be too different.

Part3  




When L=50, we get the best accuracy 84.7%



In order to solve the problem of unbalanced dataset, we can under sample the majority classes base on ‘Bayesian argument of Wallace et al’. Under sample will generate more variances, but we can ensemble all classifier to average the variances.

‘Tomek links’ is another good idea to remove the noisy pairs of the data and it will make the border clear so that it is easier for ML algorithm to make the decision boundary.

‘SMOTE and descendants’ is a way of over sample, it will generate more sample for the minority group.

‘Adjusting class weights’ will make minority samples more important, as a result, the decision boundary will bias to majority samples less and make a more fair judgement.

I feel for this project, ‘SMOTE and descendants’ is a good idea, since people who vote for the same party will have close similarities.