Tutorial: Geometric Graphs for Instance-Based Learning

Godfried Toussaint *
School of Computer Science
McGill University
Montréal, Québec, Canada
Tel: 514-398-7077
e-mail: godfried@cs.mcgill.ca

Abstract

In the typical nonparametric approach to pattern classification and instance-based learning, random data (the training set of patterns) are collected and used to design a decision rule (classifier). One of the most well known such rules is the k-nearest-neighbor decision rule (also known as *lazy learning* and *case-based reasoning*). With this rule an unknown pattern is classified into the majority class among its k nearest neighbors in the training set. Several questions related to this rule have received considerable attention over the years. Such questions include the following. How can the storage of the training set be reduced without degrading the performance of the decision rule? How should the reduced training set be selected to represent the different classes? How large should k be? How should the value of k be chosen? Should all k neighbors be equally weighted when used to decide the class of an unknown pattern? If not, how should the weights be chosen? Should all the features (attributes) we weighted equally and if not how should the feature weights be chosen? What distance metric should be used? How can the rule be made robust to overlapping classes or noise present in the training data? How can the rule be made invariant to scaling of the measurements? How can the nearest neighbors of a new point be computed efficiently? What is the smallest neural network that can implement nearest neighbor decision rules? Geometric proximity graphs such as Voronoi diagrams and their many relatives provide elegant solutions to most of these problems.

*This research was supported by NSERC and FCAR.

1 List of Topics

1.1 Nearest neighbor decision rules and Bayes optimality.

1.2 Hart's condensed nearest neighbor rule and its relatives.

1.3 Wilson's edited nearest neighbor rule.

1.4 Introduction to proximity graphs.

1.5 Condensing and editing with Voronoi diagrams.

1.6 Condensing and editing with proximity graphs.

1.7 Proximity graph neighbor decision rules.

1.8 The rectangle-of-influence neighbor decision rule.

1.9 Neural networks for nearest neighbor decision rules.

2 Speaker

Godfried T. Toussaint {see above for address, phone, and e-mail}

3 Motivation and Objectives

The problems mentioned in the preceeding abstract have traditionally been solved with a variety of ad hoc techniques. It turns out that almost all of them can be solved elegantly and efficiently using proximity graphs. Such techniques appear not to be well known in the *knowledge engineering, data mining, and machine learning* communties. One objective of this tutorial is to bring such techniques to light. Another is to show how such techniques can be made to be Bayes optimal. Recently, there has been much work on designing neural networks to implement nearest neighbor decision rules. Such networks invariably use three layers and $O(n^2)$ neurons, where n is the number of patterns in the training data. Furthermore, they are not Bayes optimal. One motivation for this tutorial is to show how we can obtain Bayes optimal neural networks with only one layer and $o(n)$ neurons that require no learning and can be designed easily in $O(n)$ time.

4 Primary/Secondary Audience

The primary audience for this tutorial would consist of researchers and applications developers in the areas of knowledge engineering, data mining and machine learning. The secondary audience would be computer scientists and engineers specializing in algorithms and data structures, particularly in the areas of computational geometry and graph theory.