Homework Assignment 8

Due: Friday, December 6, 2024, 11:59 p.m. Mountain time
Total marks: 15

Policies:

For all multiple-choice questions, note that multiple correct answers may exist. However, selecting an incorrect option will cancel out a correct one. For example, if you select two answers, one correct and one incorrect, you will receive zero points for that question. Similarly, if the number of incorrect answers selected exceeds the correct ones, your score for that question will be zero. Please note that it is not possible to receive negative marks. **You must select all the correct options to get full marks for the question.**

While the syllabus initially indicated the need to submit a paragraph explaining the use of AI or other resources in your assignments, this requirement no longer applies as we are now utilizing eClass quizzes instead of handwritten submissions. Therefore, you are **not** required to submit any explanation regarding the tools or resources (such as online tools or AI) used in completing this quiz.

This PDF version of the questions has been provided for your convenience should you wish to print them and work offline.

Only answers submitted through the eClass quiz system will be graded. Please do not submit a written copy of your responses.

Question 1. [1 MARK]

Let $\hat{f}_{\text{ERM}}$ be as defined in section 9.1.1 of the course notes. Which of the following is true?

a. We can think of $\hat{f}_{\text{ERM}}(\mathbf{x})$ as predicting the probability of $\mathbf{x}$ belonging to class 1.

b. It is always the case that f_{Bayes} outputs class 1 if $\hat{f}_{\text{ERM}}(\mathbf{x}) \geq 0.5$, and class 0 otherwise.

c. f_{Bayes} is equal to $\hat{f}_{\text{ERM}}$.

d. $\hat{f}_{\text{ERM}}$ has the same closed-form solution as the ERM predictor for linear regression with the squared loss.

Solution:

Answer: a.

a. **True.** The ERM predictor $\hat{f}_{\text{ERM}}(\mathbf{x})$ in logistic regression estimates the probability that a data point $\mathbf{x}$ belongs to class 1. This is because the logistic function $\sigma(\mathbf{x}^\top \mathbf{w})$ outputs values in the range $[0, 1]$, which can be interpreted as probabilities.

b. **False.** In the binary classification setting $f_{\text{Bayes}}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x})$. This is equivalent to saying $f_{\text{Bayes}}(\mathbf{x}) = 1$ if $p(y = 1|\mathbf{x}) \geq 0.5$ and $f_{\text{Bayes}}(\mathbf{x}) = 0$ otherwise. Since $\hat{f}_{\text{ERM}}(\mathbf{x})$ is only an estimate of $p(y = 1|\mathbf{x})$, it is not always the case that $\hat{f}_{\text{ERM}}(\mathbf{x}) = p(y = 1|\mathbf{x})$.

c. **False.** f_{Bayes} outputs class labels, while $\hat{f}_{\text{ERM}}$ outputs probabilities. The two are not the same.

d. **False.** The ERM predictor for logistic regression does not have a closed-form solution and typically requires iterative optimization algorithms such as gradient descent. On the other hand, linear regression with squared loss has a closed-form solution.

Question 2. [1 MARK]

In class we used the following function class for logistic regression:

$$\mathcal{F} = \left\{ f \mid f : \mathbb{R}^{d+1} \to [0,1], \text{ where } f(\mathbf{x}) = \sigma(\mathbf{x}^\top \mathbf{w}), \text{ and } \mathbf{w} \in \mathbb{R}^{d+1} \right\}.$$

Suppose that we would like to use a larger function class that contains polynomial features of the input $\mathbf{x}$. Recall that $\phi_p(\mathbf{x})$ is the degree p polynomial feature map of $\mathbf{x}$. We define the new function class as follows

$$\mathcal{F}_p = \left\{ f \mid f : \mathbb{R}^{d+1} \to [0,1], \text{ where } f(\mathbf{x}) = \sigma(\phi_p(\mathbf{x})^\top \mathbf{w}), \text{ and } \mathbf{w} \in \mathbb{R}^{\bar{p}} \right\},$$

where $\bar{p} = \binom{d+p}{p}$ is the number of features in the polynomial feature map $\phi_p(\mathbf{x})$. Is the following statement true or false? For all $p \in \{2, \dots\}$ it holds that $\mathcal{F} \subset \mathcal{F}_p \subset \mathcal{F}_{p+1}$.

