Naïve Bayes Report

Jess Nguyen 000747411

# Data Description

Data set that was used in this test is Website phishing data <https://archive.ics.uci.edu/ml/datasets/Website+Phishing>. It contained 9 features name: 'SFH', 'popUpWindow', 'SSLfinal\_State', 'Request\_URL', 'URL\_of\_Anchor', 'web\_traffic', 'URL\_Length', 'age\_of\_domain', and 'having\_IP\_Address'. The result of each row would be classified as -1, 0, or 1 which transcribes to be Phishy, Suspicious, and Legitimate. These number and names are also used in the columns data.

# The Tests

A screenshot of a computer

Description automatically generated with medium confidence

# Discussion

To recap, my best score for kNN was K=8, Manhattan distance, Normalized z-score with an average of 86-87%. As for decision tree, a max depth of 40 returned an average of 90%. Naïve Bayes’ was worse than my Manhattan distance and my decision tress results with 82.63%. This has to do with the fact that Naïve Bayes’ uses normal distribution, but the dataset has multiple overlapping nodes that doesn’t have a clear classification at a certain point on a grid making it seem random. This causes Naïve Bayes’ to predict more generally, similar to kNN. Decision trees allows for better fitting especially when the max nodes were set to 40. With the max node to 40, it showed that it was the sweet spot for the dataset causing it to not overfit and not be too general.

The average probability score for correct was 90.52% and incorrect was 81.47%. With these scores I could possibly use it to forecast other phishy website before they become phishy. Since Naïve has a high confidence of 90.52%, taking precautions with Naïve Bayes’ prediction is worth acting on.