Submitted by: Oluwasegun Somefun, somefuno@oregonstate.edu

AI534 — IA4 Homework Report Due Dec 3rd 11:59pm, 2021

1 Introduction

This report is on Decision Tree and Ensembles for Mushroom Classification, with respect to the Implementation Assignment 4.
Team Name: Oluwasegun Ayokunle Somefun

2 Part 1: Decision Tree (50pts)

a. What are the first three splits selected by your algorithm? This is for the root, and the two splits immediately beneath the root. What are their respective information gains?

The first 3 splits selected by the decision algorithm using maximum mutual information gain split criterion are highlighted in Table 1.

Table 1: First 3 node splits

Node	feature	information gain
Root Split	odor=n	**0.54**
True Branch Split	bruises?=f	**0.4**
False Branch Split	spore-print-color=r	**0.1**

b. Evaluate and plot the training and validation accuracies of your trees as a function of d_{max} ranging from 1 to 10. At which depth does the train accuracy reaches to 100%? Do you observe any overfitting?

At depth $d_{\mathsf{max}} = 6$, the train accuracy reaches 100%. This together with the validation accuracy of 100% is reported in Figure 1. Technically, there was no observation of overfitting, since the validation accuracy does not reduce even as $d_{\mathsf{max}} > 6$ increases. However, this would imply overtraining, which could lead to overfitted decisions, depending on the data distribution.

3 Part 2: Random Forest (35pts)

a. For each d_{max} value, create two figures, one for training accuracy and one for validation accuracy. The training accuracy figure should contain four curves, each showing the train accuracy of your random

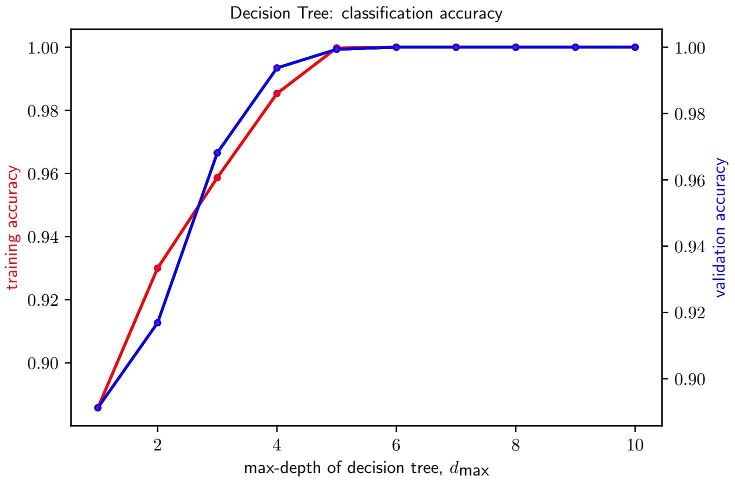

Figure 1: Decision Tree: training/validation class accuracies as a function of maximum depth

forest with a particular m value as a function of T. Repeat the same process for validation accuracy.

Compare your training curves with the validation curves, do you think your model is overfitting or underfitting for particular parameter combinations? And why?

The training curves and validation curves are oultined in Figures 2–7. Comparing both curves, we see that for each maximum depth $d_{\max}$ the model performance of the random forest is more dependent on the number of random subsampled features m compared to the increase in the number of trees T in the forest ensemble.

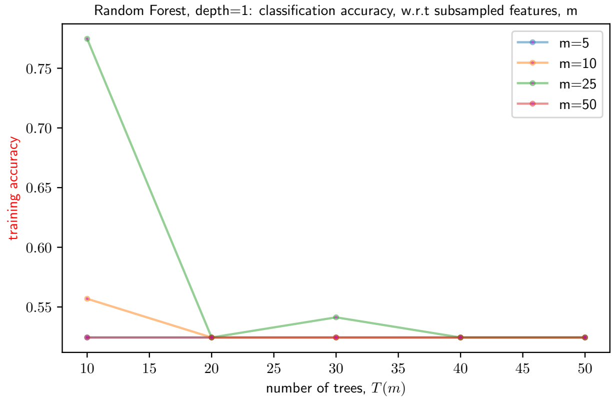

Figure 2: Random Forest $d_{\max} = 1$: Training accuracy

Also, we observe that as expected, both train and validation curves behave similarly, hence indicating random forest leads to low variance models.

