*SECTION A* – Documentation

*Module*: CIS6005 Computational Intelligence

*School*: Cardiff School of Technologies/Varna University of Management

Lecturer: Osman Osman

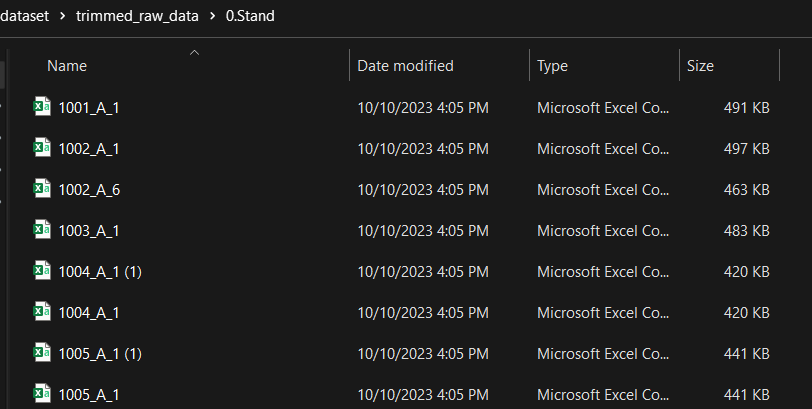
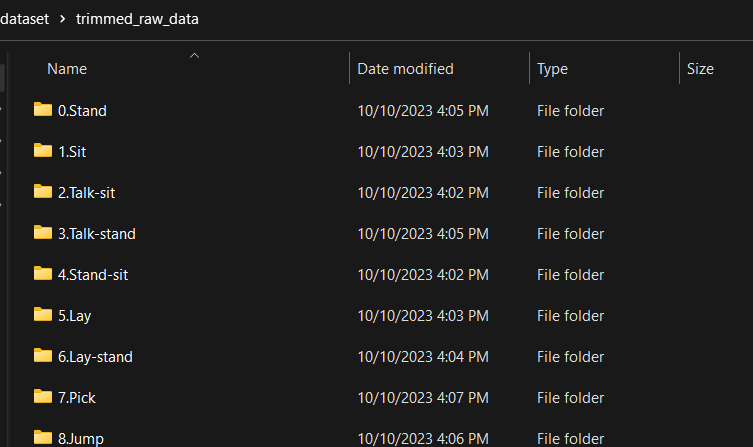
*Student number*: 20283946

*Github link:* [*https://github.com/lightonray/st20283946\_CIS6005*](https://github.com/lightonray/st20283946_CIS6005)

*Introduction to Dataset*

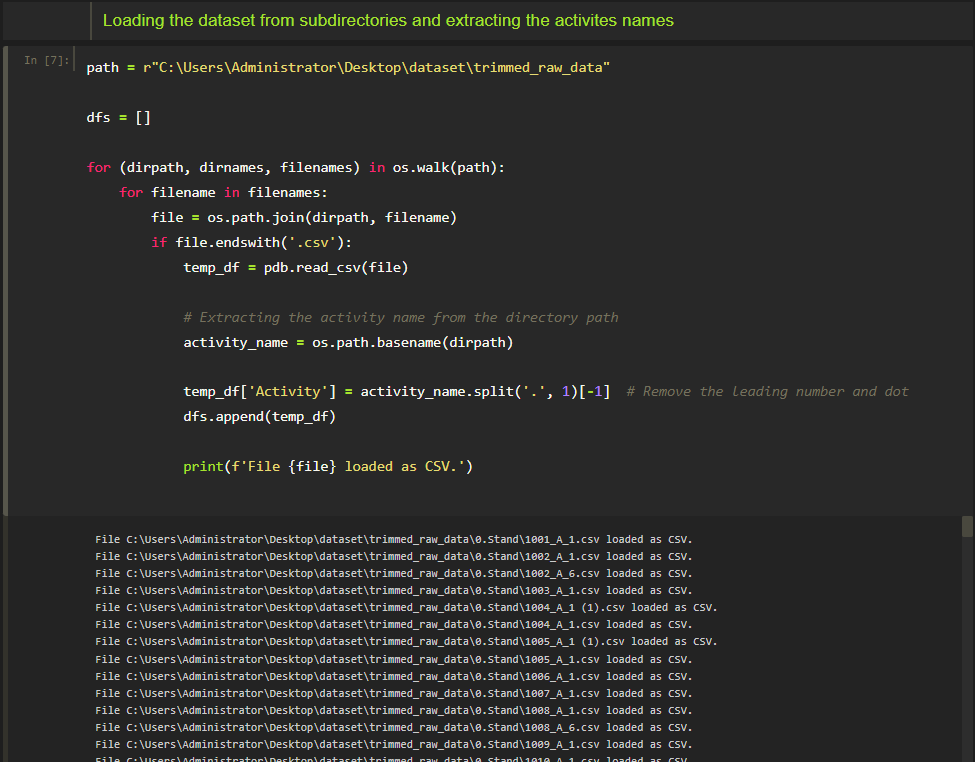
The dataset comprises sensor data collected for various physical activities like 'Stand', 'Sit', 'Talk-sit', 'Talk-stand', 'Stand-sit', 'Lay', 'Pick', 'Jump', 'Push-up', 'Sit-up', 'Walk', 'Walk-backwards' etc. These records were gathered from sensor-equipped devices capturing accelerometer and gyroscope readings. Dataset uses the X,Y,Z coordinates which denote height, width and dept.

