CS559-B HW1

Due: Oct. 4th, 2022

Problem 1 (5pt): Provide an intuitive example to show that $P(A|B)$ and $P(B|A)$ are in general not the same. Provide matrix examples to show $AB \neq BA$. (No math derivation is needed).

Problem 2 (10pt): Independence and un-correlation

(1) (5pt) Suppose X and Y are two continuous random variables, show that if X and Y are independent, then they are uncorrelated.

(2) (5pt) Suppose X and Y are uncorrelated, can we conclude X and Y are independent? If so, prove it, otherwise, give one counterexample. (Hint: consider $X \sim Uniform[-1, 1]$ and $Y = X^2$)

Problem 2 (15pt): [Minimum Error Rate Decision] Let $\omega_{max}(x)$ be state of nature for which $P(\omega_{max}|x) \geq P(\omega_i|x)$ for all $i = 1, \dots, c$.

(1) Show that $P(\omega_{max}|x) \geq \frac{1}{c}$

(2) Show that for minimum-error-rate decision rule, the average probability of error is given by

$$P(error) = 1 - \int P(\omega_{max}|x)p(x)dx$$

(3) Show that $P(error) \leq \frac{c-1}{c}$

Problem 4 (10pt): [Likelihood Ratio] Suppose we consider two category classification, the class conditionals are assumed to be Gaussian, i.e., $p(x|\omega_1) = \mathrm{N}(4, 1)$ and $p(x|\omega_2) = \mathrm{N}(8, 1)$, based on prior knowledge, we have $P(\omega_2) = \frac{1}{4}$. We do not penalize for correct classification, while for misclassification, we put 1 unit penalty for misclassifying ω_1 to ω_2 and put 3 unit for misclassifying ω_2 to ω_1. Derive the bayesian decision rule using likelihood ratio.

Problem 5 (15pt): [Minimum Risk, Reject Option] In many machine learning applications, one has the option either to assign the pattern to one of c classes, or to reject it as being unrecognizable. If the cost for reject is not too high, rejection may be a desirable action. Let

$$\lambda(\alpha_i|\omega_j) = \begin{cases} 0, \ i = j \text{ and } i, j = 1, \dots, c \\ \lambda_r, \ i = c + 1 \\ \lambda_s, \ \text{otherwise} \end{cases}$$

where λ_r is the loss incurred for choosing the $(c+1)$-th action, rejection, and λ_s is the loss incurred for making any substitution error.

(1) (5pt) Derive the decision rule with minimum risk.

(2) (5pt) What happens if $\lambda_r = 0$?

(3) (5pt) What happens if $\lambda_r > \lambda_s$?

Problem 6 (25pt): [Maximum Likelihood Estimation (MLE)] A general representation of a exponential family is given by the following probability density:

$$p(x|\eta) = h(x)\exp\{\eta^T T(x) - A(\eta)\}$$

- η is *natural parameter.*
- $h(x)$ is the *base density* which ensures x is in right space.
- $T(x)$ is the *sufficient statistics.*
- $A(\eta)$ is the *log normalizer* which is determined by $T(x)$ and $h(x)$.
- exp(.) represents the exponential function.

(1) (5pt) Write down the expression of $A(\eta)$ in terms of $T(x)$ and $h(x)$.
(2) (10pt) Show that $\frac{\partial}{\partial\eta}A(\eta) = \mathrm{E}_\eta T(x)$ where $\mathrm{E}_\eta(.)$ is the expectation w.r.t $p(x|\eta)$.
(3) (10pt) Suppose we have n i.i.d samples $x_1, x_2, \ldots, x_n$, derive the maximum likelihood estimator for η. (You may use the results from part(b) to obtain your final answer)

Problem 7 (20pt): [Logistic Regression, MLE] In this problem, you need to use MLE to derive and build a logistic regression classifier (suppose the target/response $y \in \{0, 1\}$):

(1) (5pt) Suppose the classifier is $y = x^T\theta$, where θ contains the weight as well as bias parameters. The log-likelihood function is $LL(\theta)$, what is $\frac{\partial LL(\theta)}{\partial\theta}$?

(2) (15pt) Write the codes to build and train the classifier on Diabetes dataset (attached in Canvas). The Diabetes dataset contains 768 samples with 9 features for 2 outcomes. To simplify the problem, we only consider: **Glucose** and **BMI** as our features. Based on the simplified settings, train the model using gradient descent. Please show the classification results. (Note that (1) you could split the Diabetes dataset into train/test set. (2) You could visualize the results by showing the trained classifier overlaid on the train/test data. (3) You could tune several hyperparameters, e.g., learning rate, weight initialization method etc, to see their effects. (3) you **can not** use the package to directly train the model (e.g., sklearn.linear_model.LogisticRegression)).