

# CS422 Project 3

Nicholas Ang

November 2022

# Adaboost

## Adaboost using Decision Tree Stump

Adaboost uses a tree stump over a tree with greater depth because the purpose of boosting is to use multiple weak learners to form a strong classifier that can classify most of the data. Stumps are very weak and generally have very poor accuracy as it separates based on a single feature. However, as Adaboost iterates over and forms new stumps using previous information, the final classification improves drastically. Trees with greater depths are not as simple as one feature splits like stumps. Trees with greater depths have to account for more features. In addition, the combination of the stumps creates a voted weighted final classifier that essentially mimics a decision tree with greater depth.

Adaboost differs from random forest ensembles as it utilizes previous information to adjust for present information. Adaboost would create a classifier, find what went wrong, and adjust for it so that it does not keep making the same mistakes. Adaboost typically works with the entire set of training data. The data is worked with weak learners and weighs each sample of data according to correctness. The weak learners then classify samples and the prediction is voted on based on their weighted classifications. Random forests are not as smart and do not account for mistakes. Instead it uses bagging to select portions of data to train each tree with. The trained trees all predict a sample and then it is voted on. In addition, random forests work well with larger decision trees and poorly works with stumps.

## Adaboost using Perceptron

Adaboost using decision tree stumps creates linearly separable divisions that are either vertical or horizontal. Adapting Adaboost to be able to use perceptrons would require the algorithm to use single perceptrons as classifiers. The perceptron would also have to be stopped by a certain epoch. Perceptrons are lines that linearly separate the data but in any direction compared to the decision stump's only vertical and horizontal lines. The perceptron would work like a decision stump so Adaboost would be able to adjust and adapt to mistakes to try and create a classifier that can correctly predict the data.