*Data Preparation and Exploration*

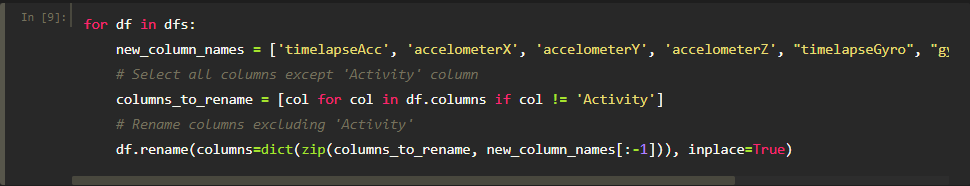
**

***Data Transformation***

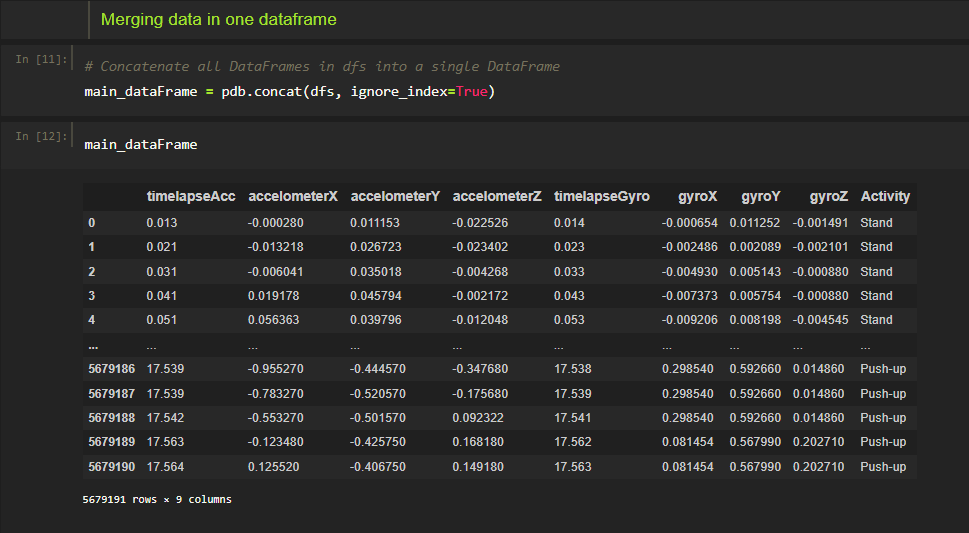
*The dataset concatenation* from multiple CSV files from each activity which located in different sub-directories, activity name extracted from each specific folder which represented an activity.

******

***Setting column names***

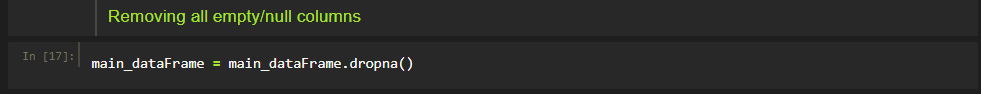
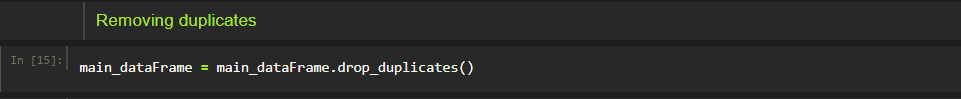


