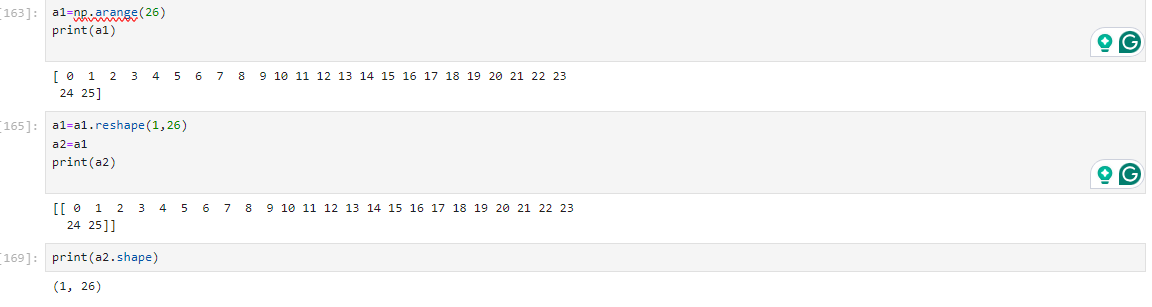
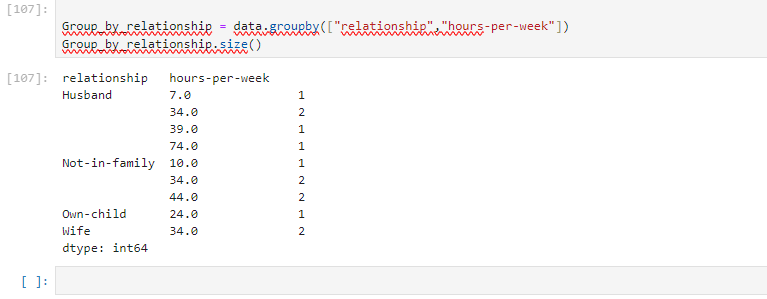
LAB Logbook

Lab 1

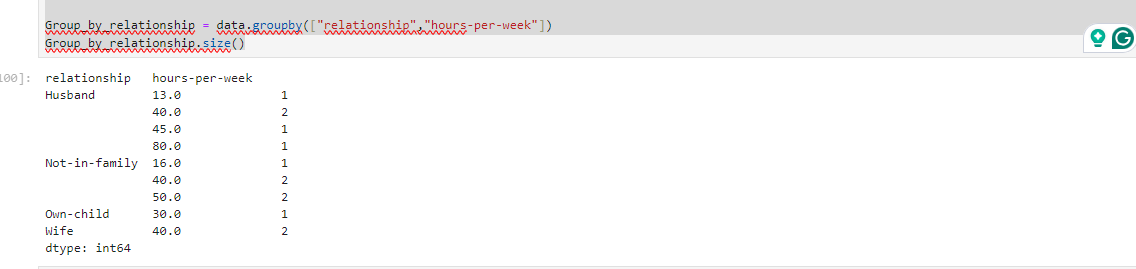


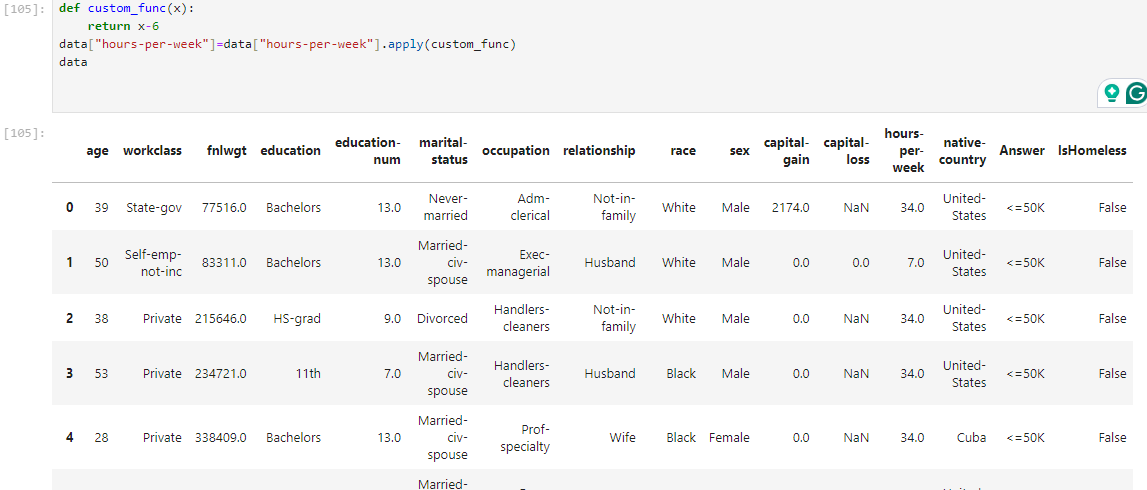


Summary:

This task involves transforming a matrix (or array) called an into a one-row 2D array. In other words, the array must be restructured to maintain its two sets of brackets, indicating that it is two-dimensional. Ultimately, the modified array and its shape attribute are displayed after transforming the array into a variable. This process demonstrates how to reshape an array and utilize the shape attribute to verify its dimensions. The second part of the process includes two actions: (1) employing np.arange to generate a numerical vector with at least 100 elements if the total is under 10; and (2) transforming a matrix or array into a numerical vector, where the number of elements corresponds to the last two digits of a student ID (SID).

