CIS 700-004 Spring 2019 **Homework 0 Solutions**

This is a review of machine learning designed as a memory refresher;f not all concepts have been discussed in class. Feel free to consult any reference materials.

1. **[10 pts]** **Understanding Entropy**

   This problem reviews Shannon's informational entropy. Please use log_2 for all computations.

   (a) Compute the entropy of the following distribution.

$$X = \begin{cases} 1 & \text{w.p. } 0.3 \\ 0 & \text{w.p. } 0.7 \end{cases}$$

   (b) Compute the entropy of the following distribution.

$$X = \begin{cases} 1 & \text{w.p. } 0.6 \\ 0 & \text{w.p. } 0.4 \end{cases}$$

   (c) Compute the entropy of the following distribution.

$$X = \begin{cases} 1 & \text{w.p. } 0.998 \\ 0 & \text{w.p. } 0.002 \end{cases}$$

   (d) Note that increasing $\Pr[X = 1]$ monotonically from 0.3 to 0.6 to 0.998 does not increase entropy monotonically. Assuming two outcomes, what distribution of X maximizes entropy?

   (e) In your own words, relate entropy to the concept of surprise (aka information content).

Solution:

(a)

$$\begin{aligned} H(X) &= -0.3\log_2(0.3) - 0.7\log_2(0.7) \\ &= \boxed{0.881} \end{aligned}$$

(b)

$$\begin{aligned} H(X) &= -0.6\log_2(0.6) - 0.4\log_2(0.4) \\ &= \boxed{0.971} \end{aligned}$$

(c)

$$\begin{aligned} H(X) &= -0.998 \log_2(0.998) - 0.002 \log_2(0.002) \\ &= \boxed{0.0208} \end{aligned}$$

(d) Entropy is maximized by a coin toss.

(e) Entropy is expected surprise over the distribution of outcomes.

---

2. **[10 pts]** **Understanding Cross-Entropy**

Suppose we have a true distribution of

$$P = \begin{cases} 10 & \text{w.p. } 0.65 \\ 11 & \text{w.p. } 0.25 \\ 12 & \text{w.p. } 0.1 \end{cases}$$

and an estimated distribution of

$$Q = \begin{cases} 10 & \text{w.p. } 0.1 \\ 11 & \text{w.p. } 0.4 \\ 12 & \text{w.p. } 0.5 \end{cases}$$

(a) Compute the cross-entropy $H(P, Q)$.

(b) Suppose we obtain a better estimate Q' s.t.

$$Q' = \begin{cases} 10 & \text{w.p. } 0.6 \\ 11 & \text{w.p. } 0.3 \\ 12 & \text{w.p. } 0.1 \end{cases}$$

Compute the cross-entropy $H(P, Q')$.

(c) Is cross-entropy symmetric? i.e. if we reverse the distributions for P, Q, will we get the same cross-entropy?

(d) Suppose we have a true label $y = [1, 0, 0]$, and a logistic regression model computes estimated probabilities $\hat{y} = [0.91, 0.08, 0.01]$. Compute the cross-entropy loss.

(e) What would happen to the cross-entropy loss if any of the estimated probabilities were zero? Is this possible in a logistic regression?

Solution:

(a)

$$\begin{aligned} H(P, Q) &= -0.65 \log_2(0.1) - 0.25 \log_2(0.4) - 0.1 \log_2(0.5) \\ &= \boxed{2.590} \end{aligned}$$

(b)

$$\begin{aligned} H(P, Q') &= -0.65 \log_2(0.6) - 0.25 \log_2(0.3) - 0.1 \log_2(0.1) \\ &= \boxed{1.245} \end{aligned}$$

(c) No.

(d)

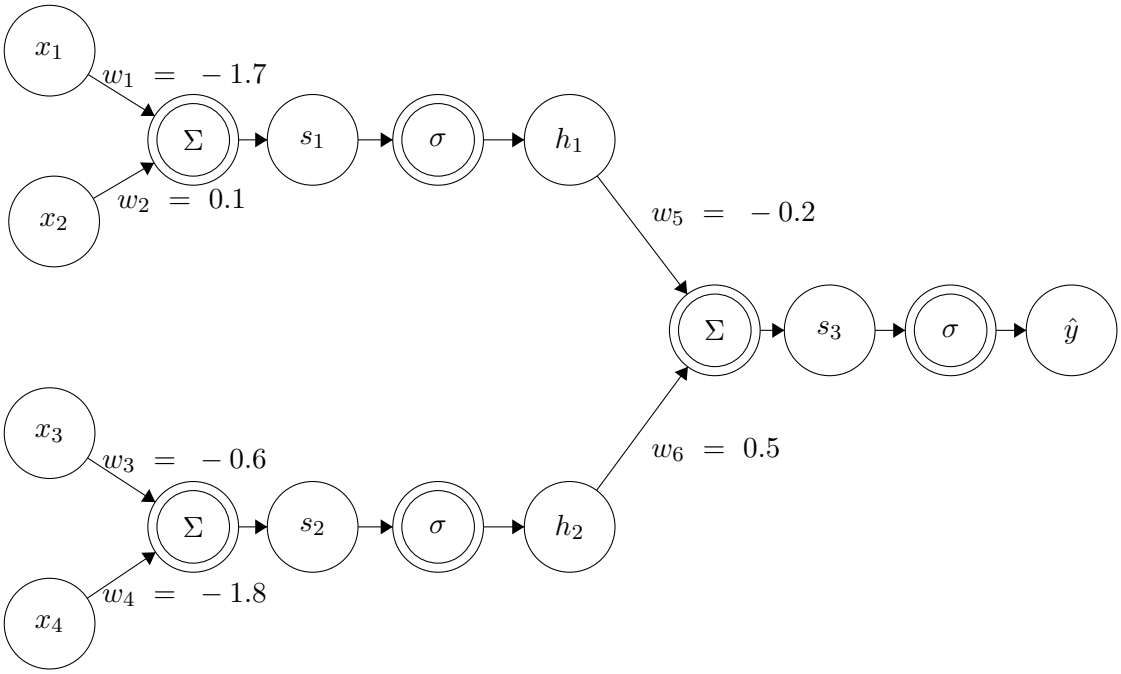
$$\begin{aligned} H(y, \hat{y}) &= -1 \log_2(0.91) \\ &= \boxed{0.136} \end{aligned}$$

