COMP3314_2C Machine Learning

Homework2:

1. Suppose you have developed a machine learning model to identify spam emails, and you have tested it on a dataset of 500 emails. The model classified 100 emails as spam, of which 80 were actually spam, and 20 were not spam. The model also classified 400 emails as not spam, of which 380 were actually not spam, and 20 were actually spam. Calculate the True Positive, False Positive, Precision, Recall, and F1 score of the model. (10 points)

Solution:

True Positive (TP) = 80 (number of emails that were actually spam and correctly classified as spam by the model)

False Positive (FP) = 20 (number of emails that were actually not spam but incorrectly classified as spam by the model)

False Negative (FN) = 20 (number of emails that were actually spam but incorrectly classified as not spam by the model)

True Negative (TN) = 380 (number of emails that were actually not spam and correctly classified as not spam by the model)

Precision = TP / (TP + FP) = 80 / (80 + 20) = 0.8

Recall = TP / (TP + FN) = 80 / (80 + 20) = 0.8

F1 score = 2 * Precision * Recall / (Precision + Recall) = 2 * 0.8 * 0.8 / (0.8 + 0.8) = 0.8

Therefore, the True Positive is 80, False Positive is 20, Precision is 0.8, Recall is 0.8, and F1 score is 0.8.

2. In k-means clustering, the input contains n sample points $X_1, X_2, \dots, X_n \in \mathbb{R}^d$ and integer k. We would like to assign each sample point X_j a label $y_j \in \{1, 2, \dots, k\}$, interpreted as "$X_j$ is assigned to cluster y_j." The means of the clusters are written $\mu_1, \mu_2, \dots, \mu_k$. (20 points)

   (a) What cost function does k-means clustering use? Note: Please provide a formula that utilizes the means rather than the formula for within-cluster variation. Feel free to use any notation to sum the terms in the cost function and provide clear explanations for the notation. (10 points)

Solution:

$$L = \frac{1}{2}\sum_{j=1}^{n} ||X_j - \mu_{y_j}||^2$$

(b) Consider the step of the algorithm where the labels y_j are held fixed while the cluster means μ_i are updated. You are asked to show that the following claim is correct: If we want to minimize the cost function, with appropriate calculus, we can choose each μ_i to be the mean (centroid) of the sample points assigned to cluster i. Make sure you explain your notation for counting the points in a cluster. Show your work, and don't skip any steps of the derivation. (15 points)

Solution:

We have

$$\frac{\partial L}{\partial u_i} = \frac{1}{2}\frac{\partial}{\partial \mu_i}\sum_{y_j=i} ||X_j - \mu_i||^2 = \sum_{y_j=i}(\mu_i - X_j) = 0$$

Therefore,

$$\mu_i = \frac{1}{n_i}\sum_{y_j=i} X_j$$

n_i denotes the number of sample points assigned to cluster i.

3. Given a maxpooling function $y = f_{maxpool}(x)$, perform one step maxpooling with pooling kernel size set to 3. (20 points)

(1) calculate the forward pass result y. (5 points)

(2) assume L is the final loss function and $\frac{\partial L}{\partial y}$ is given, calculate $\frac{\partial L}{\partial x}$ using the chain rule. (15 points)

Note: (i). no padding is needed and the pooling stride is set to 1; (ii). x is a 5x5 matrix, y is a 3x3 matrix, $\frac{\partial L}{\partial y}$ is a 3x3 matrix and $\frac{\partial L}{\partial x}$ is a 5x5 matrix. (iii) when multiple max values exist in the kernel, select the one with the smallest index.

x

6	2	5	4	4
9	1	5	3	7
7	2	4	3	8
4	5	5	1	0
0	2	8	0	8

$\frac{\partial L}{\partial y}$

0.1111	-0.0007	0
0	0.1104	0.1111
0.1111	0.1111	0.1111

example kernel index

1	2	3
4	5	6
7	8	9

Solution:

y

$\frac{\partial L}{\partial x}$

9	5	8
9	5	8
8	8	8

0	0	-0.0007	0	0
0.1111	0	0.1104	0	0
0	0	0	0	0.2221
0	0	0	0	0
0	0	0.2221	0	0

4. Suppose we have a multilayer perceptron (MLP) model with 3 inputs, 2 outputs, and a hidden layer with 4 neurons. The goal is to compute the partial derivative of the loss function with respect to a particular element $w_{1,2}^{(2)}$ of the second layer weight matrix $w^{(2)}$ for a given input sample. $w_{1,2}^{(2)}$ connects the first neuron in the hidden layer and the second neuron in the output layer. Assume that we use the squared loss function $L = \frac{1}{2}||y - \hat{y}||^2$, where y is the actual output and $\hat{y}$ is the predicted output of the model given an input x. $ReLU(x) = max(0, x)$ is applied to the hidden layer as the activation function and the output layer uses linear activation. (20 points)

