Jeffrey Li, Nikilas John, Shariq Azeem, Valari Graham
Dr. Mazidi
CS 4375

a. How kNN and decision trees work for classification

kNN: First, the system loads the entire training data into memory, instead of making a model out of it. Then, for each test observation, the system looks at the feature space and assigns a class to it, based on the class of the majority of its k nearest neighbors.

Decision Trees: First, the system builds a tree by continuously splitting the training dataset into subsets, based on the values of its attributes, until the subset at a node has all the same values, or if further splitting will not add any value to the predictions. Then, for each test observation, it goes through the tree, compares the test value with node condition, follows the corresponding branch, and makes a decision when it has gone through the entire tree and reached one of the leaf nodes.

b. How kNN and decision trees work for regression

kNN: This model is a supervised, non-parametric algorithm. This is used to calculate the average of the numerical target of the K nearest neighbors. K in kNN regression indicates the count of the nearest neighbors. The kNN model predicts the value of the output variable by using a local average.

Decision Trees: This model builds regression in the form of a tree structure. It breaks a data set into smaller subsets. The output is a tree with decision nodes and leaf nodes. The model is used to predict continuous valued outputs. The MSE value explains how far our predictions are from the original target variable. The decision tree algorithm is prone to overfitting, so it is best to use cross validation to find the optimal minimum number of children per leaf node.

c. How the 3 clustering methods of step 3 work

k-Means: This clustering method works by iteration, assigning k centroids randomly on the plot, then assigning each point on the plot to the nearest centroid. The centroid is then recalculated

and then the algorithm iterates. The exit case for this algorithm is when convergence is realized. Convergence is realized when the recalculated centroid does not move a set percentage away from their initial random assignment. For example, if the set percentage to trigger convergence was .01%, your run of k-Means would take a lot of iterations to spit back out a result, but would be very specific.

Hierarchical: This clustering method works as an alternative to k-Means due to it being completely different in execution. k-Means is held back due to its necessary defining of "k", and the steps you need to take to define an optimal "k". This greedy algorithm works by building a dendogram from the bottom up using these steps. First, the algorithm places each observation into their own cluster, then calculates the distance between each cluster. After this, the closest two clusters are combined and the algorithm iterates until every cluster is combined into one. There are also three different measurement methods you can use to calculate the distance between the clusters.

1. Single linkage: calculates the shortest distance between any points in the clusters.
2. Complete linkage: calculates the longest distance between any points in the clusters.
3. Average linkage: calculates the average distance between all the points in the clusters.

Model-Based: This clustering method is more of an extension to k-Means, but the biggest difference is that it defines its own "k". The execution of this algorithm involves applying the maximum likelihood of fitting to the data and then outputting a model with the number of clusters it used.

d. How PCA and LDA work, and why they might be useful techiques for machine learning

PCA: Principal component analysis (PCA) is a dimensionality reduction technique that reduces the number of input variables of the data. It transforms the data into a new coordinate space by reducing the number of axes. This process is done by standardizing the data, calculating the covariance matrix of the data set, and then calculating the eigenvalues and eigenvectors of the covariance matrix to find the principle components. Finally, the data is recast on the principle component axes. This technique is useful for data sets with large amounts of variables that can be confusing to understand, but since data is being lost, the accuracy will also decrease.

LDA: Linear Discriminant Analysis (LDA) is another dimensionality reduction technique, but it is supervised whereas PCA is unsupervised. The goal of LDA is to find a maximum separation of the classes while minimizing the standard deviation within the class. This is accomplished by calculating the mean vectors of the classes, calculating scatter matrices within and between the classes, and then calculating the eigenvalues and eigenvectors. Afterwards, the eigenvectors with the largest eigenvalues are chosen to transform the data into a new space. Like PCA, LDA is useful for reducing the amount of variables involved, but it is better when the class is known.