# ECE 492-45 Homework 3

## Material Covered: Statistical Learning Introduction, Linear Regression

**Problem 1** (20 points) [Optimality of Mean Operators]

**a)** We are given two variables $X$ and $Y$ that are not independent. Hence, we may use one to estimate the other. Find the best deterministic function $g(\cdot)$ such that it minimizes the expected squared error between $Y$ and $g(X)$ conditioned on $X = x$. You may find a change of variable using $\theta$ in the place of $g(x)$ helpful.

**b)** *Arithmetic average*, or the *sample mean* in a statistical context, is commonly used in everyday life for making quantitative description. We examine a statistical interpretation for the arithmetic average below. A person weighs $\mu$ lb. He tried multiple scales in a supermarket and recorded the reading from each scale, denoted by $Y_i$ for the $i$th scale. We may create a linear model as follows to relate the true weight $\mu$ and the measurement $Y_i$:

$$Y_i = \mu + e_i, \quad i = 1, \ldots, N,$$

where $e_i$ is the measurement error of the $i$th scale. Use the mean-square criterion $J(\mu) = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \mu)^2$ to find the closed-form expression for the best estimator for $\mu$. The expression should contain $\{Y_i\}_{i=1}^{N}$ only, and should not contain such symbols as $\mu$ or $e_i$ as they were not available when readings were recorded. Does the expression make intuitive sense?

**Problem 2** (20 points) [Alternative Neighbor Averaging Method for Simulated Data]

**a)** Given a regression function $f(x) = x^2 + 2x + 1$ and a linear 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 $\delta$ 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)** (Bonus, 5 points) Vary the neighborhood radius $\delta$, how does the shape of the estimated regression function change?

**Problem 3** (20 points) [$k$-Nearest Neighbors] Complete *ISLR-2.4.7*. Repeat (a)–(c) for $(X_1, X_2, X_3) \in \{(1, 2, 3), (1, -1, 1)\}$. Bonus (10 points): Using a programming language of your choice, refactor your code into a function named `MyKnn` with the following input and output variables. We have shown below examples in R and Matlab, but you may also use Python.

R:

```
MyKnn = function(x1, x2, x3, k) {
...
return(Y)
}
```

Matlab:

```
function Y = MyKnn(x1, x2, x3, k)
...
end
```

The file containing the function should be named `MyKnn` with extension `.r`, `.m`, or `.py` and uploaded to the submission link posted on Piazza. The performance of the uploaded function will be automatically/manually assessed, and bonus will be given solely on the percentage of correct classifications using test data. You can assume that when the function is evaluated, the input variables `x1`, `x2`, `x3` will be any value in $\mathbb{R}$, `k` will be less than 6, and the return value being checked against will be either `"Red"` or `"Green"`.

**Problem 4** (20 points) [Least squares]

**a)** We have proved in class that the least-squares estimator for $\beta_1$ of model $y_i = \beta_0 + \beta_1 x_i + e_i$ is

$$\hat{\beta}_1 = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}. \tag{2}$$

Prove that the above expression is equivalent to

$$\hat{\beta}_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}. \tag{3}$$

**b)** (Try it only after 9/16 lecture) Write model $y_i = \beta_0 + \beta_1 x_i + e_i,\ i = 1, \ldots, n$ into the matrix-vector form. Clearly indicate what are $\underset{\sim}{y}, \mathbf{X}, \beta$, and $\underset{\sim}{e}$. Set up the cost function $J(\underset{\sim}{\beta})$ to be minimized. Take gradient with respect to $\underset{\sim}{\beta}$ and set it to zero. Express $\hat{\underset{\sim}{\beta}}$ in terms of $\mathbf{X}$ and $\underset{\sim}{y}$. What are the closed-form solution for $\hat{\beta}_0$ and $\hat{\beta}_1$. Is $\hat{\beta}_1$ consistent with that in a)?

**Problem 5** (20 points) [Linear Regression with R] Complete *ISLR-3.6.1–3, 3.6.7, 3.7.8.*