# CS760 Midterm Practice Solutions

October 26, 2021

## 1 True/False

- Unsupervised learning methods deal with instances without labels, and can reveal patterns of the data.
  $\boxed{Correct}$
- In cross validation, we train the classifier using all of the data, and predict the classification of the left-out set.
  $\boxed{Incorrect}$
  - In cross validation, we train the classifier using all but one fold of the data. Please refer to the lecture slides: evaluating learning algorithms part 1 - Page 13.
- High capacity models are more likely to overfit.
  $\boxed{Correct}$
- A fully connected feedforward neural network has input $\mathbb{R}^2$ , a first hidden ReLU layer with 4 units, a second hidden ReLU layer with 3 units, and a single sigmoid output unit. The number of parameters in the neural network (including offset weights) = 30.
  $\boxed{Incorrect}$
  - Consider a fully connected feedforward neural network has input $\mathbb{R}^d$, a first hidden ReLU layer with $h_1$ units, a second hidden ReLU layer with $h_2$ units, and a single sigmoid output unit. Each hidden and output unit has an offset parameter. Input to first hidden layer: $(1 + d)h_1$. First to second hidden layer: $(1 + h_1)h_2$. To output: $1 + h_2$. The total is the sum of these i.e. $(1 + d)h_1 + (1 + h_1)h_2 + 1 + h_2 = 3 * 4 + 5 * 3 + 4 = 31$

## 2 Neural Networks

(a) Derive the derivative of ReLU activation function function $ReLU(x) = max(0, x)$. (5 pts)

Sol)

$$\frac{\partial}{\partial x} ReLU(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x < 0 \end{cases}$$

And, undefined at x=0 since left derivative and right derivative are different.

(b) Derive the derivative of hyperbolic tangent activation function $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. (5 pts)

Sol)

$$\frac{\partial}{\partial x} tanh(x) = \frac{(e^x + e^{-x})(e^x + e^{-x}) - (e^x - e^{-x})(e^x - e^{-x})}{(e^x + e^{-x})^2} \tag{1}$$

$$= 1 - \frac{(e^x - e^{-x})(e^x - e^{-x})}{(e^x + e^{-x})^2} \tag{2}$$

$$= 1 - tanh^2(x) \tag{3}$$

(c) Compare sigmoid, hyperbolic tangent, and ReLU activation functions. (5 pts)

- $tanh(x) = 2\sigma(2x) - 1$
- ReLU has a non-differentiable point, but sigmoid and hyperbolic tangent are differentiable at all points.
- Sigmoid and hyperbolic tangent suffer from the vanishing gradient problem, but ReLU does not.
- They have different ranges.

# 3 Evaluation metrics

Suppose you trained a classifier for a spam filter. The prediction result on the test set is summarized in the following table. Here, "+" represents spam, and "-" means not spam.

| Confidence positive | Correct class | Predicted class |
|---|---|---|
| 0.95 | + | + |
| 0.85 | + | + |
| 0.8 | - | + |
| 0.7 | + | + |
| 0.55 | + | + |
| 0.45 | - | - |
| 0.4 | + | - |
| 0.3 | + | - |
| 0.2 | - | - |
| 0.1 | - | - |

Set the threshold for spam as 0.5 (i.e. if confidence positive is more than 0.5, it is classified as +). Calculate the following evaluation metrics.

Sol) TP: 4, TN: 3, FP: 1, FN: 2

- Accuracy (2.5 pts) **0.7**
- False positive rate (2.5 pts) **FP/(FP+TN)=1/4**
- Precision (2.5 pts) **TP/(TP+FP)=4/5**
- Recall (2.5 pts) **TP/(TP+FN)=2/3**

# Problem 4

Suppose you have $n$ samples drawn $\{x_i\}_{i=1}^n$ i.i.d from a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$. What is the negative log-likelihood (NLL) of these samples? Derive the MLE (Maximum Likelihood Estimator ) for $\mu$ and $\sigma$ using these samples. Comment on the behaviour of your estimators when $n \to \infty$. What can you say about the convexity or strong convexity of the NLL function.
**answer:** see notes here
`http://jrmeyer.github.io/machinelearning/2017/08/18/mle.html` This function is twice differentiable so you can take double derivative and see the conditions for convexity, strong convexity.

