Fast k-Nearest Neighbor Classifier (beta)

Contents

Fast KNN with shrinkage estimator for the class membership probabilities

Fast Nearest Neighbor Searching

The `fastknn` method implements a k-Nearest Neighbor (KNN) classifier based on the ANN library. ANN is written in `C++` and is able to find the k nearest neighbors for every point in a given dataset in `O(N log N)` time. The package RANN provides an easy interface to use ANN library in `R`.

The FastKNN Classifier

The `fastknn` was developed to deal with very large datasets (> 100k rows) and is ideal to Kaggle competitions. It can be up to 20x faster then the popular `knn` method from the `R` package class, for large datasets. Moreover, `fastknn` provides a shrinkage estimator to the class membership probabilities, based on the inverse distances of the nearest neighbors (**see the PDF version**):

$$P(x_i \in y_j) = \frac{\displaystyle\sum_{k=1}^{K} \left(\frac{1}{d_{ik}} \cdot (n_{ik} \in y_j) \right)}{\displaystyle\sum_{k=1}^{K} \left(\frac{1}{d_{ik}} \right)}$$

where x_i is the i^{th} test instance, y_j is the j^{th} unique class label, n_{ik} is the k^{th} nearest neighbor of x_i, and d_{ik} is the distance between x_i and n_{ik}. This estimator can be thought of as a weighted voting rule, where those neighbors that are more close to x_i will have more influence on predicting x_i's label.

In general, the weighted estimator provides more **calibrated probabilities** when compared with the traditional estimator based on the label proportions of the nearest neighbors, and reduces **logarithmic loss** (log-loss).

How to use

Required Packages

Find the Best k

Plot Classification Decision Boundary