For $d_{\max} = 1$, see Figures 2–3, observe that the model underfits the data for all m as the number of

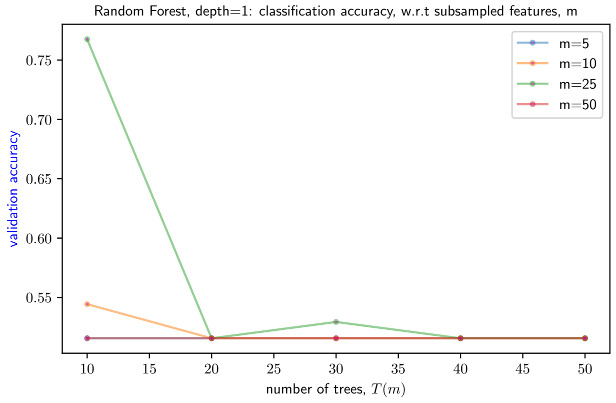

Figure 3: Random Forest $d_{\text{max}} = 1$: Validation accuracy

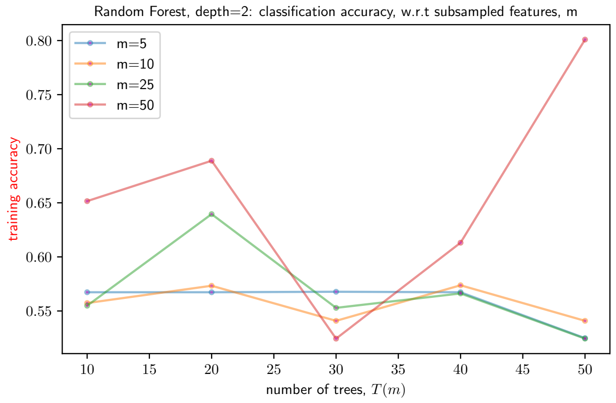

Figure 4: Random Forest $d_{\text{max}} = 2$: Training accuracy

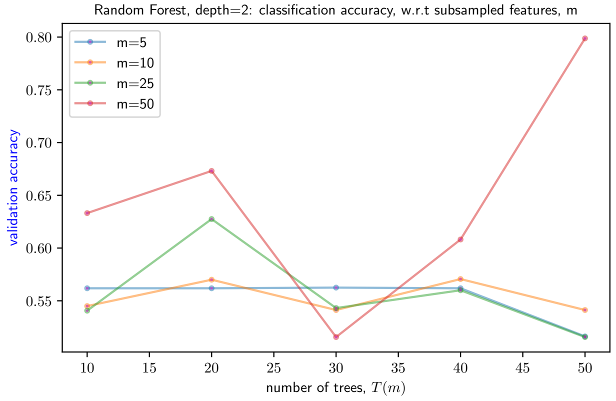

Figure 5: Random Forest $d_{\text{max}} = 2$: Validation accuracy

trees $T > 10$ is increased.

For $d_{\text{max}} = 2$, see Figures 4–5, observe that as m increases the model performance improves on the

