Due November 6, 2024 at 11:59pm

Homework 4

ESE 4020/5420

Problem 1. Suppose you have bivariate data $(x_1, y_1), \ldots, (x_n, y_n)$. The data has random noise and because of this, we assume that the x_i are not random but that for some values of the parameters a and b the value y_i is drawn from the random variable

$$Y_i \sim ax_i + b + \varepsilon_i$$

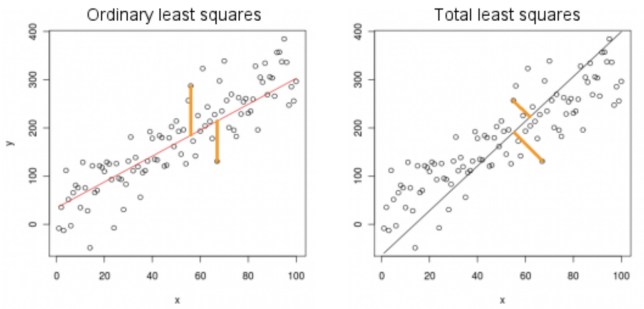
where ε_i is a normal random variable with mean 0 and variance σ^2. We assume all of the random variables ε_i are independent and that σ is a known constant

(a) The distribution of Y_i depends on a, b, σ and x_i. Of these only a and b are not known. Give the formula for the likelihood function $f(y_i \mid a, b, x_i, \sigma)$ corresponding to one random value y_i.

(b) For the data $(x_1, y_1), \ldots, (x_n, y_n)$ give the likelihood and and log likelihood functions (again as functions of a, b, and σ).

(c) Find the maximum likelihood estimates for a and b.

Problem 2. Assume that we are given sample points $\{(x_1, y_1), \ldots, (x_n, y_n)\}$. In the class, we derived the slope β_1 and intercept β_0 minimizing $\sum_{i=1}^{n}(y_i - \beta_1 x_i - \beta_0)^2$. Now, assume that instead of mean squared error, we are interested in minimizing *total squares*. Specifically, we want to find a line such that the sum of the *squared* distance of data points (x_i, y_i) to that line is minimized. The difference between ordinary regression and the minimum total square is shown in the figure below.



Given a line $y = \beta_1 x + \beta_0$ and a point (x_0, y_0), compute the distance between the point and the line. Use this to write down the quantity that we wish to minimize as a function of β_0 and β_1 and the sample points. Then, set the gradients of the loss function you derived in the last part to zero and derive expressions for the optimal β_1 and β_0.

Problem 3. See the Jupyter notebook file for problem 3.

Problem 4. In this question we assume that data is generated according to a distribution $P(X = x, Y = y)$ given as follows: $x \in \mathbb{R}$ and $y \in \{-1, 1\}$, i.e. the data is one-dimensional and the label is binary. Write $P(X = x, Y = y) = P(Y = y)P(X = x|Y = y)$. We let $P(y = +1) = \frac{3}{4}$, and $P(Y = -1) = \frac{1}{4}$, and

$$P(X = x|Y = +1) = \frac{1}{2}\exp(-|x - 2|), \text{ and,}$$

$$P(X = x|Y = -1) = \frac{1}{2}\exp(-|x + 2|).$$

i.e. $P(X = x|Y = +1)$ and $P(X = x|Y = -1)$ follow the Laplace distribution. (The mean of a Laplace distribution with pdf $\frac{1}{2b}e^{(-\frac{|x-\mu|}{b})}$ is μ and the variance is $2b^2$.)

1. Derive the expression for $P(X = x, Y = y)$.
2. Plot the conditional distributions $P(X = x, Y = +1)$ and $P(X = x, Y = -1)$ in one figure, i.e. you should have two labeled curves on one set of axes.
3. Write the Bayes optimal classification rule given the above distribution P and simplify it (**Hint:** in the end you should arrive at a very simple classification rule that classifies an input x based on whether or not its value is greater than a threshold).
4. Compute the probability of classification error for the Bayes optimal classifier.

   Note: *Parts 5 - 7 below can be solved even if you weren't able to do Parts 1 - 4 above*

5. Let us now consider QDA. Given training data $(x_1, y_1), \cdots, (x_n, y_n)$, explain briefly the main steps of training the QDA model. i.e. what quantities/probabilities are being estimated by QDA? What is the parametric model used? How are the parameters of the model found? (Recall that QDA is similar to LDA with the exception that the variance per class can be different.)
6. Assume that the number of training data points is very large (i.e. $n \to \infty$); What will be the exact value of the parameters of the trained QDA model in this case?
7. Simplify the QDA classifier that you obtained in the previous part. (**Hint:** Similar to Part 3, in the end you should arrive at a very simple classification rule that classifies an input x based on whether or not its value is greater than a threshold).
8. Given your answers to Part 3 vs. Part 7, what do you think about the performance of QDA compared to what can be done optimally? Does QDA perform optimally when we have many training data points?