Lloyd Beaufils, Jerrry Bonnell, Randall Naar, Gururaj Shriram

ECE-648

18 October 2017

Project 1 - Group 5

The Effect of Irrelevant Attributes on k-NN Classifiers

The k-NN classifier is a version of the nearest neighbor classifier that looks at k neighbors as opposed to simply one. For any given example, the classifier finds its k nearest neighbors and chooses a class label based upon those of those neighbors. Since this classifier makes use of geometric distances to other examples when deciding class labels, it is essential that those examples be relevant. Each attribute contributes equally to the distance, so attributes that should have no bearing on the class label can quickly overpower the relevant attributes and lead to improper classification. With the irrelevant attributes adding essentially random values to and dominating the distance, the classifier degenerates until it does little better than label classes at random. This behavior is dependent upon the ratio of relevant attributes to irrelevant attributes. The higher that ratio, the less of a negative impact the irrelevant attributes will have. The more irrelevant attributes, then the more terms negatively impacting the distance formula and the worse the classifier will perform, on average. Our group has run some tests to obtain evidence of these claims, and in this paper, we will outline our findings.

For this project, we used two separate k-NN classifiers to verify two <<DEBUG: THREE?>> different datasets. For the k-NN classifiers, we used a built-in Python algorithm known as Scikit and compared our results with an algorithm we wrote based upon the pseudocode in the book. <<DEBUG: OURS IS BETTER? BECAUSE BREAKERS?>> We ran these algorithms on two separate datasets- the Iris dataset from the UCI repository and the Animals dataset that we fabricated. The results of our experiments are located below.

We obtained the Iris dataset from the UCI machine learning repository. The attributes are sepal and petal length and width, and the examples are classified into one of three species of iris. This domain is well-known for being simple, and that simplicity aids us in demonstrating our findings. With no irrelevant attributes, the k-NN classifiers (with k ranging from 1 to 10) all classify randomly-partitioned testing sets with at least 95% accuracy. Our data clearly show that adding irrelevant attributes immediately worsens our accuracy rate, plummeting about twenty points when two irrelevant attributes are added to the dataset. This downward trend continues all the way until the accuracy rate is around an abysmal 30% with 18 irrelevant attributes. This is the case for both k-NN classifiers, with varied values for k. The error rate is above 50%, indicating that having irrelevant attributes makes the classifier do worse than random; the bad attributes actively make worse decisions than flipping a coin. The sharpest drop in accuracy occurs with the addition of the first 6 irrelevant attributes. This makes sense because the amount of bad values eclipse and overcome the amount of good values in the distance formula (3 relevant attributes), so at that point it reaches and dips slightly below 50%, or pure randomness.

The Animals dataset yielded similar results to the Iris dataset. We generated this data programmatically, specifying ranges for valid attributes and generating random numbers within those ranges. There are 5 classes representing different animals, with both relevant (correlated) and irrelevant attributes. The k-NN classifiers do a slightly poorer job with no irrelevant attributes when compared to those classifying the Iris dataset. With only relevant attributes, both classifiers average about 91% accuracy across varying values for k. The higher the k value, the more accurate the classifier is. Again, as with the Iris dataset, the classifiers decrease in accuracy with the addition of irrelevant attributes. However, for this dataset, the decline is more gradual.





