DataLab Preparation (Week 2, DataLab II, Thursday)

**2. Introduction to Model Evaluation**

**2a How accuracy is calculated? Why we cannot rely only on accuracy when evaluating a model?**

Accuracy is calculated as: # of correctly classified instances/ # of all instances.

We cannot rely only on accuracy when evaluating a model because it simplifies the situation too much.

**2b How precision and recall are calculated?**

Precision = TP (True Positives) / (TP + FP (False Positives))

Recall = TP/ (TP + FN(False Negatives))

**2c How the F1 score is calculated?**

F1 score = 2/(1/Precision + 1/Recall)

**2d What does a precision-recall curve represent?**

A precision-recall (PR) curve is a graphical representation that shows the trade-off between precision and recall for different thresholds used in a binary classification model.

**2e How Mean Absolute Error (MAE) is calculated?**

MAE = Sum of the absolute values of all errors

**2f Can we use MAE to evaluate a classification model? Why?**

Mean Absolute Error (MAE) is used to evaluate regression models, not classification models. MAE is designed for regression tasks, measuring the average absolute difference between continuous predictions and actual values, making it unsuitable for evaluating classification models, which predict discrete class labels.

**2g How Mean Squared Error (MSE) is calculated?**

MSE = The mean of the sum of the squared errors

**2h How Root Mean Squared Error (RMSE) is calculated?**

RMSE = Root of MSE

**2i What is the different between MAE, MSE, and RMSE?**

MAE is the average of absolute errors, MSE is the average of squared errors, and RMSE is the square root of MSE. They all provide measures of model performance in terms of prediction accuracy, but they differ in their sensitivity to outliers and in the scale of the error metric.

**4. Improving the Performance of Your Model**

**4a What is the expected behavior of a loss curve?**

The expected behaviour of a loss curve is an initial decrease in loss as the model learns from the training data, followed by convergence or stabilisation, and potential overfitting indicated by increasing loss on the validation set.

**4b What is the expected behavior of a accuracy curve?**

The expected behaviour of an accuracy curve during the training of a machine learning model typically involves an initial increase in accuracy as the model learns from the training data, followed by stabilisation or fluctuations, and potential overfitting indicated by a decrease in accuracy on the validation set.

**4c How can we identify overfitting in a loss curve?**

Overfitting in a loss curve can be identified by observing a significant decrease in loss on the training data while loss on the validation data begins to increase or stagnate, which would indicate that the model is fitting too closely to the training data and may not generalise well to new data.

**4d How can we set the initial weights of a model in Keras?**

We can set the initial weights of a model by specifying the 'kernel\_initializer' parameter when defining each layer.

**4e Explain in your own words how an activation function works.**

An activation function takes as input: the sum of the inputs from a neuron multiplied by each connection, plus bias weight. Then it decides what the neuron does and what output it produces.

**4f How to select which activation function to use?**

We can choose an activation function based on their properties, the problem at hand, and the layer we are looking at in our network, one activation function. As a starting point, the best activation function is ReLU as they train fast and will tend to generalise well to most problems.

**4g What is the effect of the parameter batch\_size in the fit function in Keras?**

It controls how many samples are processed before the model's internal parameters (weights) are updated based on the computed gradients.

**4h What is batch normalization?**

Batch normalization normalizes the activations of each layer by adjusting and scaling them so that they have a mean of zero and a standard deviation of one, which is then applied to each mini-batch during training.

**4i What is hyperparameter tuning?**

Hyperparameter tuning is basically the optimization of parameters set before model training. The aim is to find the best hyperparameter combination for optimal model performance on a validation set.