SEARCHING FOR SIMILARITY: NARRATIVE DOCUMENT

Isabelle Kirby, Bridgette Bryant, Rikita Patangay

Zuhayr Ali

How kNN and Decision Trees Work for Classification and Regression

Regression

kNN

In regression, kNN works by averaging the values of the k closest neighbors to estimate a value for the observation being tested. This trait makes kNN especially effective for regression problems in which the majority of observations cluster close together, as the neighbors deviate so little from each other that the tested observation must also deviate very little. It also doesn't need a linear distribution of data to make useful predictions and is generally quite accurate. However, outlier data is conversely prone to being predicted inaccurately, and the process of kNN is much more computationally intensive and takes longer especially as the value of k grows.

Decision Trees

In regression, decision trees work by using a top-down greedy approach to estimate the best splits in data to minimize the residual sum of squares of each region, repeating the process until reaching a threshold that is often specified. When the data is qualitative, decision trees make their decisions based on a binary feature. Their structure is akin to if-else statements and thus are much more interpretable than other regression models but are less accurate by design and run the risk of severely overfitting data, generating an overabundance of leaf nodes. This latter problem can be addressed by the practice of pruning to produce fewer, larger leaf nodes.

Classification

kNN

In classification, kNN works by finding the distances between the closest k neighbors of the data point to try to classify it based on how close it is to its neighbors (whom may be of different classes). In a way it basically groups data points together based on how close they are to each other. This makes it very easy to implement, useful for nonlinear data, and versatile. It usually has a high accuracy however, it sensitive to the scale of the data as well as irrelevant features of the data. It also requires a lot of computation and memory while being relatively slow to predict, especially with a large k value.

Decision Trees

In classification, decision trees work by finding the best predictors which give good 'splits. With these predictors it will create a split in the tree (a decision). Each split gives you a logical representation of which predictor value can easily classify it. It calculates each split using a greedy-recursive algorithm which split based on a top-down approach to partition and examine predictors (to see if they will make good splits). It will choose what seems best greedily and recursively continue until it hits a stopping point which can either be after a specific number of splits or until the cost of splitting isn't worth the split for classification. Trees can often overfit the data, to avoid this you can attempt to prune the tree, which will remove some of the last few splits to make the tree a bit more generic for the test data.

How the 3 Clustering Methods of Step 3 Work

k-Means

K-means clustering identifies centroids and groups observations by how similar they are to the nearest centroid. First it starts with a random assignment and assigns each observation to the nearest centroid. It then recalculates the centroids and repeats the steps repeatedly until 2 converge. This is a good way to see trends in data, as groups will be made based on whether the computer thinks a certain row belongs to the same entity or classification. However, a k needs to be specified before the function can run and will repeatedly be recalculated until the optimal k value is reached.

Hierarchical Clustering

With hierarchical clustering, we don't have to specify the number of clusters we think will be needed. This algorithm works by placing each observation into its own cluster, calculating the distance between all clusters,

and combining the two closest ones until there is only one cluster left behind. It tends to run slowly with large amounts of data and is not as flexible as kmeans is (i.e.: once an observation is grouped into a cluster it is stuck there).

Research Model-Based Clustering

The Research Model-Based clustering algorithm calculates the p-values for the hierarchical clustering based on multiscale bootstrap resampling. This will help determine if the clusters calculated are supported by the data. This helps with understanding the graph produced by RStudio.

How PCA and LDA Work, and Why they Might be Useful Techniques for Machine Learning

PCA

Principal Components Analysis is a data reduction technique that helps us reduce the dimensions of our datasets. PCA will manipulate the data and reduces the number of axes in a new coordinate space. In this reduced new coordinate space, each axis will represent a principal component. The first principal component (PC1) will represent the dimension of the most significant variance, and the other principal components represent decreasing variance. Since it is a data reduction technique, we will be losing data and may also lose accuracy in any models. Many times, in machine learning you will come upon high-dimensional datasets that can be hard to explore without reduction. This is when PCA is used in ML.

LDA

LDA works by seeking to find a linear combination of the predictors that will maximize the separation of the classes while minimizing the within-class standard deviation. LDA is a supervised classification technique. Again, it is basically used to reduce the data, so we are able to see the numbers clearer and make predictions with the data.