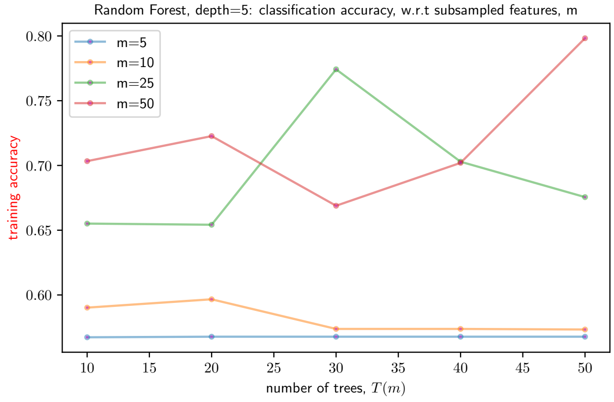

Figure 6: Random Forest $d_{\text{max}} = 5$: Training accuracy

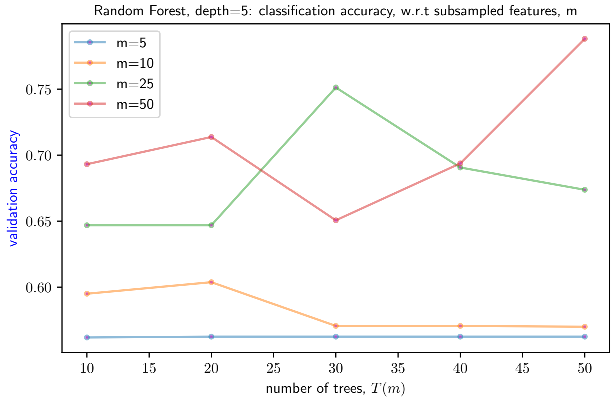

Figure 7: Random Forest $d_{\text{max}} = 5$: Validation accuracy

data. Increase in the number of trees $T > 20$ leads to underfitting, except for when $m = 50$, where the increase in number of trees caused a significant improvement in performance accuracy.

For $d_{\text{max}} = 5$, Figures 6–7 shows that as m increases the model performance improves on the data better than the previous lower mamxium depths. Further increase in the number of trees $T > 20$ led to underfitting for the two lower values of m. When $m = 25$, the threshold moves to $T > 30$, and when $m = 50$, the increase in number of trees eventually causes a significant improvement in performance accuracy.

b. For each d_{max} value, discuss what you believe is the dominating factor in the performance loss based on the concept of bias-variance decomposition. Can you suggest some alternative configurations of random forest that might lead to better performance for this data? Why do you believe so?

For $d_{\text{max}} = 1$, irrespective of m and T, the model has the highest bias, and a low variance. For $d_{\text{max}} = 2$, while variance is low, as m increases, the bias error generally reduces, for $T <= 20$ and

$T >= 40$, otherwise bias increases. For $d_{\max} = 5$, again the variance is low, and as m increases, the bias error generally reduces, irrespective of T.

To obtain better performance with lower bias error, the number of sub-sampled features with replacement m, in the **Random forest** should be increased to the full or total number of features in this data-set, that is: $m = 117$.

4 Bonus Part 3: AdaBoost (20pts) and Kaggle competition (5pts)

a. For each $d_{\max}$ value, create a figure showing two curves, showing the accuracy (y-axis) on train and validation of your ensemble as a function of T. Repeat the same process for validation accuracy.

Compare your training curves with the validation curves, do you think your model is overfitting or underfitting for particular parameter combinations? And why?

Comparing the training curves with the validation curves in Figures 8-10, neither underfitting nor overfitting can be observed. This is clearly seen as the training and validation accuracies are similar.

This behaviour is expected in boosting, since boosting leads to decision-tree models with a lower bias and lower variance. However, overtraining can be observed as the number of trees in the ensemble increase.

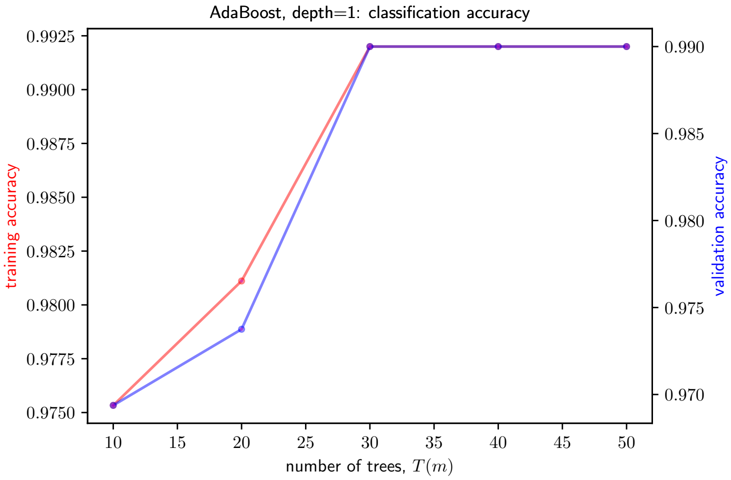

Figure 8: Adaboost: Train and validaition curves for $d_{\max} = 1$

b. For each $d_{\max}$ value, discuss what you believe is the dominating factor in the performance loss based on the concept of bias-variance decomposition. Can you suggest some alternative configurations of the

