A close-up of a note

Description automatically generated

Went to office hours, helped/got help from TAs and various students.

A math equations on a piece of paper

Description automatically generated

A close-up of a note

Description automatically generated

A close-up of a paper

Description automatically generatedA paper with mathematical equations

Description automatically generated

Q4.1:

­­­

4.2



4.3

1. How did you deal with categorical features and missing values?
   1. Substituted for categorical features the most frequent occurrence, based on the idea that most likely, the missing features would most likely be the most common feature. For the missing values, replaced them with median value.
   2. I completed them by using scikit pipeline coupled with simpleimputer, onehotencoder, and ColumnTransformer.
2. What was your stopping criterion?
   1. When depth was hit, it is a tunable hyperparameter.
3. How did you implement random forests?
   1. Leveraged the decision tree. The random forest implementation is basically just a decision tree in disguise.
4. Did you do anything special to speed up training?
   1. Nothing special other than changing the hyperparameters (slowly and painfully)
5. Anything else cool you implemented?
   1. I’m proud of my pre-processing, took me a very long time to figure it out and a lot of reading python libraries commands.

4.4

A black background with white text

Description automatically generated

A black background with white text

Description automatically generated

Spam Kaggle: 0.817

Titanic Kaggle: 0.822

4.5

2.

A screenshot of a computer

Description automatically generated

3.

A graph with numbers and lines

Description automatically generated

Depth 10, 20, 27, 34, 37 gives the highest accuracy. This is broken. It should not be like this. I do not know why, I swear it was working earlier.

4.6

A screenshot of a computer

Description automatically generated