ECE 792-41 Homework 3

Material Covered: Nearest-Neighbor Regression, Curse of Dimensionality, Generalization Error, Bias–Variance Tradeoff, PCA/KLT

Problem 1 (Alternative Neighbor Averaging Method for Simulated Data)

a) Given a regression function $f(x) = x^2 + 2x + 1$ and a generative model $Y = f(X) + e$, where $e \sim N(0, 1)$ and $X \sim \text{Uniform}(-1, 1)$, generate 50 pairs of (x_i, y_i) and graph them using black circles. Also plot the regression function using a black solid curve.

b) We use a method similar to the nearest neighbor averaging to estimate the regression function. We use a neighborhood of fixed radius $\delta = 0.1$. The estimated regression function takes the following form:

$$\hat{f}(x) = \frac{1}{|I(x)|} \sum_{i \in I(x)} y_i, \quad I(x) = \{i : |x - x_i| \leq \delta\}, \tag{1}$$

where $I(x)$ is the set of indices of x_i such that they are within δ in terms of distance from x, and $|I(x)|$ is the number of elements of set $I(x)$. For example, when $x = 0.9$ and $\delta = 0.1$, you first need to find all points that are within the range of [0.8, 1.0] in the x-direction, and then take the average of their values in the y-direction to obtain $\hat{f}(0.9)$. You may want to calculate $\hat{f}(\cdot)$ for all $x \in [-0.9, 0.9]$ with a stepsize 0.01. If there is not a single point within the current neighborhood, use the $\hat{f}$ from the previous step as that for the current step. Draw the estimated regression function using a red solid curve in the same plot of a).

c) Vary the neighborhood radius δ, how does the shape of the estimated regression function change?

Problem 2 (Curse of Dimensionality) Read the first paragraph of the problem statement of *ESL-2.4*. Note that we may also write $\mathbf{X} = (X_1, X_2, \ldots, X_p)$, where $X_k \sim \mathcal{N}(0, 1)$ for $k = 1, \ldots, p$. Use a programming language of your choice. To get started, set $p = 10$. Note that in this problem, all vectors are column vectors.

a) Write a computer program to randomly draw/generate $N = 100$ vectors from the template random vector $\mathbf{X}$, namely, $\{\underset{\sim}{x}^{(i)},\ i = 1, \ldots, N\}$. Note that each vector should contain p normally distributed random numbers. Plot all vectors as points in a 3-D space consisting of the first, second, the last coordinates.

b) Calculate the coordinate value of each point after being projected on to a fixed direction specified by $\mathbf{a} = \underset{\sim}{x}_0/||\underset{\sim}{x}_0||$, namely, $z^{(i)} = \mathbf{a}^T \underset{\sim}{x}^{(i)}$. Here, $\underset{\sim}{x}_0$ is an arbitrary nonzero vector of length p, "T" is the transpose operation, and $z^{(i)} \in \mathbb{R}$. What are the sample mean and sample variance of the projected coordinates $\{z^{(i)},\ i = 1, \ldots, N\}$?

c) Repeat a) and b) for $p \in [1, 80]$. You may want to use a `for` loop to achieve this. Optionally, put your code for parts a) and b) into a function to make your code easier to read. Plot the sample variance of the projected coordinates as a function of p.

d) Calculate the squared distance of each point to the origin, namely, $d_i^2 = ||\underset{\sim}{x}^{(i)}||^2$. What is the sample mean of $\{d_i^2,\ i = 1, \ldots, N\}$? Plot the sample mean of the squared distance as a function of p in the same plot of c). Limit the range of y-axis between 0 and 80. For $p = 5$, inspect the values of any five d_i^2's. Do the results in b) and c) match with conclusion drawn in the third paragraph of *ESL-2.4*?

e) Use the formulas from (b), prove that $\mathrm{Var}(Z) = 1$ where $Z = \mathbf{a}^T\mathbf{X}$, and $\mathbb{E}[D^2] = p$ where $D = ||\mathbf{X}||$. Are the theoretical results in this part consistent with the simulated results obtained in c) and d)? (Hint: The sum of p squared normal random variables is a chi-square random variable χ_p^2. The mean of χ_p^2 is p.)