Lab 2





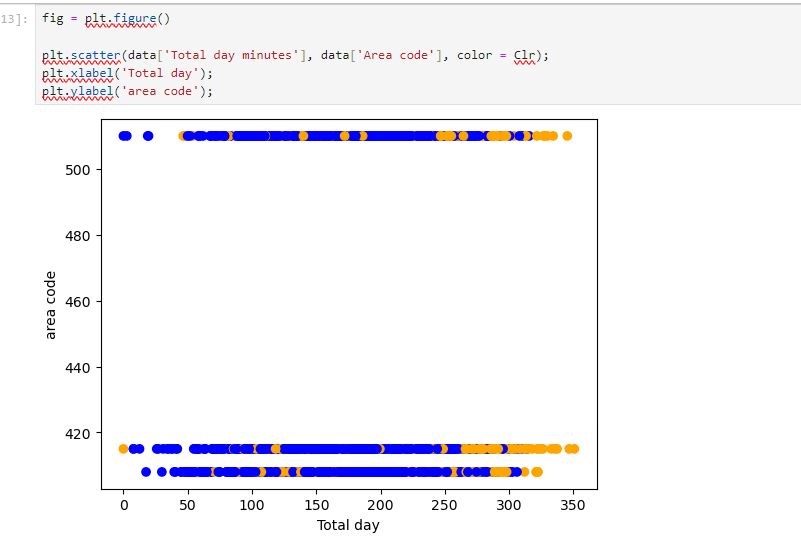


Summary:

The initial phase of analyzing the dataset involves determining a number n using the last digit of a student ID. The initial columns to be utilized for grouping the

dataset are "relationship" and "hours-per-week." Next, the data undergoes transformation by deducting n from the "hours-per-week" figures in the initial DataFrame. To facilitate additional examination of the altered data, the revised information is finally organized by "relationship" and the lowered "hours-per-week" figures.

Lab 3



Summary:

Generate a bicolor features interaction plot using two columns chosen according to the last two digits of your SID. Specifically, one column from the list below related to the last digit, and one column related to the second-to-last digit. Employ the subsequent number in order for the second column, if the final two numbers are the same. Utilize columns 7 and 8 if the SID concludes with 77, and columns 9 and 0 if it concludes with 99 for Total day calls and Total day charge, respectively. Employing a two-color scheme, the illustration should visually represent the relationship or interaction between these two columns.

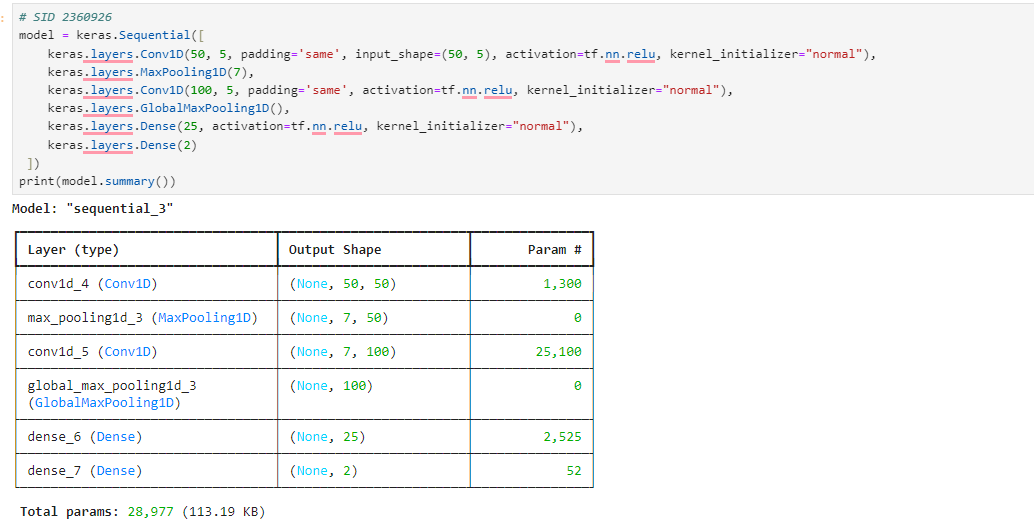
Lab 4



Summary:

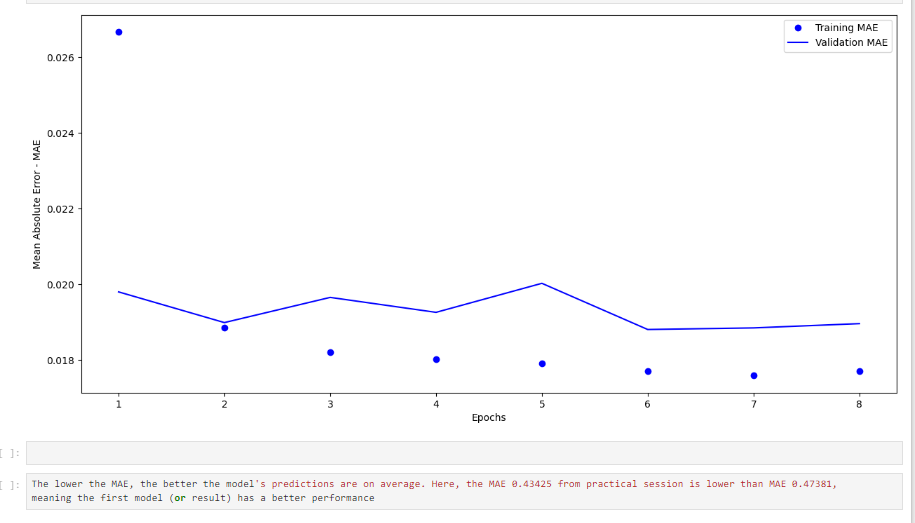
Our task is to create a Multi-Layer Perceptron (MLP) featuring two hidden layers. The quantity of neurons (cells) in the initial hidden layer should be determined by the last three digits of your SID, whereas the second hidden layer should contain about half that number of neurons (rounded as needed). For instance, it contains 167 neurons in the initial hidden layer and 84 neurons in the subsequent layer if the SID ends with 167. In this task, you will similarly train the MLP for ten epochs using the same dataset and parameters from a prior practical session. To disclose the performance disparity, you must compute and present the Mean Absolute Error (MAE) of your model after training and compare it with the MAE obtained during the practical.

