HW 7

BRIAN KING

1. DERIVE 8.16 $\hat{p}(C_i|\bar{x}) = \frac{k_i}{k}$

$$\hat{P}(C_i|\bar{x}) = \frac{\hat{p}(\bar{x}|C_i)\,\hat{P}(C_i)}{\Sigma_j\,\hat{p}(\bar{x}|C_j)\,\hat{P}(C_j)}$$

where $\hat{p}(\bar{x}|C_i) = \frac{k_i}{N_i V^k(\bar{x})}$ and $\hat{P}(C_i) = \frac{N_i}{N}$

$$\therefore\ \hat{P}(C_i|\bar{x}) = \left(\frac{k_i}{N_i V^k(\bar{x})} \cdot \frac{N_i}{N}\right) \Big/ \left(\Sigma_j \frac{k_j}{N_j V^k(\bar{x})} \frac{N_j}{N}\right)$$

$$\hat{P}(C_i|\bar{x}) = \frac{\frac{1}{V^k(\bar{x})N} \cdot k_i}{\frac{1}{V^k(\bar{x})N} \Sigma_j k_j} = \frac{k_i}{\Sigma_j k_j}$$

$$\boxed{\hat{P}(C_i|\bar{x}) = \frac{k_i}{k}}$$

Brian King
HW#7

2D Histogram Estimator

Accuracy with h = 1: **42.40%**

Halving the bin size to 0.5 decreased the accuracy by over two-fold (18.14%). Doubling the bin size to 2 increased the accuracy to 56.04%. The maximum attainable accuracy was around 60% with a bin size around 4.

Naïve Estimator

Accuracy with h = 1: **41.07%**

Halving the bin size to 0.5 had the same effect that it did with the histogram estimator. The accuracy dropped to 17.70%. Doubling the bin size to 2 also increased the accuracy to 55.59%. The Naïve Estimator was able to attain a marginally greater accuracy than the 2D Histogram Estimator with results above 60.21% when a bin size of 4 was used.

Kernel Estimator (Gaussian Kernel)

Accuracy with h = 1: **60.55%**

Halving the h-value to 0.5 decreased the accuracy to **57.21%** since the data became less smooth. If the data is smoothed out too much the accuracy also decreased. For instance, at h=20, the accuracy dropped below 60%.

K-Nearest Neighbor

Accuracy with k = 1: **52.48%**

An optimal value for k can be heuristically determined by taking the square root of N. Since N = 1797 for the test data, an adequate estimate for an optimal k is 42. This k values gives an accuracy of **61.49%.**

64D Histogram Estimator

Observation: The computational time and memory space required to utilize a histogram estimator in the 64-Dimensional hyperspace is exponentially greater than that of 2-Dimensional space. This provides an example of the practical utility of principal component analysis in decreasing the power and time to execute a program. The obtainable accuracy will be higher when using more dimensions, but the trade-off is time and memory. An optimal number of dimensions should be sought out to balance computational time/space and the error.