(e) The cross-entropy would fall to negative infinity if the corresponding label was nonzero. This is not possible in a logistic regression because the output of a softmax is in the interval $(0, 1)^k$ – the sigmoid function never reaches zero.

---

3. **[10 pts]** **Backpropagation by Hand**

Consider the following neural network. Single-circled nodes denote variables (e.g. x_1 is an input variable, h_1 is an intermediate variable, $\hat{y}$ is an output variable), and double-circled nodes denote functions (e.g. Σ takes the sum of its inputs, and σ denotes the logistic function $\sigma(x) = \frac{1}{1+e^{-x}}$. In the network below, $h_1 = \frac{1}{1+e^{-x_1 w_1 - x_2 w_2}}$.



Suppose we have an L2 loss $L(y, \hat{y}) = ||y - \hat{y}||_2^2$. We are given a data point $(x_1, x_2, x_3, x_4) = (-0.7, 1.2, 1.1, -2)$ with true label 0.5. Use the backpropagation algorithm to compute the partial derivative $\frac{\partial L}{w_1}$.

(Hint: the gradient of an L2 loss function $||\hat{y} - y||_2^2$ is $2||\hat{y} - y||$.)

Solution:

We compute the following on the forward pass:

$$\begin{aligned} s_1 &= 1.31 \\ s_2 &= 2.94 \\ h_1 &= 0.7875 \\ h_2 &= 0.9498 \\ s_3 &= 0.3173 \\ \hat{y} &= 0.5787 \end{aligned}$$

Backpropagation then gives us:

$$\begin{aligned} \frac{\partial E}{\partial w_1} &= \frac{\partial E}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial s_3} \times \frac{\partial s_3}{\partial h_1} \times \frac{\partial h_1}{\partial s_1} \times \frac{\partial s_1}{\partial w_1} \\ &= 2||\hat{y} - y|| \times \sigma'(s_3) \times w_5 \times \sigma'(s_1) \times x_1 \\ &= 2(0.5787 - 0.5) \times 0.2438 \times -0.2 \times 0.1673 \times -0.7 \\ &= \boxed{0.0008988} \end{aligned}$$

4. **[10 pts]** **Other Definitions**

Define the following terms.

(a) Leave-one-out cross validation

(b) Bayes error

(c) Supervised learning

(d) AUC

(e) Bias-variance decomposition

(f) confusion matrix

(g) sparse feature

(h) PCA

(i) epoch

(j) L1 regularization

Solution:

(a) Leave-one-out cross validation

Validation of a model using k-fold validation for $k = 1$. This is typically used for very small datasets, where the loss of even 10% of the data would significantly reduce training efficacy.

(b) Bayes error

Irreducible error; the error of the best possible model over the true distribution of data.

(c) Supervised learning

The learning setting in which examples (x, y) have inputs and true labels.

(d) AUC

The integral of the ROC curve, which plots the true positive rate (TPR) against the false positive rate (FPR). Since the worst case of the ROC is the identity line ($y = x$) and the best case is the constant line ($y = 1$), the AUC is bounded on the interval $[0, 1]$.

(e) Bias-variance decomposition

The linear tradeoff between bias (errors in the form of the learning algorithm) and variance (susceptibility to fluctuations in the input of the model). The mean squared error is equal to the sum of these two quantities and the noise in the data-generating process:

$$E\left[(\theta - \hat{\theta})^2\right] = (Bias[\theta])^2 + Var[\theta] + \sigma^2$$

(f) confusion matrix

A matrix M where element $M[i, j]$ describes the frequency of a model classifying examples of true label i as label j.

(g) sparse feature

A feature where non-zero elements are rare.

(h) PCA

A featurization technique that chooses a new orthogonal basis (principal components) of the original feature space by maximizing the variance of the principal components . PCA is frequently used for dimensionality reduction by truncating to the first k principal components.

(i) epoch

A single pass of the entire dataset through the training of an online or minibatch algorithm. If the training process presents data batches randomly rather than sequentially, an epoch is reached when the size of the training points used (with multiplicity) is equal to the size of the dataset.

(j) L1 regularization

Regularization by adding the L1 norm of the weights to the loss function. This is known as LASSO in the context of regression and functionally reduces weights and selects features simultaneously.