Solution:

Answer: True.
Explanation: $\mathcal{F}$ is equivalent to $\mathcal{F}_1$ since it uses linear features. The function class $\mathcal{F}_p$ is a subset of $\mathcal{F}_{p+1}$ because polynomial features of degree p are included within those of degree $p+1$. Therefore, for each p, it follows that $\mathcal{F} \subset \mathcal{F}_p \subset \mathcal{F}_{p+1}$.

Question 3. [1 MARK]

Let everything be as defined in the previous question. Let $\hat{f}_{\text{ERM},p}(\mathbf{x}) = \sigma(\phi_p(\mathbf{x})^\top \mathbf{w}_{\text{ERM},p})$ be the ERM predictor for the function class $\mathcal{F}_p$, where $\mathbf{w}_{\text{ERM},p}$ is the minimizer of the estimated loss (with the binary cross-entropy loss function). Is the following statement true or false? The binary predictor $\hat{f}_{\text{Bin}}$ that outputs class 1 if $\hat{f}_{\text{ERM},p}(\mathbf{x}) \geq 0.5$ and class 0 otherwise can be equivalently defined as

$$\hat{f}_{\text{Bin}}(\mathbf{x}) = \begin{cases} 1 & \text{if } \phi_p(\mathbf{x})^\top \mathbf{w}_{\text{ERM},p} \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

Solution:

Answer: True.
Explanation: The sigmoid function $\sigma(z)$ satisfies $\sigma(z) \geq 0.5$ if and only if $z \geq 0$. As a result, $\hat{f}_{\text{ERM},p}(\mathbf{x}) \geq 0.5$ is equivalent to $\phi_p(\mathbf{x})^\top \mathbf{w}_{\text{ERM},p} \geq 0$. Thus, the two definitions of $\hat{f}_{\text{Bin}}$ are equivalent.

Question 4. [1 MARK]

In class we worked out the MLE solution for binary classification. In this question we are going to work out the MAP solution. Suppose that the setting is the same as in the MLE setting defined in section 9.1 of the course notes. However, we will also assume that the weights $w_1^*, \dots, w_d^*$ are i.i.d. Gaussian random variables with mean 0 and variance $1/\lambda$. The bias term w_0^* is also independent of the other weights and has a uniform distribution on $[-a, a]$, for a very large a. Which of the following is equal to $\mathbf{w}_{\text{MAP}} = \arg\max_{\mathbf{w} \in \mathbb{R}^{d+1}} p(\mathbf{w} \mid \mathcal{D})$?

a.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n} \left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 \right]$$

b.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\frac{\lambda}{2}\sum_{i=1}^{n} \left(\left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) \left(\sum_{j=1}^{d} w_j^2\right)\right)\right]$$

c.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[\sum_{i=1}^{n} \left(y_i - \sigma(\mathbf{x}_i^\top \mathbf{w})\right)^2 + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 \right]$$

d.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n} \left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) + \lambda\sum_{j=1}^{d} w_j \log(w_j) \right]$$

Solution:

Answer: a.
To find the MAP estimate $\mathbf{w}_{\text{MAP}}$, we maximize the posterior probability $p(\mathbf{w} \mid \mathcal{D})$, which is proportional to the product of the likelihood and the prior:

$$p(\mathbf{w} \mid \mathcal{D}) \propto p(\mathcal{D} \mid \mathbf{w})\, p(\mathbf{w}).$$

Likelihood:
For binary classification using logistic regression, the likelihood is:

$$p(\mathcal{D} \mid \mathbf{w}) = \prod_{i=1}^{n} \left[\sigma(\mathbf{x}_i^\top \mathbf{w})^{y_i} \left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)^{1-y_i} \right],$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function.
Taking the logarithm, we get the log-likelihood:

$$\log p(\mathcal{D} \mid \mathbf{w}) = \sum_{i=1}^{n} \left[y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right].$$

Prior:
The prior for w_j (for $j = 1, \ldots, d$) is:

$$p(w_j) = \frac{\sqrt{\lambda}}{\sqrt{2\pi}} \exp\left(-\frac{\lambda}{2} w_j^2\right).$$

Since w_0 has a uniform prior over a very large interval, it contributes a constant to the posterior and can be ignored in optimization.
The log-prior is:

$$\log p(\mathbf{w}) = -\frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 + \text{const.}$$