Solution:

(1) Define notation:

- $x = (x_1, x_2, x_3)$: inputs
- $y = (y_1, y_2)$: actural outputs
- $\hat{y} = (\hat{y}_1, \hat{y}_2)$: predicted outputs
- $h = (h_1, h_2, h_3, h_4)$: outputs of hidden layer neurons

(2) Given the squared loss function $L = \frac{1}{2}||y - \hat{y}||^2$, we first compute the gradient with respect to the output layer neuron outputs: $\frac{\partial L}{\partial \hat{y}_1} = -(y_1 - \hat{y}_1), \frac{\partial L}{\partial \hat{y}_2} = -(y_2 - \hat{y}_2)$

Now we want to find the gradient with respect to $w_{1,2}^{(2)}$.

Since $\hat{y}_2 = w_{1,2}^{(2)} h_1 + w_{2,2}^{(2)} h_2 + w_{3,2}^{(2)} h_3 + w_{4,2}^{(2)} h_4$, we have $\frac{\partial \hat{y}_2}{\partial w_{1,2}^{(2)}} = h_1$

Now we can apply the chain rule to find the gradient of the loss function with respect to $w_{1,2}^{(2)}$:

$$\frac{\partial L}{\partial w_{1,2}^{(2)}} = \frac{\partial L}{\partial \hat{y}_2} \cdot \frac{\partial \hat{y}_2}{\partial w_{1,2}^{(2)}} = -(y_2 - \hat{y}_2) \cdot h_1$$

5. Suppose you have a dataset with 3 data points, each with 2 features:

$$\begin{pmatrix} 1 & 3 \\ 2 & 4 \\ 3 & 5 \end{pmatrix}$$

(a) Compute the covariance matrix of the dataset. (5 points)

(b) Perform an eigendecomposition of the covariance matrix and determine the eigenvectors and eigenvalues. (5 points)

(c) Project the data onto the subspace spanned by the first two principal components. (10 points)

(d) Reconstruct the original data points using the projections from part (c) and the first two principal components. (10 points)

Solution:

(a) The covariance matrix of the dataset is given by:

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

(b) To perform an eigendecomposition of the covariance matrix, we first subtract the mean of each feature from each data point to center the data. Then, we compute the covariance matrix and find its eigenvectors and eigenvalues. The eigenvectors are given by:

$$\mathrm{v}_1 = \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \mathrm{v}_2 = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

with corresponding eigenvalues of 0 and 2.

(c) To project the data points onto the subspace spanned by the first two principal components, we will first center the data by subtracting the mean of each feature:

Centered data points:

$$\begin{pmatrix} 1-2 & 3-4 \\ 2-2 & 4-4 \\ 3-2 & 5-4 \end{pmatrix} = \begin{pmatrix} -1 & -1 \\ 0 & 0 \\ 1 & 1 \end{pmatrix}$$

So, the projections are:

$$\begin{pmatrix} -1 & -1 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 & -\sqrt{2} \\ 0 & 0 \\ 0 & \sqrt{2} \end{pmatrix}$$

(d) To reconstruct the original data points, we multiply the projections by the transpose of the Projection_matrix and then add back the mean:

$$\text{mean} = \begin{pmatrix} 2 & 4 \end{pmatrix}$$

$$\begin{pmatrix} 0 & -\sqrt{2} \\ 0 & 0 \\ 0 & \sqrt{2} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} -1 & -1 \\ 0 & 0 \\ 1 & 1 \end{pmatrix}$$

$$\mathrm{r}_1 = \begin{pmatrix} -1 & -1 \end{pmatrix} + \begin{pmatrix} 2 & 4 \end{pmatrix} = \begin{pmatrix} 1 & 3 \end{pmatrix}$$

$$\mathrm{r}_2 = \begin{pmatrix} 0 & 0 \end{pmatrix} + \begin{pmatrix} 2 & 4 \end{pmatrix} = \begin{pmatrix} 2 & 4 \end{pmatrix}$$

$$\mathrm{r}_3 = \begin{pmatrix} 1 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 4 \end{pmatrix} = \begin{pmatrix} 3 & 5 \end{pmatrix}$$

So, the reconstructed results are:

$$\begin{pmatrix} 1 & 3 \\ 2 & 4 \\ 3 & 5 \end{pmatrix}$$