Lab 5









Summary:

For the task, alterations need to be applied to a CNN model from the hands-on session. Initially, configure the batch size to 50 and reduce the size of the convolutional kernel to 5. Next, calculate the total number of epochs using this formula: Z + YZ + YZ + Y if Z ≠ 0Z.eq 0Zʀ =0; 10+Y10 + Y10+Y if Z=0Z = 0Z=0 and Y≠0Y eq 0Yʀ�=0; or 101010 if Z=Y=0Z = Y = 0Z=Y=0. In this context, ZZZ and YYY represent the last two digits of your SID. Please rephrase the following text while keeping the word count the same and not altering any other parameters. Utilize the identical dataset to build and train the CNN model, and then report the Mean Absolute Error for the test.

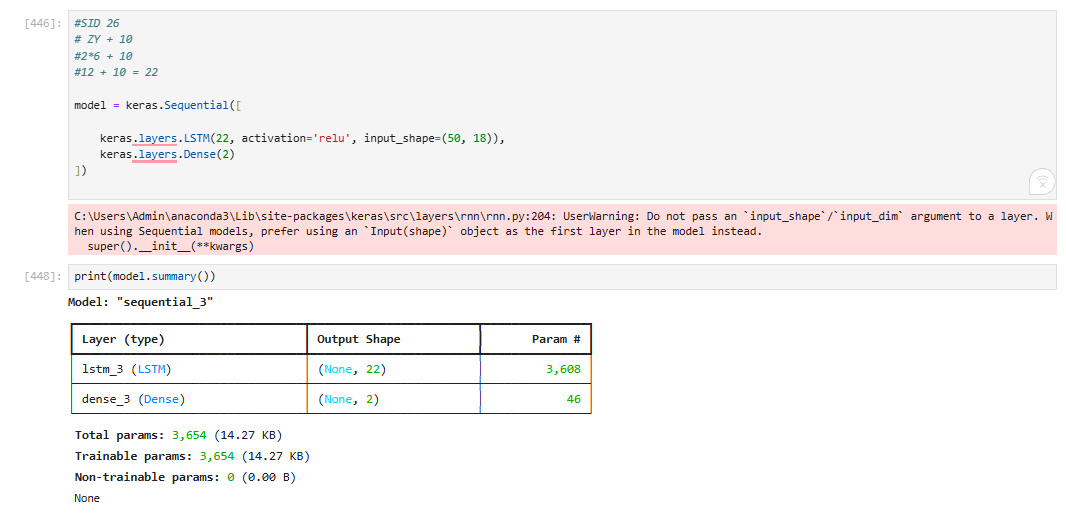
Lab 6

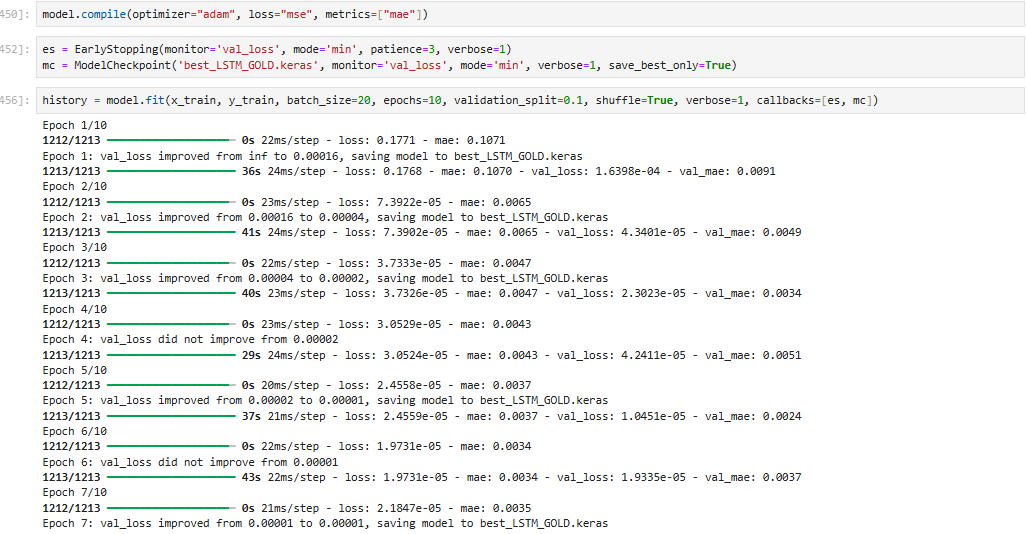


Summary:

It employs iplot() to create the price chart for 'High\_Bid' and 'Low\_Bid' within a dataset. The information begins with the final five digits of your SID number, which serves as the row index to slice the dataset. The duration for the chart (in minutes) is determined by the final three digits of your SID. This will partition the data as specified, after which iplot() can be utilized for an interactive chart depicting 'High\_Bid' and 'Low\_Bid' prices over time. This plot would enable the examination of the pricing trend within the designated range.

