UNIVERSITY OF
SOUTH DAKOTA

Machine Learning Fundamentals (CSC 722)

Question/Answer

Name: Al-Amin Hossain
ID: 101131177

Ques: e1. How can I measure the performance of my model?

Ans: To assess the performance and the effectiveness of my logistic regression model, I can use evaluation metrics. As it is a classification model I will use Accuracy score, Precision, Recall, F1-Score, ROC and AUC, Log Loss, and Confusion Matrix to evaluate the model.

If my model was a Regression model I would have used Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R2) Score.

Ques: e2: What are: Accuracy, Confusion Matrix, Precision, Recall, F1 Score, ROC & AUC, Log Loss?

Accuracy: The accuracy score measures the proportion of correctly predicted instances out of the total instances in a classification task. For example, if a model correctly predicts 90 out of 100 instances, the accuracy score is 90%.

Example:
Total instances: 100
Correctly predicted instances: 90
Accuracy Score: 90%
It's a simple and intuitive metric to assess the overall performance of a classification model.

Confusion Matrix: A confusion matrix is a tabular representation that summarizes the performance of a classification model. It displays counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

Example:
Consider a binary classification task to distinguish between cats and dogs.
True Positives (TP): The model correctly predicts cat images as cats.
True Negatives (TN): The model correctly predicts dog images as dogs.
False Positives (FP): The model incorrectly predicts dog images as cats.
False Negatives (FN): The model incorrectly predicts cat images as dogs.

```
              Predicted Class
               |  Cat  |  Dog  |
Actual Class   |-------|-------|
        Cat    |   TP  |   FN  |
        Dog    |   FP  |   TN  |
```

The diagonal elements represent correct predictions, while off-diagonal elements represent errors.

Precision: Precision is a metric used in binary classification to measure the accuracy of positive predictions made by a model. It calculates the proportion of true positive predictions (correctly identified positives) out of all positive predictions made by the model.

Example:
Let's say we have a spam email classifier. Precision would measure the ratio of correctly identified spam emails (true positives) out of all emails predicted as spam by the model (true positives + false positives).

If the model correctly identifies 80 spam emails out of 100 emails predicted as spam and incorrectly identifies 20 non-spam emails as spam, the precision would be:

Precision = True Positives / (True Positives + False Positives)
Precision = 80 / (80 + 20)
Precision = 80%

So, the precision of the model in this example would be 80%. This indicates that out of all emails predicted as spam by the model, 80% are actually spam.

Recall: Recall, also known as sensitivity, is a metric used in binary classification to measure the ability of a model to correctly identify all positive instances in the dataset. It calculates the proportion of true positive predictions (correctly identified positives) out of all actual positive instances.

Example:
Consider a medical test for detecting a particular disease. Recall would measure the ratio of correctly identified diseased individuals (true positives) out of all individuals who actually have the disease (true positives + false negatives).

If the medical test correctly identifies 90 out of 100 individuals who have the disease and misses 10 individuals who have the disease, the recall would be:

Recall = True Positives / (True Positives + False Negatives)
Recall = 90 / (90 + 10)
Recall = 90%

So, the recall of the medical test in this example would be 90%. This indicates that out of all individuals who actually have the disease, the test correctly identifies 90% of them.

F1-Score:
The F1 score is a metric used in binary classification that combines precision and recall into a single value. It provides a balance between precision and recall by taking their harmonic mean.

Example:

Suppose we have a binary classification task to identify spam emails. We have a model that predicts spam and non-spam emails.

Let's say the precision of the model (correctly identified spam emails out of all emails predicted as spam) is 80% and the recall (correctly identified spam emails out of all actual spam emails) is 85%.

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)
F1 Score = 2 * (0.80 * 0.85) / (0.80 + 0.85)
F1 Score ≈ 0.82

So, the F1 score of the model in this example is approximately 0.82. It indicates a balance between precision and recall, where higher values signify better performance in both metrics.

ROC & AUC: The Receiver Operating Characteristic (ROC) curve is a graphical representation of the trade-off between the true positive rate (sensitivity) and the false positive rate (1 - specificity) for different threshold values. It helps visualize the performance of a binary classification model across various discrimination thresholds.

Example:
Consider a medical test to detect a disease. The ROC curve would plot the true positive rate (sensitivity) against the false positive rate (1 - specificity) at different threshold settings for the test. Each point on the curve represents the performance of the test at a particular threshold level.

The Area Under the ROC Curve (AUC) quantifies the overall performance of a binary classification model across all possible threshold settings. A higher AUC value indicates better discrimination ability of the model.

<u>Example:</u>
Suppose we have two medical tests (Test A and Test B) for detecting a disease. Test A has an AUC of 0.85, while Test B has an AUC of 0.75. This means that Test A has better discrimination ability compared to Test B in distinguishing between individuals with and without the disease.

Log Loss: Log Loss, also known as logarithmic loss or cross-entropy loss, is a metric used to evaluate the performance of a classification model that outputs probabilities. It measures the accuracy of the probabilities predicted by the model by penalizing confident but wrong predictions.

<u>Example:</u>
Let's consider a binary classification problem where we predict whether an email is spam or not spam. The true label for an email is either 0 (not spam) or 1 (spam).

Suppose our model predicts the following probabilities for three emails:

Email 1: Actual label 0, Predicted probability of being not spam = 0.9
Email 2: Actual label 1, Predicted probability of being spam = 0.2
Email 3: Actual label 1, Predicted probability of being spam = 0.7
Using the formula for log loss:

Log Loss = - (1/N) * Σ [y * log(p) + (1 - y) * log(1 - p)]

Where:

N is the number of samples
y is the actual label (0 or 1)
p is the predicted probability

Substituting the values:

Log Loss = - (1/3) * [(0 * log(0.9) + (1 - 0) * log(1 - 0.9)) + (1 * log(0.2) + (1 - 1) * log(1 - 0.2)) + (1 * log(0.7) + (1 - 1) * log(1 - 0.7))]

Calculating each term:

Log Loss = - (1/3) * [(0 * -0.105 + 1 * -2.303) + (1 * -1.609 + 0 * -0.223) + (1 * -0.357 + 0 * -1.204)]

Log Loss ≈ - (1/3) * [2.408 + 1.609 + 0.357]

Log Loss ≈ - (1/3) * 4.374

Log Loss ≈ -1.458

So, the log loss for this example is approximately 1.458. Lower log loss values indicate better performance, with 0 representing perfect predictions.