Problem 3 (Effect of Smaller Training Set on Generalization Error) Given a training sample $\{X_i\}_{i=1}^n$ and a testing sample $\{Y_i\}_{i=1}^m$ that are drawn independent from a normal distribution $N(\mu, \sigma^2)$. We are interested in quantifying the test/prediction/generalization error for $\{Y_i\}_{i=1}^m$.

a) Show that one good estimator for Y_i is $\hat{Y}_i = \frac{1}{n}\sum_{j=1}^n X_j$. (Hint: Construct an estimator for μ using $\{X_i\}_{i=1}^n$, and then propose an estimator for a random variable Y with mean μ.)

b) Show that the expected test/prediction/generalization error is $(1 + \frac{1}{n})\sigma^2$. Plot expected generalization error as a function of the training sample size.

c) Generate one empirical curve using $m = 10$ and varying n. Repeat the empirical curve generation process for 100 times and overlay the curves in one single plot. How is this plot compared to that resulted from b)?

Problem 4 (Bias–Variance Tradeoff) Given a true model $Y_i^{(0)} = \beta_1 x_i + \mu + e_i,\ i = 1, \ldots, n$, $e_i \sim \mathcal{N}(0, \sigma^2)$, we draw a sample $\{(x_i, Y_i^{(0)})\}_{i=1}^n$. Someone falsely believes that the sample is generated from a smaller model $Y_i = \mu + \epsilon_i$, and is trying to estimate μ based on his/her belief using least-squares. Denote the estimate by $\tilde{\mu}$.

a) Calculate the bias of $\tilde{\mu}$. How is the result compared to the bias if the true model is used for estimation?

b) Show that the variance expressions of the estimated μ are $\frac{\sum x_i^2}{\sum x_i^2 - n\bar{x}^2} \cdot \frac{\sigma^2}{n}$ and $\frac{\sigma^2}{n}$ if the true model and the smaller model are used for estimation, respectively. Which one is smaller? Consider the results in (a) and (b), argue whether the smaller model is better.

Problem 5 (Bias and Variance Curves for Polynomial Regression) Assume y is a 5th-order polynomial function of x corrupted by additive Gaussian noise. Select by yourself the true weights $\{\beta_i\}_{i=0}^{5}$ and the noise variance and fix them throughout this problem. Generate a dataset $\{(x_i, y_i)\}_{i=1}^{1000}$, where $x_i \sim \mathcal{N}(0, 1)$. Below, we examine the bias and variance behaviors of the estimators of β_0 (the intercept) at different complexity levels of a fitted model.

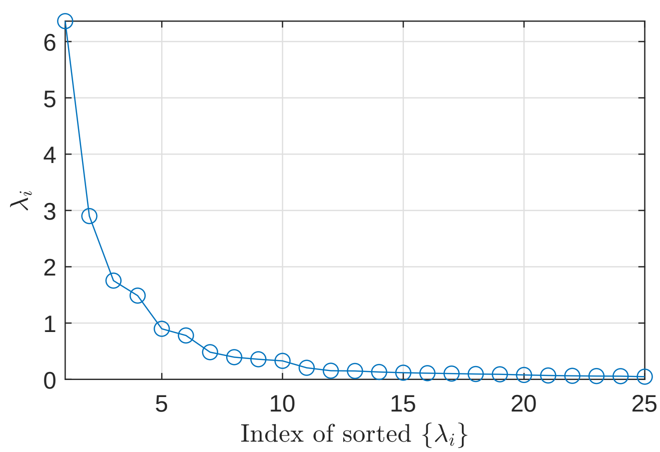
a) Calculate and draw the theoretical curves of bias^2 and variance for fitted models whose polynomial order equals $0, 1, \ldots, 10$. (You may use the under-/overfit formula derived in class.)

b) Keep $\{\beta_i\}_{i=0}^{5}$ and $\{x_i\}_{i=1}^{1000}$ unchanged, repeatedly generate 50 datasets and draw the empirical curves for bias^2 and variance.

Problem 6 (Bonus) (PCA via KLT on Downsampled Yale Face Database) In this problem, we will explore PCA as a visualization tool for Yale Face Database. Download the `.m` files and the database. Extract the face image files into a folder named `yalefaces` and put the `.m` files at the same level of the folder. Open `main_pca_visualization.m` in Matlab.

a) Run the code of (a), explain the data structure of variable `img_buffer`. Set `preview_img_flag` to `1`, re-run the code to visually inspect the whole database.

b) Complete Matlab function `[V, Lambda_mat] = PcaViaKlt(data)` by implementing PCA using eigendecomposition on a sample covariance (not correlation) matrix of the face data. The detailed information about the input and outputs are given in the comments of the incomplete function. You may use built-in function `eig` for eigendecomposition. If your implementation is correct, after running the code of (b), you will obtain a plot similar to the following.



c) Run the code of (c) to visualize a couple of dominating eigenvectors. Comment on whether they reflect some characteristics of the faces you saw in (a).

d) The code of (d) projects each face image (coming from one of the four selected classes) onto a 2D space. Comment on PCA's data visualization performance in this specific example.