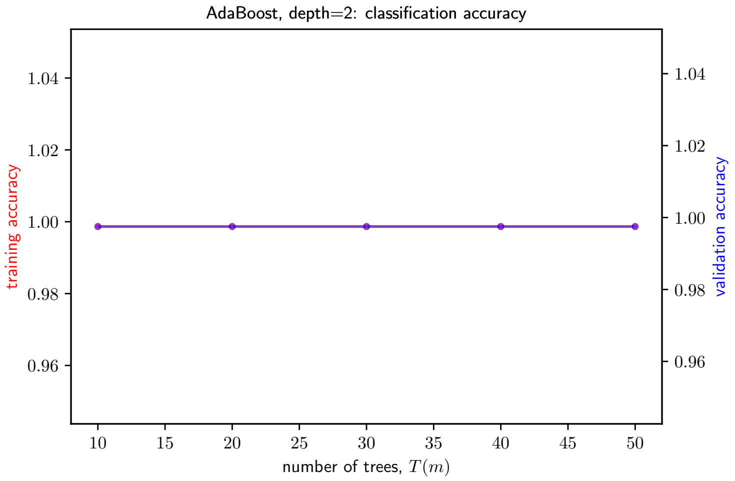

Figure 9: Adaboost: Train and validaition curves for $d_{\text{max}} = 2$

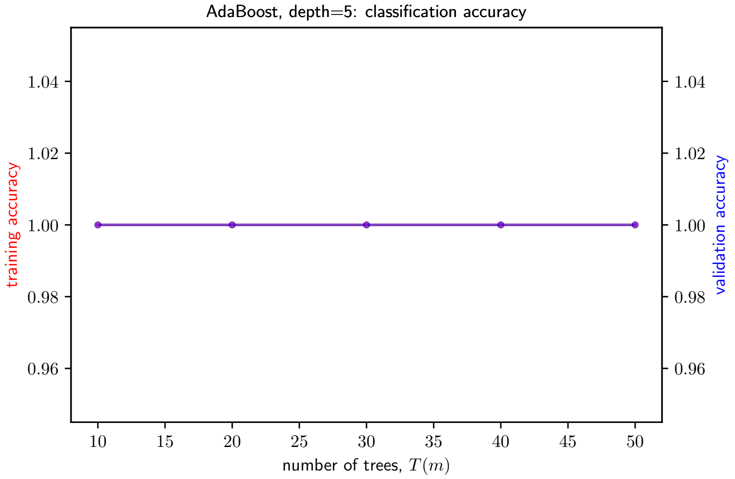

Figure 10: Adaboost: Train and validaition curves for $d_{\text{max}} = 5$

ensemble that might lead to better performance for this data? Why do you believe so?

From the given Figures 8-10, we observe that for $d_{\text{max}} = 1$, as T increases, performance increases, and that the boosted model is of low bias, and low variance. Similarly, for $d_{\text{max}} = 2$, the model is of lower bias, and lower variance, and performance improves closer to the optimum compared to the previous $d_{\text{max}} = 1$ setting. Again, for $d_{\text{max}} = 5$, again the the model is of lower bias, and lower variance, and performance improves to the optimum value of 100% compared to the previous $d_{\text{max}} = 2$ setting.

However, we observe that it is possible that overtraining occurs at $T > 30$ for $d_{\text{max}} = 1$ and $d_{\text{max}} = 5$, while for $d_{\text{max}} = 2$, similar pattern is observed at $T > 10$. That is, it was observed that, there is no benefit or loss from adding more trees to the ensemble given these thresholds. In simple words, adding more trees past these thresholds make the ensemble bloated.

Therefore, to obtain the best decision, with the lowest performance loss for this data-set, the ensemble can be minimally configured using $T = 30$ and $d_{\max} = 5$. This configuration was used for the **Kaggle Competition**.

Generally, we observe that for **Adaboost**, the dominating factor in the low-bias low-variance performance is the number of trees T in the ensemble, followed by the $d_{\max}$ of each tree.