Posterior:
Combining the log-likelihood and log-prior, the log-posterior (up to a constant) is:

$$\begin{aligned}\log p(\mathbf{w} \mid \mathcal{D}) &= \log p(\mathcal{D} \mid \mathbf{w}) + \log p(\mathbf{w}) \\ &= \sum_{i=1}^{n}\left[y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right] - \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 + \text{const.}\end{aligned}$$

MAP Estimate:
To find $\mathbf{w}_{\text{MAP}}$, we minimize the negative log-posterior:

$$\begin{aligned}\mathbf{w}_{\text{MAP}} &= \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\log p(\mathbf{w} \mid \mathcal{D})\right] \\ &= \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n}\left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2\right].\end{aligned}$$

Question 5. [1 MARK]

In this question, we are going to work out the MAP solution for binary classification using a Laplace prior. Suppose that the setting is the same as in the MLE setting defined in section 9.1 of the course notes. However, we will also assume that the weights $w_1^*, \ldots, w_d^*$ are i.i.d. Laplace random variables with mean 0 and scale parameter $1/\lambda$. The bias term w_0^* is also independent of the other weights and has a uniform distribution on $[-a, a]$ for a very large a.
Which of the following is equal to $\mathbf{w}_{\text{MAP}} = \arg\max_{\mathbf{w}\in\mathbb{R}^{d+1}} p(\mathbf{w} \mid \mathcal{D})$?

a.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n}\left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) + \lambda\sum_{j=1}^{d} |w_j|\right]$$

b.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n}\left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) + \lambda\sum_{j=0}^{d} |w_j|\right]$$

c.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[\sum_{i=1}^{n}\left(y_i - \sigma(\mathbf{x}_i^\top \mathbf{w})\right)^2 + \lambda\sum_{j=1}^{d} |w_j|\right]$$

d.
$$\arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n}\left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1-y_i)\log\left(1-\sigma(\mathbf{x}_i^\top \mathbf{w})\right)\right) + \lambda\sum_{j=1}^{d} \log(|w_j|)\right]$$

Solution:

Answer: a.
To find the MAP estimate $\mathbf{w}_{\text{MAP}}$, we need to maximize the posterior probability $p(\mathbf{w} \mid \mathcal{D})$, which is proportional to the product of the likelihood and the prior:

$$p(\mathbf{w} \mid \mathcal{D}) \propto p(\mathcal{D} \mid \mathbf{w})\, p(\mathbf{w}).$$

Likelihood:
For binary classification using logistic regression, the likelihood is:

$$p(\mathcal{D} \mid \mathbf{w}) = \prod_{i=1}^{n} \left[\sigma(\mathbf{x}_i^\top \mathbf{w})^{y_i} \left(1 - \sigma(\mathbf{x}_i^\top \mathbf{w})\right)^{1-y_i} \right],$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function.
Taking the logarithm, we obtain the log-likelihood:

$$\log p(\mathcal{D} \mid \mathbf{w}) = \sum_{i=1}^{n} \left[y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1 - y_i) \log\left(1 - \sigma(\mathbf{x}_i^\top \mathbf{w})\right) \right].$$

Prior:
The prior for each w_j (for $j = 1, \ldots, d$) is a Laplace distribution:

$$p(w_j) = \frac{\lambda}{2} \exp\left(-\lambda |w_j|\right).$$

Since w_0 has a uniform prior over a very large interval, its contribution to the posterior is constant and can be ignored during optimization.
The log-prior is therefore:

$$\log p(\mathbf{w}) = -\lambda \sum_{j=1}^{d} |w_j| + \text{const.}$$

Posterior:
Combining the log-likelihood and log-prior, the log-posterior (up to a constant) is:

$$\begin{aligned} \log p(\mathbf{w} \mid \mathcal{D}) &= \log p(\mathcal{D} \mid \mathbf{w}) + \log p(\mathbf{w}) \\ &= \sum_{i=1}^{n} \left[y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1 - y_i) \log\left(1 - \sigma(\mathbf{x}_i^\top \mathbf{w})\right) \right] - \lambda \sum_{j=1}^{d} |w_j| + \text{const.} \end{aligned}$$