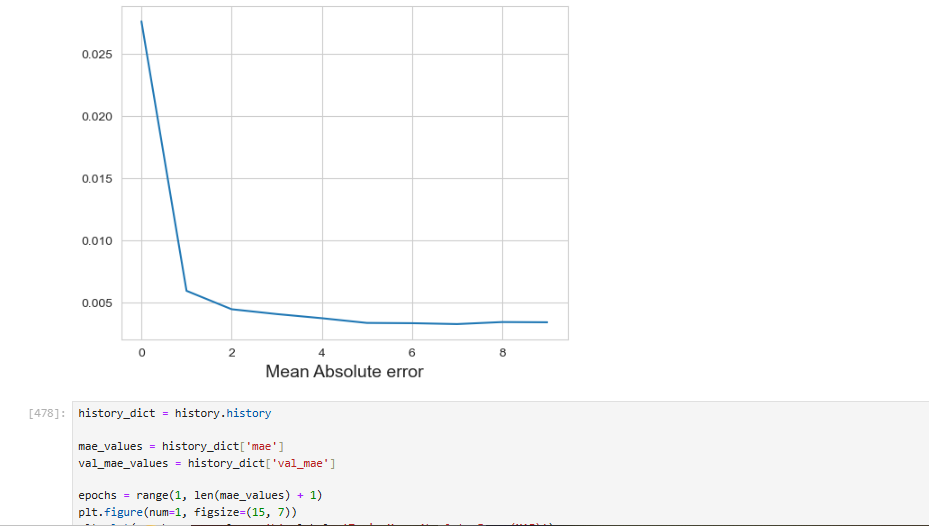
Lab 7

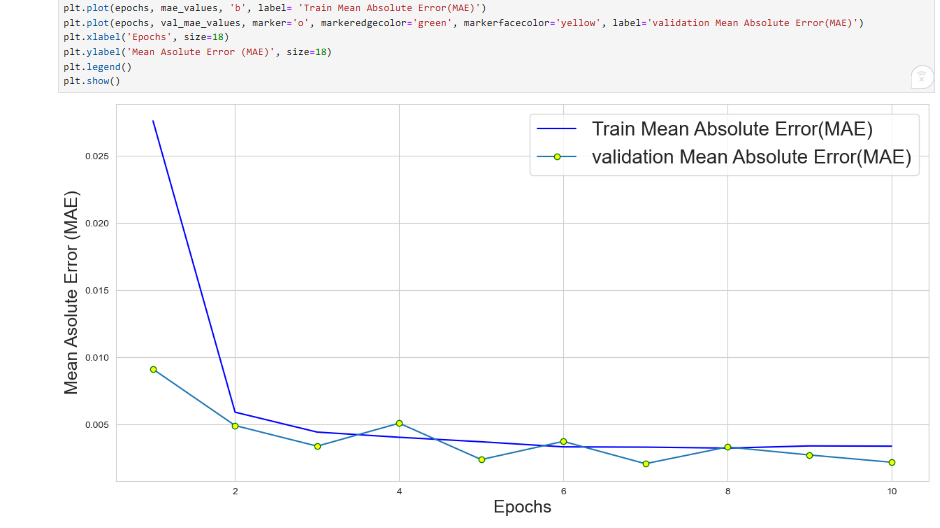








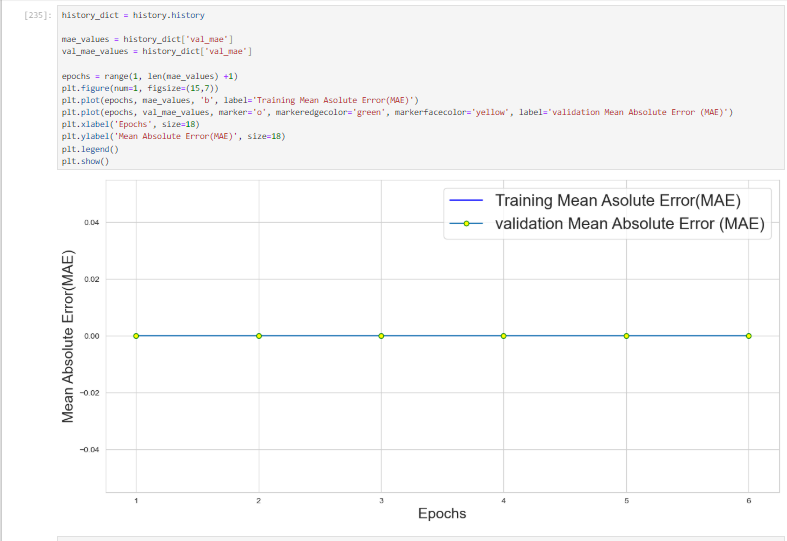


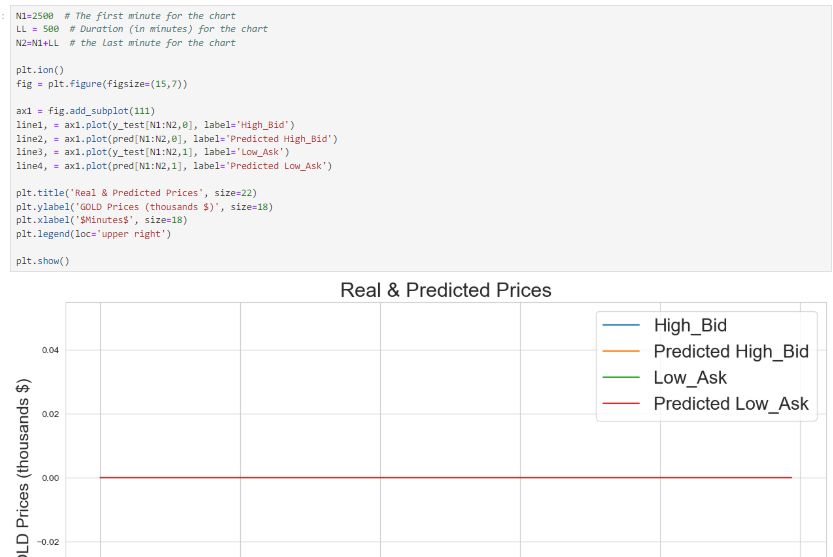


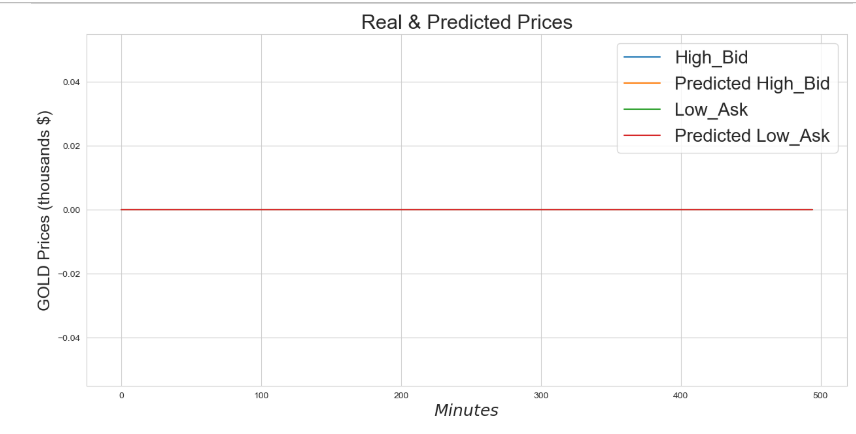
Summary:

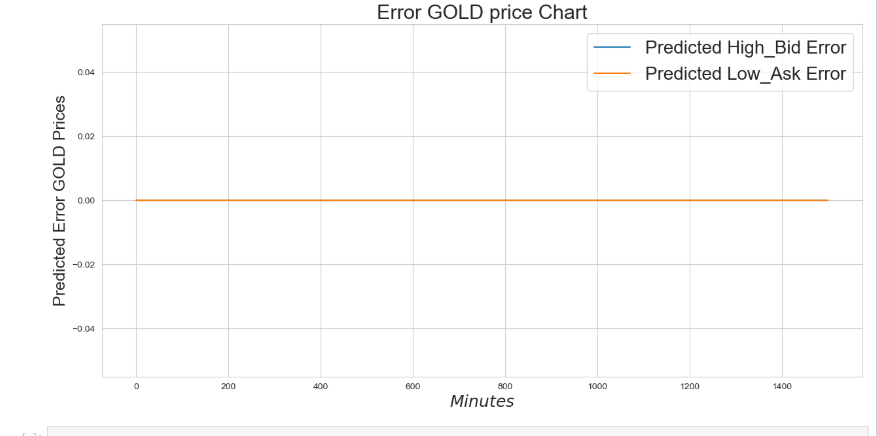
Adjust the LSTM model in the hands-on session for the assignment. Utilize the formula ZY+10ZY + 10ZY+10, where ZZZ and YYY represent the final two digits of your SID, to compute a crucial LSTM parameter (for instance, the number of units). For early termination, set the patience at 3 and the total epochs at 10. Maintain all other parameters as they were in the practical session, including the optimizer and learning rate. Employ the identical dataset to assemble and train the new LSTM model. Display the MAE and