2021s18852

1.)

a.)

Binary classification is a type of classification task in machine learning and statistics where the goal is to categorize data points into one of two distinct groups or classes. These two classes are typically labeled as 0 and 1, True and False, Yes and No, or any other pair of contrasting labels.

b.)

Logistic regression starts by applying a linear model to the features of the dataset. For a given data point x, it computes a weighted sum of the features plus a bias term:

$$z = w_1x_1 + w_2x_2 + \cdots + w_nx_n + b$$

where:

- $x_1, x_2, \ldots, x_n$ are the features of the dataset
- $w_1, w_2, \ldots, w_n$ are the weights (parameters) of the model
- b is the bias term
- z is the output of the linear model

The linear output zzz is then passed through the **sigmoid function** to map it to a probability between 0 and 1:

$$h(z) = \frac{1}{1 + e^{-z}}$$

- $h(z)$ is the sigmoid function
- If sigmoid (z) ≥ 0.5 the class of the value is positive, otherwise if sigmoid is < 0.5, the class of the variable is negative.

2.)

a.) $$J(b) = -\frac{1}{m}\sum_{i=1}^{m}\left[y_1 \log\left(h(z_1)\right) + (1 - y_1)\log\left(1 - h(z_1)\right)\right]$$

$$J(b) = Cost\ function$$

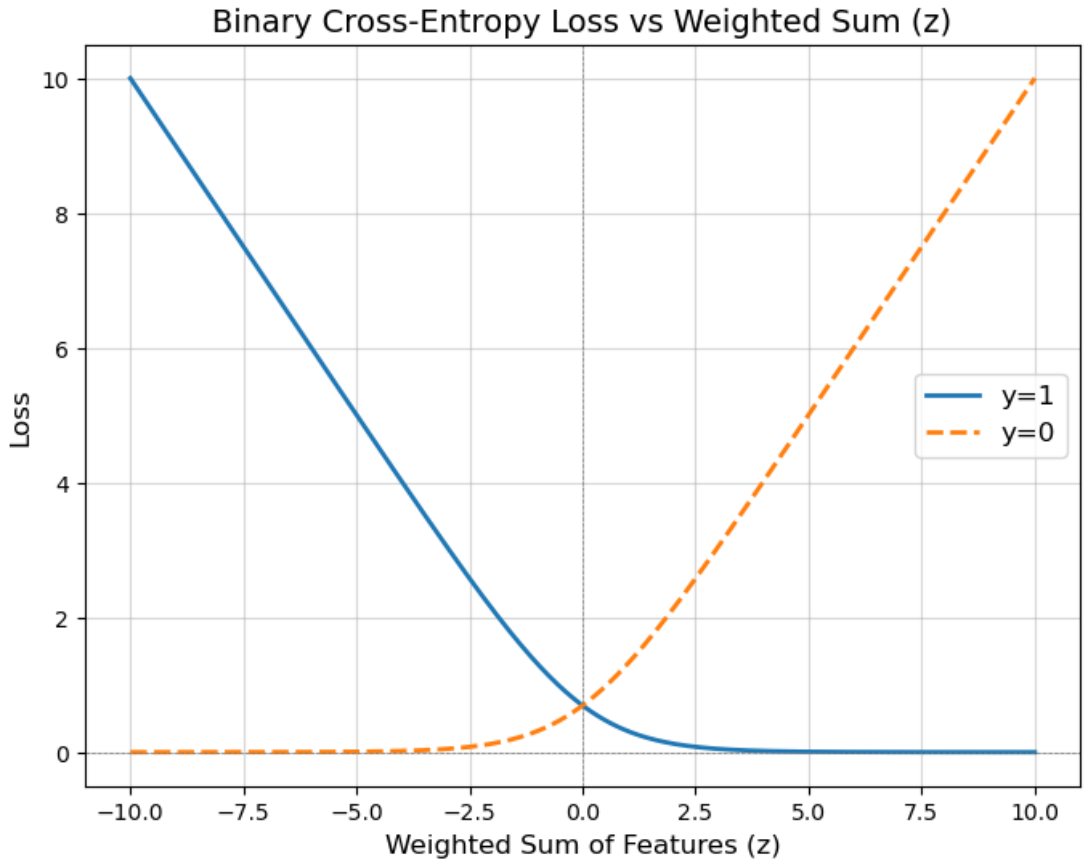
$$L(h) = -y_1 \log\left(h(z_1)\right) - (1 - y_1)\log\left(1 - h(z_1)\right) = Loss\ function\ where,$$

$$y_i = true\ class\ label(0\ or\ 1)\ of\ the\ i^{th}\ instance$$

$h(x_i) = predicted\ probability\ that\ the\ i^{th}\ instance\ belongs\ to\ class\ 1$
$m = Number\ of\ data\ instances$

b.) $L(h) = -y_1 \log\big(h(z_1)\big) - (1 - y_1) \log\big(1 - h(z_1)\big)$

c.)



d.)