MAP Estimate:
To find $\mathbf{w}_{\text{MAP}}$, we minimize the negative log-posterior:

$$\begin{aligned} \mathbf{w}_{\text{MAP}} &= \arg\min_{\mathbf{w} \in \mathbb{R}^{d+1}} \left[-\log p(\mathbf{w} \mid \mathcal{D}) \right] \\ &= \arg\min_{\mathbf{w} \in \mathbb{R}^{d+1}} \left[-\sum_{i=1}^{n} \left(y_i \log\left(\sigma(\mathbf{x}_i^\top \mathbf{w})\right) + (1 - y_i) \log\left(1 - \sigma(\mathbf{x}_i^\top \mathbf{w})\right) \right) + \lambda \sum_{j=1}^{d} |w_j| \right]. \end{aligned}$$

Question 6. [1 MARK]

Suppose that things are as defined in the previous question. However, we assume the weights $w_0^*, w_1^*, \ldots, w_d^*$ are all i.i.d. uniform random variables on $[-a, a]$ for a very large a. Is the following statement true or false? The MAP solution with this prior is equivalent to the MLE solution.

Solution:

Answer: True.
Explanation: Given that each weight w_j (for $j = 0, 1, \ldots, d$) is uniformly distributed over $[-a, a]$, the prior probability is:

$$p(\mathbf{w}) = p(w_0) \cdot p(w_1) \cdots p(w_d) = \left(\frac{1}{2a}\right)^{d+1}$$

This prior is constant with respect to $\mathbf{w}$.
The MAP estimate maximizes the posterior probability:

$$\mathbf{w}_{\text{MAP}} = \arg\max_{\mathbf{w}} p(\mathbf{w} \mid \mathcal{D}) = \arg\max_{\mathbf{w}} p(\mathcal{D} \mid \mathbf{w})\, p(\mathbf{w})$$

Since $p(\mathbf{w})$ is constant, maximizing the posterior is equivalent to maximizing the likelihood:

$$\mathbf{w}_{\text{MAP}} = \arg\max_{\mathbf{w}} p(\mathcal{D} \mid \mathbf{w}) = \mathbf{w}_{\text{MLE}}$$

Therefore, the MAP estimate coincides with the MLE estimate when using a uniform prior over a very large interval.

Question 7. [1 MARK]

Let $\hat{f}_{\text{Mul}}$ be as defined in section 9.2 of the course notes. Which of the following is true?

a. $\hat{f}_{\text{Mul}}$ outputs a vector of probabilities where the y-th element is the probability of class y.

b. $\sigma(\mathbf{x}^\top \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{x}^\top \mathbf{w}_{\text{MLE},K-1})$ outputs a vector of probabilities where the y-th element is the probability of class y.

c. $\hat{f}_{\text{ERM}}$ as defined in section 9.2.1 of the course notes is approximately equal to f_{Bayes}.

d. There is no closed-form solution for $\mathbf{w}_{\text{MLE},k}$ for any k.

Solution:

Answer: b., d.

a. **False.** $\hat{f}_{\text{Mul}}(\mathbf{x}) \in \mathcal{Y}$, and $\mathcal{Y}$ is not a set of vectors.

b. **True.** By the definition of the MLE solution.

c. **False.** $\hat{f}_{\text{ERM}}$ is a vector of probability estimates, while f_{Bayes} outputs class labels.

d. **True.** Mentioned in the course notes.

Question 8. [1 MARK]

Let everything be as defined in the previous question. Which of the following is true?

a. If $\mathbf{x}^\top \mathbf{w}_{\text{MLE},y} < \mathbf{x}^\top \mathbf{w}_{\text{MLE},k}$ for all $k \neq y$, then $\hat{f}_{\text{Mul}}(\mathbf{x}) = y$.

b. If $\mathbf{x}^\top \mathbf{w}_{\text{MLE},y} > \mathbf{x}^\top \mathbf{w}_{\text{MLE},k}$ for all $k \neq y$, then $\hat{f}_{\text{Mul}}(\mathbf{x}) = y$.

c. If $\sigma_y(\mathbf{x}^\top \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{x}^\top \mathbf{w}_{\text{MLE},K-1}) > 0.5$, then $\hat{f}_{\text{Mul}}(\mathbf{x}) = y$.

d. If $\mathbf{x}^\top (\mathbf{w}_{\text{MLE},y} - \mathbf{w}_{\text{MLE},k}) = 0$, then

$$p(y \mid \mathbf{x}, \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{w}_{\text{MLE},K-1}) = p(k \mid \mathbf{x}, \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{w}_{\text{MLE},K-1}).$$