MSE of the test following the training. Ultimately assess any modifications or enhancements by contrasting the MSE and MAE of the test.

Lab 8









Summary:

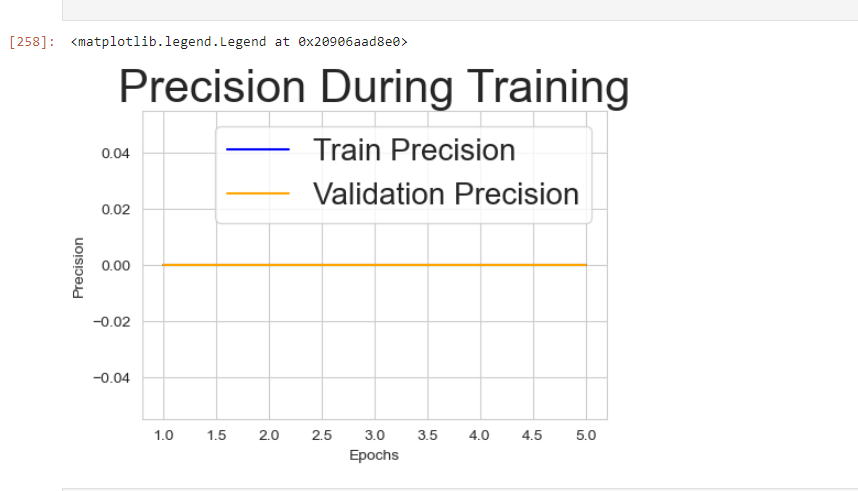
To forecast and evaluate the price of silver for the upcoming five minutes, a neural network model needs to be created and trained utilizing an MLP, CNN, or LSTM. The model must be trained on a historical dataset of silver prices including important details like price trends, timestamps, or additional predictors. Post-training, the model's performance needs to be assessed with suitable metrics like MAE or MSE. Subsequently, the trained model estimates the silver prices for the next 5-minute period, allowing for an assessment of the accuracy and dependability of the predictions.

