ECS 171 - Forest Cover Project Write-up

Ian Woods, Jesse Dyer, Miguel Covarrubias

December 3, 2014

Brief Description of KNN

The k-Nearest Neighbors (KNN) Algorithm involves classifying as a particular type, given the type of the k closest samples to it, as determined by some distance function. Our initial approach was to use a standard Euclidean distance function, defined as follows:

$$\sqrt{\sum_{k=1}^{n}(x_{jk} - x_{ik})^2}$$

We decided to begin with this naive approach to isolate failure points (essentially, if something were to work incorrectly, we could rule out the distance function). Later on, we intend to write a distance function specifically tailored to this dataset for maximum correct classification.

Using different values for k can alter the predicted results - for example, if a sample of unknown type is closest to a sample of type A with a calculated distance of 1, but two samples of type B are at a distance of 1.1 from our unknown sample, choosing $k = 1$ will yield the result A for the unknown's type, and $k = 3$ will yield the result B.

Results of Standard KNN Algorithm for Various K Values

We ran 20 preliminary trial runs on the standard KNN algorithm and found that, at these low values, the predictor is relatively stable as k changes. We will test higher k values against the entire dataset (these were tested against 1/10th of the dataset to save time).

k	1	2	3	4	5	6	7	8	9
% correct	0.909845	0.908847	0.910052	0.908158	0.908503	0.910465	0.915628	0.910981	0.913150

k	10	11	12	13	14	15	16	17	18
% correct	0.913150	0.909122	0.908124	0.907504	0.906816	0.910843	0.911979	0.906506	0.909673

k	19	20
% correct	0.907229	.906988