# Problem 5

Let's say you have 3 points from a 3 dimensional space, namely $\mathbf{x}_1 = [2, -1, 0], \mathbf{x}_2 = [-1, 1, 1], \mathbf{x}_3 = [0, 1, 0]$. Also let, $y_1 = 0, y_2 = 5, y_3 = 2$. Assume there is some true $\mathbf{w} \in \mathbb{R}^3$ such that $\mathbf{w}^T \mathbf{x}_i = y_i$.

1. Write expression for closed form solution for $\mathbf{w}$. Does it exist? If yes, calculate it.
   **Answer**: $\mathbf{w} = \mathbf{X}^{-1}\mathbf{y} = [1, 2, 4]$ because $\mathbf{X}$ is invertible. If it was not the case but $\mathbf{X}^T\mathbf{X}$ was invertible then $\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}\mathbf{y} = [1, 2, 4]$

2. Do at least 2 steps of gradient descent (manually). Use learning rate 0.02 and initialize $\mathbf{w}$ to $[0, 1, 0]$. At each step compare the current estimate by gradient descent to the closed form solution.
   **Answer**: $\ell(\mathbf{w}) = \frac{1}{2}||\mathbf{X}\mathbf{w} - \mathbf{y}||_2^2$, We want to minimize this function using gradient descent. Using the above information we get the following iterates. Which are converging towards the optimal solution.

$$[-0.20, 1.40, 0.40]$$

$$[-0.14, 1.58, 0.70]$$

$$[-0.02, 1.69, 0.95]$$

$$[0.09, 1.78, 1.19]$$

$$[0.19, 1.85, 1.40]$$

$$[0.29, 1.91, 1.59]$$

# Problem 6

Your friend has a biased coin i.e. probability of head is $\theta \neq 1/2$. You agreed to play a game with your friend. Your friend tosses the coin and you need to guess whats on the outcome of the toss. You are overly confident about your predictions so you agreed to the conditions that you will pay \$1 if you are wrong otherwise nobody pays anything.

1. Without knowing $\theta$, what will be your strategy? What will be the expected amount of money you will be paying to your friend.
   **answer**: In this case the best strategy would be to predict randomly i.e. probability of head and tail $=1/2$. Denote your prediction by $\hat{x}$ and true outcome as $x$. Then the expected loss,

$$\begin{aligned}
\mathbb{E}_{x,\hat{x}}[\mathbb{1}(\hat{x} \neq x)] &= Pr(x = H \wedge \hat{x} = T) + Pr(x = T \wedge \hat{x} = T) \\
&= \theta/2 + (1 - \theta)/2 \\
&= 1/2
\end{aligned}$$

   So with this strategy, you will loose $1/2$ dollar in expectation in each round. ( Think of the extreme case when $\theta$=1) .

2. Can you estimate $\theta$ while playing the game? What will be the most likely estimate of $\theta$ after $n$ rounds of play. What happens to your estimate if you keep playing the game forever. ( You might go bankrupt though )

   **answer**: It will be a standard setting of MLE estimation for Bernoulli distribution. We can denote $x = 1$ when it is head and $x = 0$ otherwise and do MLE estimation for distribution $p_\theta(x) = \theta^x(1-\theta)^{(1-\mathbf{x})}$ using samples $\{x_i\}_{i=1}^n$. The estimate will be $\hat{\theta}_n = \frac{1}{n}\sum x_i$. If we keep playing this game forever, i.e. let $n \to \infty$ then by law of large numbers $\hat{\theta}_n \to \theta$.

3. Suppose your friend's friend told you the $\theta$, what will be your strategy? If you use this strategy what will be your expected loss?

   **answer**: You have to two options that can minimize your loss. Either predict $\hat{x}$ with $p_\theta(x)$, this will have expected loss $= 2\theta(1-\theta)$ or predict $\hat{x} = T$ if $\theta < 1/2$ otherwise predict $\hat{x} = H$. This strategy will have expected loss $(1 - \max\{\theta, 1-\theta\})$. The second one turns out to be the optimal one, after comparing both of them for $\theta > 1/2$ and $\theta < 1/2$ cases.

4. By now may be you know you are loosing a lot of money without knowing $\theta$. Instead of telling the true $\theta$, your friend's friend told you that the true $\theta$ follows a Beta distribution with parameters $\alpha, \beta$. How will you use this prior knowledge to improve your estimation? Derive the estimator using this prior knowledge and $n$ rounds of play.
   **answer**: See these slides for detailed analysis `http://www.mi.fu-berlin.de/wiki/pub/ABI/Genomics12/MLvsMAP.pdf`