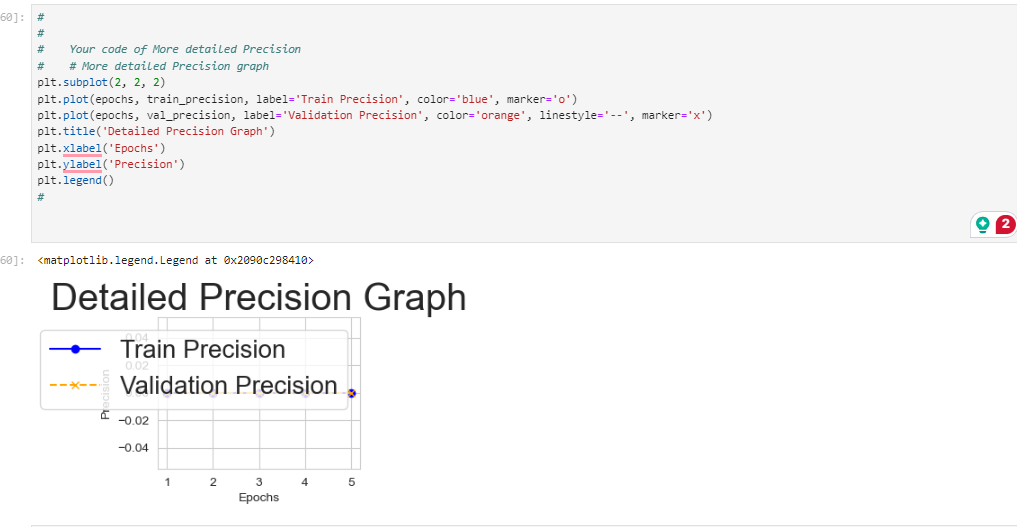
Lab 9

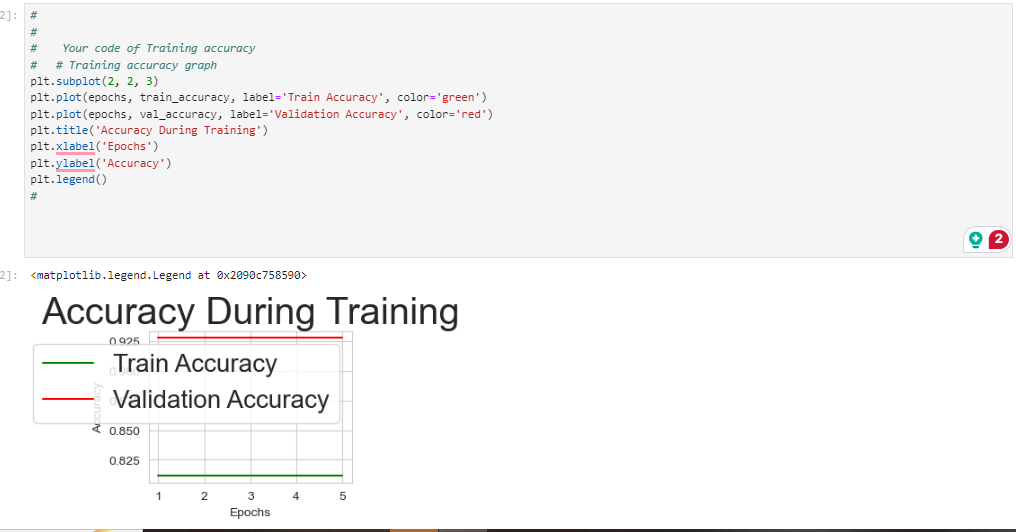
Quiz Test

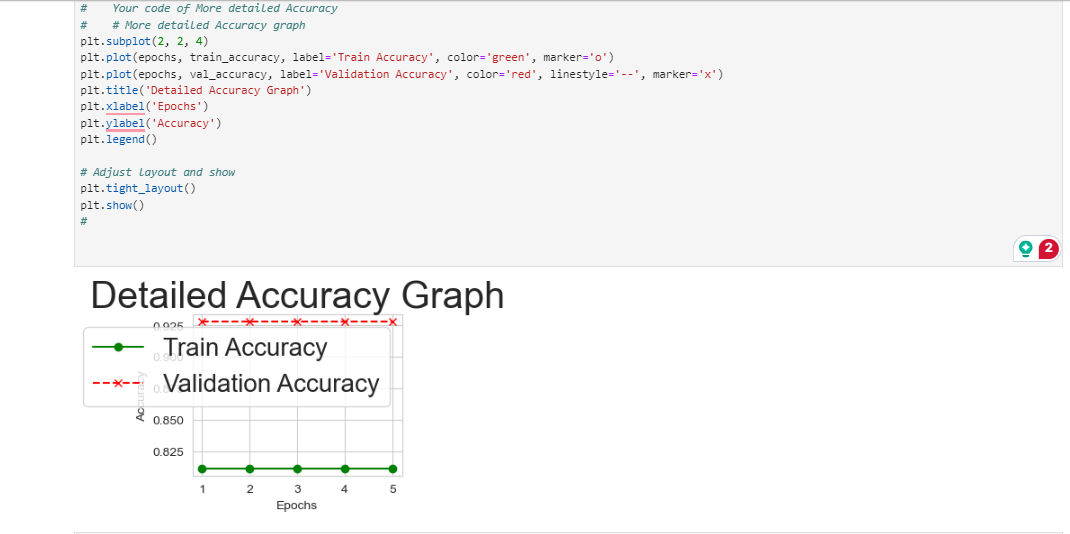