The true class (y): y=1 (non-spam).
The predicted probability (y^): Closer to 0, meaning the model predicts the email as spam.

In the plot for y=1, as z (the weighted sum of features) becomes negative (indicating a lower predicted probability for y=1), the loss increases sharply.

The incorrect classification of the model of a non-spam email as spam corresponds to this scenario, where z<0 and the loss is high.

So, we can expect a **higher loss** in this scenario because the model prediction (y^≈0) strongly contradicts the true label (y=1).

3.)

Actual label	Predicted label	TP, TN, FP, FN
Car	Van	FN
Car	Car	TP
Van	Car	FP
Van	Van	TN

4.)

a. TP = 80
TN = 75
FP = 8
FN = 6

b. $Accuracy = \frac{TP+TN}{TP+FP+FN+TN} = \frac{155}{169} = 0.9172$

c. $Precision = \frac{TP}{TP+FP} = \frac{80}{88} = 0.9091$

d. $Precision\ recall = \frac{TP}{TP+FN} = \frac{80}{86} = 0.9302$

e. $F1 = 2 \times \frac{precision \times recall}{precision+recall} = 2 \times \frac{0.9091 \times 0.9302}{0.9091+0.9302} = 0.9195$

5.)

a.)

- Number of classes: 5 (White dwarf, Brown dwarf, Red dwarf, Blue Giant, Red Giant)
- Number of classifiers: Equal to the number of classes = 5

b.)

Number of classes: 5

Number of classifiers: $\binom{n}{2} = \frac{n(n-1)}{2}$

$$\binom{5}{2} = \frac{5(5-1)}{2} = 10$$

Number of classifiers: 10

c.)

The one-vs-rest (OVR) method works as follows:

1. Create a separate binary classifier for each class in the dataset.
2. For each classifier, treat the target class as the "positive" class and combine all other classes into a single "negative" class.
3. During prediction, each binary classifier outputs a probability that the instance belongs to its respective class.
4. The final class is predicted as the one with the highest probability among all classifiers.

For the dataset:

Model	Positive Class	Negative class
1	White dwarf	[Brown dwarf, Red dwarf, Blue Giant, Red Giant]
2	Brown dwarf	[White dwarf, Red dwarf, Blue Giant, Red Giant]
3	Red dwarf	[White dwarf, Brown dwarf, Blue Giant, Red Giant]
4	Blue Giant	[White dwarf, Brown dwarf, Red dwarf, Red Giant]
5	Red Giant	[White dwarf, Brown dwarf, Red dwarf, Blue Giant]

d.)

1. Create a binary classifier for every pair of classes in the dataset.
2. Each binary classifier is trained to distinguish between two specific classes, ignoring all other classes.
3. During prediction, each binary classifier votes for one of its two classes.
4. The final class is determined by a majority voting scheme, where the class receiving the most votes across all classifiers is selected.

For the dataset:

Model	Positive Class	Negative class
1	White dwarf	Brown dwarf
2	White dwarf	Red dwarf
3	White dwarf	Blue Giant
4	White dwarf	Red Giant

5	Brown dwarf	Red dwarf
6	Brown dwarf	Blue Giant
7	Brown dwarf	Red Giant
8	Red dwarf	Blue Giant
9	Red dwarf	Red Giant
10	Blue Giant	Red Giant

6.)

Underfitting is observed at very small (C = 0.001) and very large (C = 1000) values of C. These settings constrain the model too much or too little.

The best balance is observed at C = 1, where the training and testing costs are closely aligned, indicating the model generalizes well.

A systematic evaluation of both training and testing costs shows an appropriate selection of C=1. The results are excellent and demonstrate a well-tuned model for the Iris dataset's binary classification task. However, further evaluation on larger or more challenging datasets is recommended to validate the robustness of the approach.