Jonathan Blade - JGB180000

Joshua Durana - RCD180001

Portfolio: Similarity

"k Nearest Neighbors" (kNN) is a supervised learning algorithm that can perform either classification or regression by analyzing an observation's k nearest neighbors. Unlike Linear and Logistic Regression, kNN does not create a model of the data to evaluate the data, instead the algorithm determines the output by comparing the given datapoint to its most immediate neighbors. However, due to kNN's reliance on how each observation stands relative to the rest of the data set, kNN is prone to high variance and low bias on the bias-variance-tradeoff.

Similar to kNN, decision trees can be used for either classification or regression, unlike kNN however, decision trees divide the data recursively until a conclusion is reached for each observation. This makes decision trees more prone to bias on the bias-variance-tradeoff. This method also leads decision trees to be less accurate than its counterparts for both classification and regression. However, what decision trees lack in accuracy they make up for in ease of interpretability.

K-means clustering finds clusters by first randomly choosing k observations, and k is a chosen number of clusters. Then it goes through each data point and calculates the center of each cluster. Then it repeats this process until the center of each cluster doesn't change. This model is good for larger datasets, but you need to find the best k or number of clusters.

Hierarchical clustering finds clusters by initially making each observation a cluster. Then, it finds the nearest cluster and they merge into one cluster. This process repeats until all observations are in one cluster. Hierarchical clustering is best used for data with a hierarchical

structure. But, its main downside is that it's computationally intensive on large data sets and when an observation is assigned a cluster it cannot leave that cluster.

Model based clustering tries to find clusters by the density of the observations. First it calculates the density of the data. Then it tries to build the model by trying to make the shapes of the clusters. There are different shapes and they're identified whether the shape, volume, or orientation is equal, variable, or matches with the coordinate axis. The algorithm goes through each shape to find the best fit.

Principal Component Analysis (PCA) is a dimensional reduction technique that can be used on a data set to create a new coordinate space with a reduced number of axes. PCA analyzes the variables of a data set and reduces them to a number of principal components. PCA captures a certain amount of variance in the data. Each axis is then represented by a principal component in order of decreasing variance. PCA is especially useful in machine learning when a dataset has a large number of variables. However, due to its unsupervised nature PCA loses some interpretability. Additionally, because some data is lost in PCA it tends to be less accurate than other methods.

Linear Discriminant Analysis (LDA) is another technique used to simplify data but serves to maintain the classes of the dataset. LDA attempts to find a linear combination of variables that maximizes separations between classes and minimizes the standard deviation of each class. LDA is useful for datasets with a high number of observations.