Lab 10







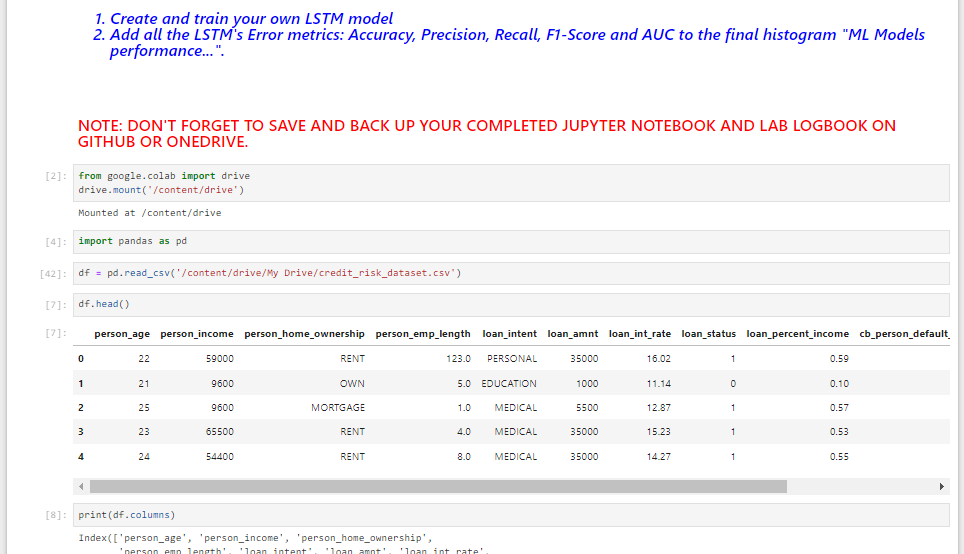


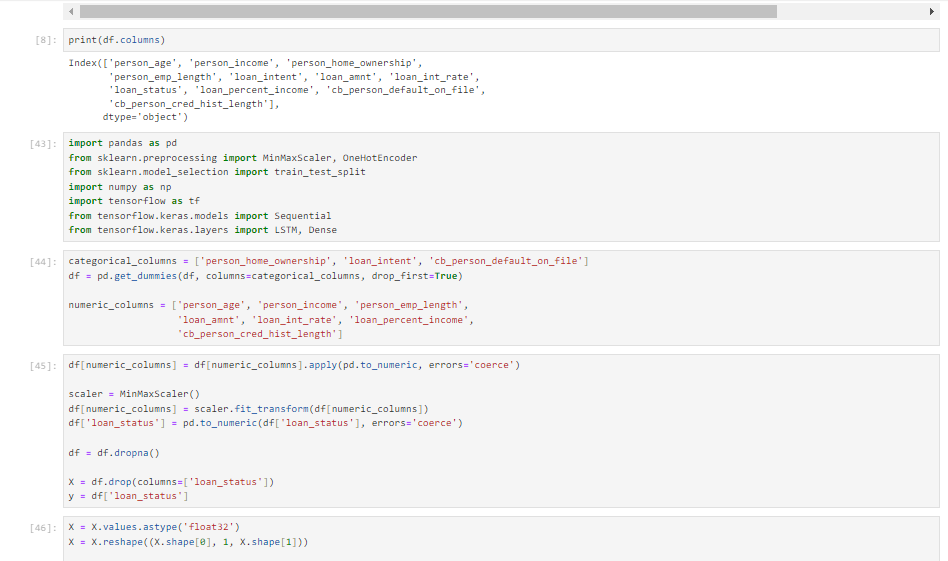


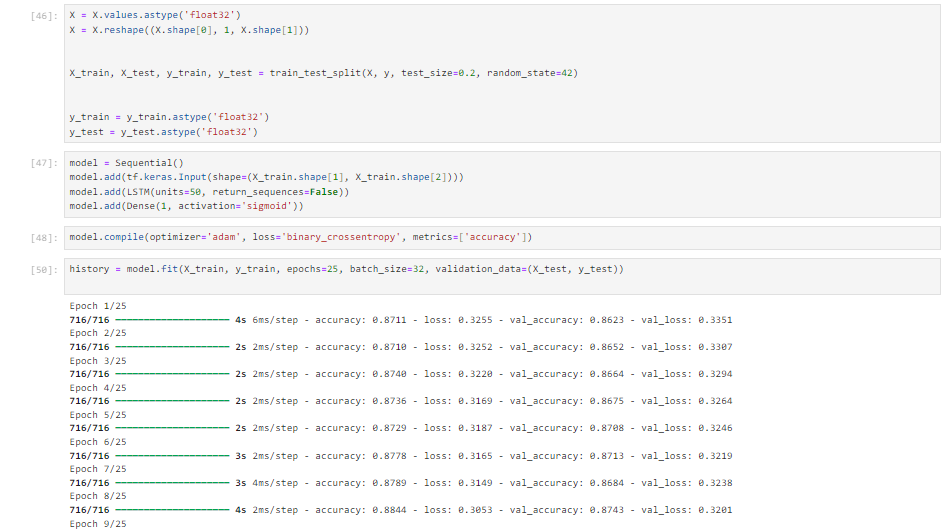
Summary:

