School of Computing and Information Systems
The University of Melbourne
COMP90049 Introduction to Machine Learning (Semester 1, 2023)

Week 9

1. What is Logistic Regression? What is "logistic"? What are we "regressing"?

2. In following dataset each instance represents a news article. The value of the features are counts of selected words in each article. Develop a logistic regression classifier to predict the class of the article (fruit vs. computer). $\hat{y} = 1$ (fruit) and $\hat{y} = 0$ (computer).

ID	*apple*	*ibm*	*lemon*	*sun*	CLASS	
A	1	0	1	5	1	FRUIT
B	1	0	1	2	1	FRUIT
C	2	0	0	1	1	FRUIT
D	2	2	0	0	0	COMPUTER
E	1	2	1	7	0	COMPUTER
T	1	2	1	5	?	

For the moment, we assume that we already have an estimate of the model parameters, i.e., the weights of the 4 features (and the bias θ_0) is $\hat{\theta} = [\theta_0, \theta_1, \theta_2, \theta_3, \theta_4] = [0.2, 0.3, -2.2, 3.3, -0.2]$.

(i). Explain the intuition behind the model parameters in relation to the features.

(ii). Predict the test label.

(iii). Recall the conditional likelihood objective

$$-\log \mathcal{L}(\theta) = -\sum_{i=1}^{n} y_i \log(\sigma(x_i;\theta)) + (1-y_i)\log(1-\sigma(x_i;\theta))$$

Design a test to make sure that the Loss of our model, is lower when its prediction the correct label for test instance T, than when it's predicting a wrong label.

3. For the model created in question 2, compute a single gradient descent update for parameter θ_1 given the training instances given above. Recall that for each feature j, we compute its weight update as

$$\theta_j \leftarrow \theta_j - \eta \sum_i (\sigma(x_i;\theta) - y_i)x_{ij}$$

Summing over all training instances i. We will compute the update for θ_j assuming the current parameters as specified above, and a learning rate $\eta = 0.1$.

4. Consider the following training set:

(x_1, x_2)	y
(0,0)	0
(0,1)	1
(1,1)	1

With the bias value of 1, the initial weight function of $\theta = \{\theta_0, \theta_1, \theta_2\}$ = {0.2, -0.4, 0.1} and learning rate of $\eta = 0.2$. Consider the activation function of the perceptron as the step function $f = \begin{cases} 1 & if\ \Sigma > 0 \\ 0 & otherwise \end{cases}$

(i). Can the perceptron learn a perfect solution for this data set?

(ii). Draw the perceptron graph and calculate the accuracy of the perceptron on the training data before training?

(iii). Using the perceptron *learning rule* and the learning rate of $\eta = 0.2$. Train the perceptron for **one epoch**. What are the weights after the training?

(iv). What is the accuracy of the perceptron on the training data after training for one epoch? Did the accuracy improve?