ECE 411 Homework 1 (Fall 2024)
Instructor: Dr. Chau-Wai Wong
Material Covered: Prerequisites, Machine Learning Overview

Problem 1 (20 points) [Course Prerequisite: Calculus]

(a) (10 points) [Partial Derivatives] Given constants Y_i, $i = 1, 2, 3$ and x_{ij}, $i = 1, 2, 3$, $j = 1, 2$, J is a function of independent variables β_1 and β_2 defined as follows:

$$J(\beta_1, \beta_2) = \Big(Y_1 - x_{11}\beta_1 - x_{12}\beta_2\Big)^2 + \Big(Y_2 - x_{21}\beta_1 - x_{22}\beta_2\Big)^2 + \Big(Y_3 - x_{31}\beta_1 - x_{32}\beta_2\Big)^2. \quad (1)$$

Under some technical conditions, we know that $J(\beta_1, \beta_2)$ to have a unique minimizer at $(\beta_1, \beta_2) = (\beta_1^*, \beta_2^*)$. With the help of partial differentiation you learned in calculus, prove that β_1^* and β_2^* satisfy the following relationships:

$$\sum_{i=1}^{3} Y_i x_{i1} = \sum_{i=1}^{3} \sum_{j=1}^{2} x_{ij}\beta_j^* x_{i1}, \quad (2a)$$

$$\sum_{i=1}^{3} Y_i x_{i2} = \sum_{i=1}^{3} \sum_{j=1}^{2} x_{ij}\beta_j^* x_{i2}. \quad (2b)$$

You will later learn in class that they are called *normal equations*.

(b) (10 points) [Limit] Simplify the following expression

$$\lim_{\beta \to \infty} \frac{\exp(\beta x)}{\exp(\beta x) + \exp(\beta y)} \quad (3)$$

(i) when $x > y$ and $x < y$, respectively. (ii) Explain the results in your own words and/or using graphs.

This problem has practical uses in linear regression and neural networks.

Problem 2 (20 points) [Course Prerequisite: Probability/Statistics and Matrix Operations]

(a) (10 points) [Linearity of the Expectation Operator] X is a random variable with probability density function $f(x)$, $x \in \mathbb{R}$. a and b are constants. (i) Prove using the definition of *expectation* that expectation is a linear operator,[1] i.e., $\mathbb{E}[aX + b] = a\,\mathbb{E}[X] + b$. ($ii$) Explain intuitively or using a real-world example why the expectation is a linear operator.

(b) (10 points) [Covariance Matrix] The *variance–covariance matrix* or simply the *covariance matrix*, which will be used later in this course, can be regarded as a generalization

[1] For those of you who took ECE 301 Linear Systems, this is the *linearity property* of the expectation operator $\mathbb{E}[\cdot]$.

from a measure of the variation of a single random variable X to a vector of random variables $\mathbf{X} = [X_1, \ldots, X_n]^T$. Its mathematical definition is as follows

$$\text{Cov}(\mathbf{X}) = \mathbb{E}\left[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{X} - \mathbb{E}[\mathbf{X}])^T\right] \tag{4}$$

and your job is to verify using simple matrix operations that the covariance matrix is a square matrix consisting of variance terms on the diagonal and covariance terms off the diagonal.

Prove that the elementwise expression of the covariance matrix for a length-3 random vector $\mathbf{X} = [X_1, X_2, X_3]^T$ is as follows:

$$\text{Cov}(\mathbf{X}) = \begin{bmatrix} \text{Var}(X_1) & \text{Cov}(X_1, X_2) & \text{Cov}(X_1, X_3) \\ \text{Cov}(X_2, X_1) & \text{Var}(X_2) & \text{Cov}(X_2, X_3) \\ \text{Cov}(X_3, X_1) & \text{Cov}(X_3, X_2) & \text{Var}(X_3) \end{bmatrix}. \tag{5}$$

Hint: Write variance and covariance in terms of $\mathbb{E}[\cdot]$ will be helpful.

This problem has practical uses in linear/quadratic discriminant analysis.

Problem 3 (20 points) [Python Basics] Python is the most popular programming language used by the machine learning community. In this problem, you will go through a Python tutorial to quickly learn its syntax in order to work on neural network problems that you will encounter in the near future. Note that this process is very different from (and much easier than) learning a programming language for the first time: You learn it by focusing on the special syntax that the "baseline" programming language you are good at does not have. Limit the time spend on this problem to one hour.

In this specific problem, we will use Google Colab to execute Python code. (You are feel to use other development environments such as `VS Code`.) An introduction on how to get started with the Colab environment can be found *here*. After opening the above link in your browser, click "Copy to Drive" to play the code in your own Google Drive. Below is *a quick tutorial* on using Python that includes examples on using common Python libraries such as `numpy` and `matplotlib`. (If you feel like to learn from another Python tutorial, feel free to follow that tutorial.) While going over the tutorial, you should note the following unique features in Python:

- Syntax wise: no "`end`" or "`}`"; use indent.
- List/set comprehension
- Built-in data structures such as list and dictionary
- The absence of `++`, `--`, `&&`, `||` operators
- `range()` function
- Exponentiation operator

What to submit: A self-designed Python coding cheat sheet of no more than one page that is tailored to yourself.