Solution:

Answer: b., c., d.

a. **False.** See explanation for b.

b. **True.** $\hat{f}_{\text{Mul}}(\mathbf{x})$ is the y with the largest value of $\sigma_y(\mathbf{x}^\top \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{x}^\top \mathbf{w}_{\text{MLE},K-1})$, which is the same as the y with the largest value of $\mathbf{x}^\top \mathbf{w}_{\text{MLE},y}$.

c. **True.** If $\sigma_y(\mathbf{x}^\top \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{x}^\top \mathbf{w}_{\text{MLE},K-1}) > 0.5$, then

$$\sigma_y(\mathbf{x}^\top \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{x}^\top \mathbf{w}_{\text{MLE},K-1}) > \sigma_k(\mathbf{x}^\top \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{x}^\top \mathbf{w}_{\text{MLE},K-1})$$

for all $k \neq y$, so $\hat{f}_{\text{Mul}}(\mathbf{x}) = y$.

d. **True.** If $\mathbf{x}^\top (\mathbf{w}_{\text{MLE},y} - \mathbf{w}_{\text{MLE},k}) = 0$ then $\mathbf{x}^\top \mathbf{w}_{\text{MLE},y} = \mathbf{x}^\top \mathbf{w}_{\text{MLE},k}$. If you plug this into the definition of $p(y \mid \mathbf{x}, \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{w}_{\text{MLE},K-1})$ you will see that it is equal to $p(k \mid \mathbf{x}, \mathbf{w}_{\text{MLE},0}, \ldots, \mathbf{w}_{\text{MLE},K-1})$.

Question 9. [1 MARK]

Let $\mathcal{Y} = \{0, 1\}$ be the set of labels. Define the following two label functions:

$$h(y) = \begin{cases} (1,0)^\top & \text{if } y = 0, \\ (0,1)^\top & \text{if } y = 1. \end{cases}$$

$$r(\hat{y}) = (1 - \hat{y}, \hat{y})^\top.$$

Is the following statement true or false? For any $\hat{y} \in (0, 1)$ and $y \in \mathcal{Y}$ (where $(0, 1)$ is the open interval from 0 to 1), the binary cross-entropy loss with input $\hat{y}$ and y is equal to the multiclass cross-entropy loss with input $r(\hat{y})$ and $h(y)$.

Solution:

Answer: True.
Explanation: To verify the equivalence of the binary cross-entropy loss and the multiclass cross-entropy loss under the given label mappings $h(y)$ and $r(\hat{y})$, we analyze both loss functions in the context of the provided definitions.
The binary cross-entropy loss for binary classification with labels $y \in \{0, 1\}$ and predictions $\hat{y} \in (0, 1)$ is defined as:

$$\ell_{\text{binary}}(\hat{y}, y) = -\left[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\right].$$

On the other hand, the multiclass cross-entropy loss for multiclass classification with one-hot encoded labels $h(y) \in \{(1,0)^\top, (0,1)^\top\}$ and predictions $r(\hat{y}) \in (0,1)^2$ is defined as:

$$\ell_{\text{multiclass}}(r(\hat{y}), h(y)) = -\sum_{j=1}^{2} h_j(y) \log(r_j(\hat{y})),$$

where $h_j(y)$ and $r_j(\hat{y})$ denote the j-th components of the vectors $h(y)$ and $r(\hat{y})$, respectively.
Given the label mapping:

$$h(y) = \begin{cases} (1,0)^\top & \text{if } y = 0, \\ (0,1)^\top & \text{if } y = 1, \end{cases}$$

and the prediction mapping:

$$r(\hat{y}) = (1 - \hat{y}, \hat{y})^\top,$$

we can simplify the multiclass cross-entropy loss based on the value of y:
Case 1: $y = 0$
When $y = 0$, the label mapping yields $h(y) = (1,0)^\top$. Substituting into the multiclass loss function:

$$\ell_{\text{multiclass}}(r(\hat{y}), h(0)) = -\left[h_1(0) \log(r_1(\hat{y})) + h_2(0) \log(r_2(\hat{y}))\right] = -\log(r_1(\hat{y})).$$