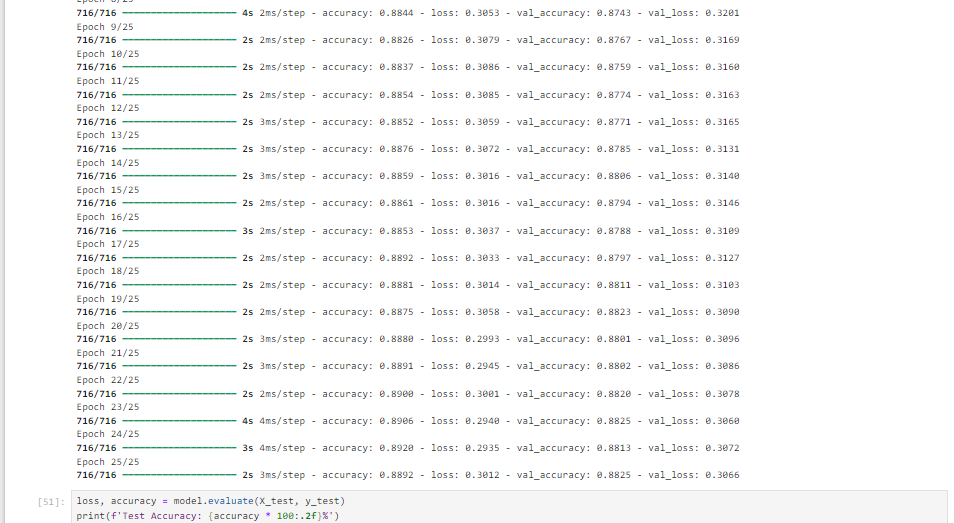
The challenge is to represent the effectiveness of a machine learning model that is undergoing training, with an emphasis on precision and accuracy. To monitor the model's performance in predicting positive cases during training, four graphs should be created: (1) a general graph depicting precision scores across epochs; (2) a more detailed precision graph for a closer look at fluctuations; (3) a training accuracy graph illustrating the ratio of accurately predicted cases over epochs; and (4) a more detailed accuracy graph to analyze subtle trends. These visual tools assist in recognizing the model's learning curve and determining where improvement should be concentrated.

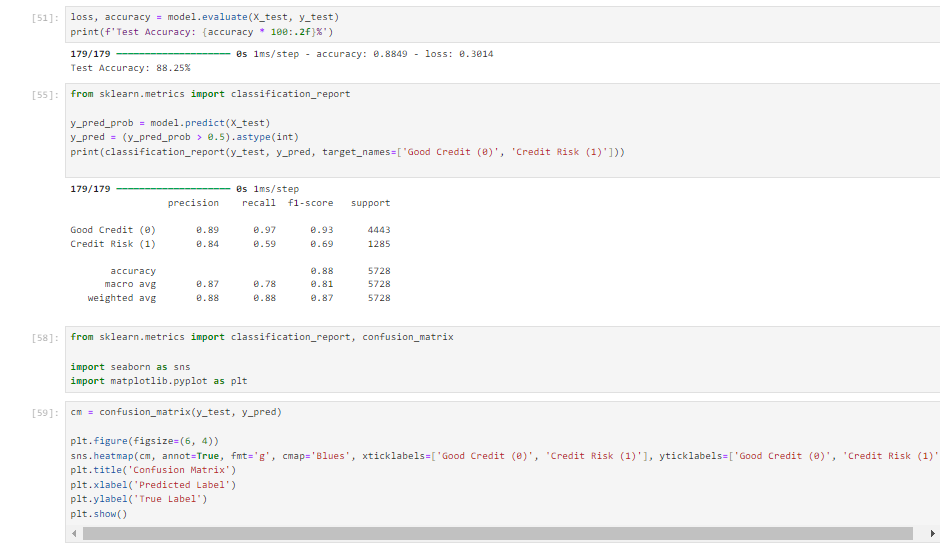
Lab 11

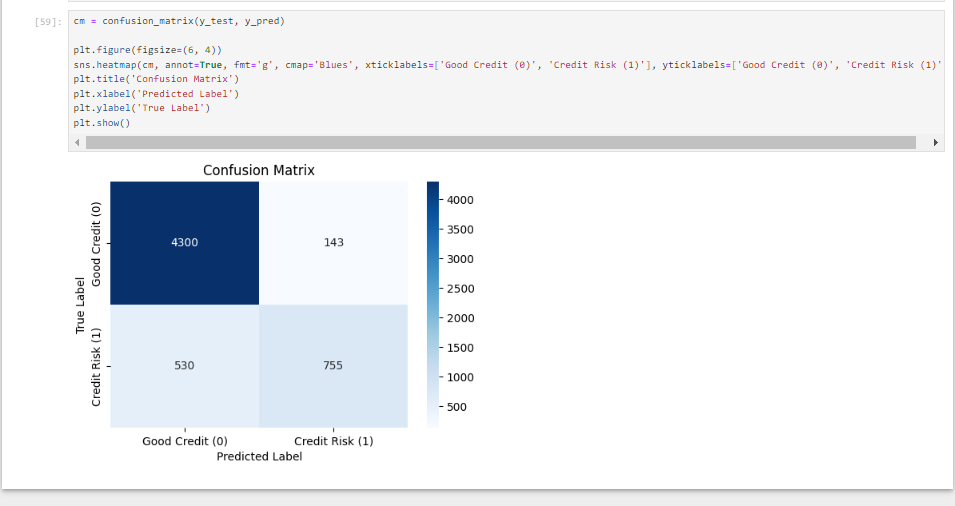












Summary:

This model employs an LSTM network for binary classification tasks like predicting credit risk, where the cases are categorized as Good Credit or Credit Risk. Following training, the effectiveness of the LSTM is assessed using important error metrics like Accuracy, Precision, Recall, F1-Score, and AUC, which means Area Under the Curve. Additionally, a visual confusion matrix displays the true positives, true negatives, false positives, and false negatives. For instance, in the matrix, there are 4300 accurate predictions for "Good Credit" and 755 accurate predictions for "Credit Risk," along with some misunderstandings. In the concluding histogram, these metrics will be employed.

Lab 12

N/A