UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
Department of Electrical and Computer Engineering

ECE 544NA Pattern Recognition

Homework 3

Fall 2013

Assigned: Thursday, September 12, 2013 Due: Tuesday, September 24, 2013

Reading: NNPR Chapters 2 & 3

Problem 3.1

We discussed how k-nearest neighbors is a very powerful tool for classification, but (in its most general form) is difficult to analyze. Consequently, there is a common misconception that the larger the k, the better. k-nearest neighbor classification works as follows:

(a) Given some new point x, find k nearest neighbors $\{X_{(1)}, X_{(2)}, ..., X_{(k)}\}$ s.t. $d(x, X_{(1)}) \leq d(x, X_{(2)}) ... \leq d(x, X_{(k)})$, where $d(x, y)$ denotes distance between two points x and y. The distances are computed over all points in the training set: $\{(X_1, Y_1), (X_2, Y_2), ..., (X_N, Y_N)\}$.

(b) $f_{knn}(x) = Y_{knn} = Majority\{Y_{(1)}, Y_{(2)}, ..., Y_{(k)}\}$, where $Majority$ selects the label Y that occurs most frequently.

In words, we first first find the k closest points to x within the training set, and from their corresponding labels, pick the one that occurs the most (majority). Ties are broken arbitrarily. In the case of 1-nearest neighbor classification, we simply take the label of the closest point in the training set.

Let us take the simple example of binary classification in which we know that $p(X|Y = 1)$ and $p(X|Y = -1)$ are uniform distributions over the unit disks centered at (2,0) and (-2,0), respectively. Prove that in this specific scenario, the risk (assume 0-1 loss) of the 1-nearest neighbor classifier is **lower** than the risk of the 3-nearest neighbor classifier.

Problem 3.2

A Voronoi tessellation is a division of the space $\mathbb{R}^D$ into K classes, $C_1, C_2, ..., C_K$ such that

$$C_k = \{x : \|x - \mu_k\| \leq \|x - \mu_i\| \ \forall \ i \neq k\}$$

Notice that by this definition, the boundary B_{ij} is a subset of both C_i and C_j; the decision is arbitrary on the boundary.

(a) Discrimination between classes μ_i and μ_j for any i and j, can be performed by evaluating the sign of the linear discriminant $y_{ij}(x) = w_{ij}^T(x - b_{ij})$. Find the vectors w_{ij} and b_{ij} in terms of μ_i and μ_j.

(b) Suppose that each class is Gaussian with a covariance matrix Σ common to all classes, and with prior probabilities $\pi_1, \pi_2, ..., \pi_K$. The posterior probability $p(C_1|x)$ can be written as an extended sigmoid function,

$$p(C_1|x) = (1 + e^{-f_{12}(x)} + e^{-f_{13}(x)} + ... + e^{-f_{1K}(x)})^{-1}$$

Write $f_{1k}(x)$ without using μ_1 or μ_k in your answer. You may include w_{1k}, b_{1k}, Σ, and $ln\frac{\pi_k}{\pi_1}$ in your answer.

Matlab Exercises

Problem 3.3

The smoothing parameter (aka bandwidth), h, plays an important role in kernel density estimation. A good criterion for selecting h is one that minimizes the mean-squared error. For a univariate Gaussian kernel, $h^* \approx 1.06\hat{\sigma}N^{\frac{-1}{5}}$ is the optimal choice, where $\hat{\sigma}$ is the estimate of the standard deviation of the samples and N is the number of samples.

(a) Write a function, $randgen(f, N)$ that generates N i.i.d samples from a given probability density function f. You may find the built-in matlab function *rand* to be useful.

(b) For $N = \{10, 100, 1000\}$, generate N independent samples from an exponential distribution with $\lambda = 1$ ($f(x) = e^{-x}[x \geq 0]$, where [.] is the indicator function).

(c) Compute the sample standard deviation, $\hat{\sigma}$, without making any prior assumptions on the distribution (i.e., DO NOT assume that the data are drawn from an exponential distribution). For each N, estimate the optimal bandwidth, $h^*(N)$.

(d) Estimate the pdf using kernel density estimation with a Gaussian kernel for each N, under three different bandwidth settings: $\{h^*(N)/3, h^*(N), 3 * h^*(N)\}$.

(e) Summarize your results by plotting the pdf estimates. You need to have 9 plots overall (3 values of N x 3 values of h). Overlay each plot with the true density, $f(x)$ for $x\epsilon[-1, 4]$ (to save space, consider using the matlab function subplot(3,3,.)). Comment on the influence of h, N, and the kernel itself on the pdf estimates.