Given $r(\hat{y}) = (1 - \hat{y}, \hat{y})^\top$, we have $r_1(\hat{y}) = 1 - \hat{y}$. Therefore:

$$\ell_{\text{multiclass}}(r(\hat{y}), h(0)) = -\log(1 - \hat{y}) = \ell_{\text{binary}}(\hat{y}, 0).$$

Case 2: $y = 1$
When $y = 1$, the label mapping yields $h(y) = (0,1)^\top$. Substituting into the multiclass loss function:

$$\ell_{\text{multiclass}}(r(\hat{y}), h(1)) = -\left[h_1(1) \log(r_1(\hat{y})) + h_2(1) \log(r_2(\hat{y}))\right] = -\log(r_2(\hat{y})).$$

Given $r(\hat{y}) = (1 - \hat{y}, \hat{y})^\top$, we have $r_2(\hat{y}) = \hat{y}$. Therefore:

$$\ell_{\text{multiclass}}(r(\hat{y}), h(1)) = -\log(\hat{y}) = \ell_{\text{binary}}(\hat{y}, 1).$$

In both cases, the multiclass cross-entropy loss $\ell_{\text{multiclass}}(r(\hat{y}), h(y))$ simplifies to the binary cross-entropy loss $\ell_{\text{binary}}(\hat{y}, y)$. This demonstrates that under the specified label and prediction mappings, the two loss functions yield identical values for all $y \in \mathcal{Y}$ and $\hat{y} \in (0,1)$.

Question 10. [1 MARK]

Suppose you are in the binary classification setting as defined in section 9.1 of the course notes and you solve for $\mathbf{w}_{\text{MLE}}$. Now suppose that the setting is the multiclass classification setting (with $K = 2$) as defined in section 9.2 of the course notes and you solve for $\mathbf{w}_{\text{MLE},0}$, $\mathbf{w}_{\text{MLE},1}$. Is the following true or false? The solution for $\mathbf{w}_{\text{MLE}}$ in the binary classification setting is the same as the solution for $\mathbf{w}_{\text{MLE},1}$ in the multiclass classification setting.

Solution:

Answer: False. As shown in section 9.2.2 of the course notes $\mathbf{w}_{\text{MLE}} = \mathbf{w}_{\text{MLE},1} - \mathbf{w}_{\text{MLE},0}$.

Question 11. [1 MARK]

You are designing a neural network architecture for a binary classification problem. You decide to have $B = 5$ layers and $d^{(1)} = 50$, $d^{(2)} = 40$, $d^{(3)} = 30$, $d^{(4)} = 20$, $d^{(5)} = 1$ neurons in each layer respectively. The input dimension is $d = 100$. How many weight vectors are there in the network?

Solution:

Answer: 141
Explanation: Each activation (other than the biases and the zeroth layer) has a weight vector associated with it. Thus there are

$$50 + 40 + 30 + 20 + 1 = 141$$

weight vectors in the network.

Question 12. [1 MARK]

Let everything be as defined in the previous question. If you sum up the dimension of all the weight vectors in the neural network you get the number of weights in the network. How many weights are there in the network?

Solution:

Answer: 8961
Explanation:
First layer: $50 \cdot 101 = 5050$ weights (50 neurons with 100 input features and 1 bias term).
Second layer: $40 \cdot 51 = 2040$ weights (40 neurons with 50 input features and 1 bias term).
Third layer: $30 \cdot 41 = 1230$ weights (30 neurons with 40 input features and 1 bias term).
Fourth layer: $20 \cdot 31 = 620$ weights (20 neurons with 30 input features and 1 bias term).
Fifth layer: $1 \cdot 21 = 21$ weights (1 neuron with 20 input features and 1 bias term).
Total number of weights: $5050 + 2040 + 1230 + 620 + 21 = 8961$.

Question 13. [1 MARK]

The tanh function is defined as $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, where $z \in \mathbb{R}$. Is the following statement true or false? The tanh function is a valid activation function?

Solution:

Answer: True.
Explanation: The tanh function is a valid activation function because it is differentiable everywhere and is a function from $\mathbb{R}$ to $(-1, 1) \subset \mathbb{R}$.

Question 14. [1 MARK]

The logistic function is defined as $\sigma(z) = \frac{1}{1+e^{-z}}$, where $z \in \mathbb{R}$. You would like to define the function $f(\mathbf{x}) = \sigma(\mathbf{x}^\top \mathbf{w})$ where $\mathbf{x} \in \mathbb{R}^{d+1}, \mathbf{w} \in \mathbb{R}^{d+1}$ as a neural network. Which of the following is correct?

a. The neural network has $B = 2$ layers, $d^{(1)} = 1, d^{(2)} = 1$ neurons, activation $h = \sigma$, and two weight vectors $\mathbf{w}_1^{(1)} = \mathbf{w}, \mathbf{w}_1^{(2)} = (1, 1)^\top$.

b. The neural network has $B = 1$ layer, $d^{(1)} = 1$ neuron, activation $h = \sigma$, and one weight vector $\mathbf{w}_1^{(1)} = \mathbf{w}$.

c. The neural network has $B = 1$ layer, $d^{(1)} = 1$ neuron, activation $h(z) = z$, and one weight vector $\mathbf{w}_1^{(1)} = \mathbf{w}$.

d. The neural network has $B = 2$ layers, $d^{(1)} = 1, d^{(2)} = 1$ neurons, activation $h = \sigma$ in the first layer, activation $h(z) = z$ in the second layer, and two weight vectors $\mathbf{w}_1^{(1)} = \mathbf{w}, \mathbf{w}_1^{(2)} = (0, 1)^\top$.

Solution:

Answer: b., d.

a. **False.** The neural network would output $f(\mathbf{x}) = \sigma((1, \sigma(\mathbf{x}^\top \mathbf{w}))^\top (1,1)^\top) = \sigma(1 + \sigma(\mathbf{x}^\top \mathbf{w})) \neq \sigma(\mathbf{x}^\top \mathbf{w})$.

b. **True.** The neural network would output $f(\mathbf{x}) = \sigma(\mathbf{x}^\top \mathbf{w})$.

c. **False.** The neural network would output $f(\mathbf{x}) = \mathbf{x}^\top \mathbf{w}$.

d. **True.** The neural network would output $f(\mathbf{x}) = (1, \sigma(\mathbf{x}^\top \mathbf{w}))^\top (0,1)^\top = \sigma(\mathbf{x}^\top \mathbf{w})$.

Question 15. [1 MARK]

You have a neural network f with $B = 2$ layers and $d^{(1)} = 3$, $d^{(2)} = 1$ neurons in each layer respectively. The input dimension is $d = 2$. You choose to use the ReLU activation function, defined as $\text{ReLU}(z) = \max(0, z)$, where $z \in \mathbb{R}$. The weight vectors have the following values:

$$\mathbf{w}_1^{(1)} = (1,1,1)^\top \quad \mathbf{w}_2^{(1)} = (-1,-1,-1)^\top \quad \mathbf{w}_3^{(1)} = (-1,0,1)^\top \quad \mathbf{w}_1^{(2)} = (1,1,1,1)^\top$$

Suppose you get a feature vector $\mathbf{x} = (1,-1,1)^\top$. What is $f(\mathbf{x})$?

Solution:

Answer: 2
Explanation: The activations for the first layer are:

$$a_1^{(1)} = \text{ReLU}(\mathbf{x}^\top \mathbf{w}_1^{(1)}) = \text{ReLU}(1 \cdot 1 + (-1) \cdot 1 + 1 \cdot 1) = \text{ReLU}(1) = 1$$

$$a_2^{(1)} = \text{ReLU}(\mathbf{x}^\top \mathbf{w}_2^{(1)}) = \text{ReLU}(1 \cdot (-1) + (-1) \cdot (-1) + 1 \cdot (-1)) = \text{ReLU}(-1) = 0$$

$$a_3^{(1)} = \text{ReLU}(\mathbf{x}^\top \mathbf{w}_3^{(1)}) = \text{ReLU}(1 \cdot (-1) + (-1) \cdot 0 + 1 \cdot 1) = \text{ReLU}(0) = 0$$

Thus, the activation vector for the first layer is $\mathbf{a}^{(1)} = (1,1,0,0)^\top$.
The activations for the second layer are:

$$a_1^{(2)} = \text{ReLU}\left(\left(\mathbf{a}^{(1)}\right)^\top \mathbf{w}_1^{(2)}\right) = \text{ReLU}(1 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1) = \text{ReLU}(2) = 2$$

Thus, $f(\mathbf{x}) = a_1^{(2)} = 2$.