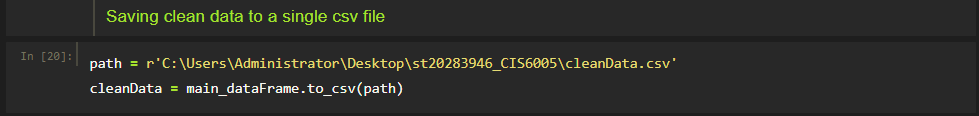
***Merging of the dataset from the list into one Data frame***



*Data Transformation*

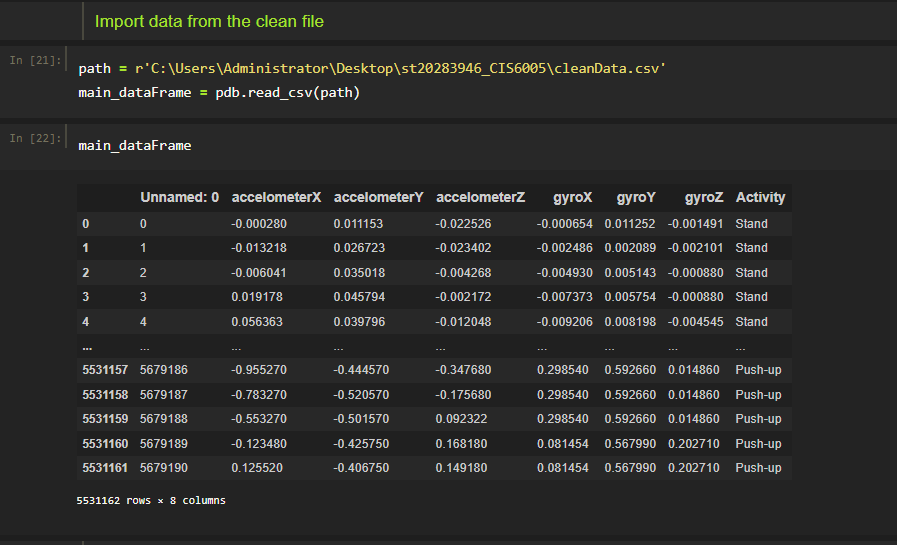
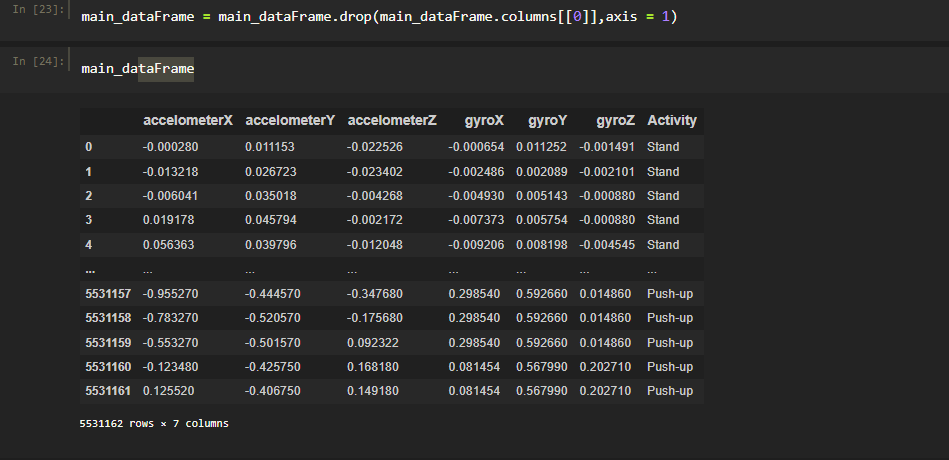
The dataset underwent thorough cleaning procedures, including the removal of duplicate entries and handling missing values to ensure the integrity of the dataset.



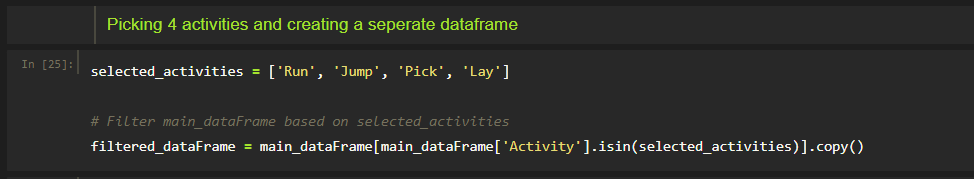
***Saving the clean data file to one file***

*Data Preprocessing*

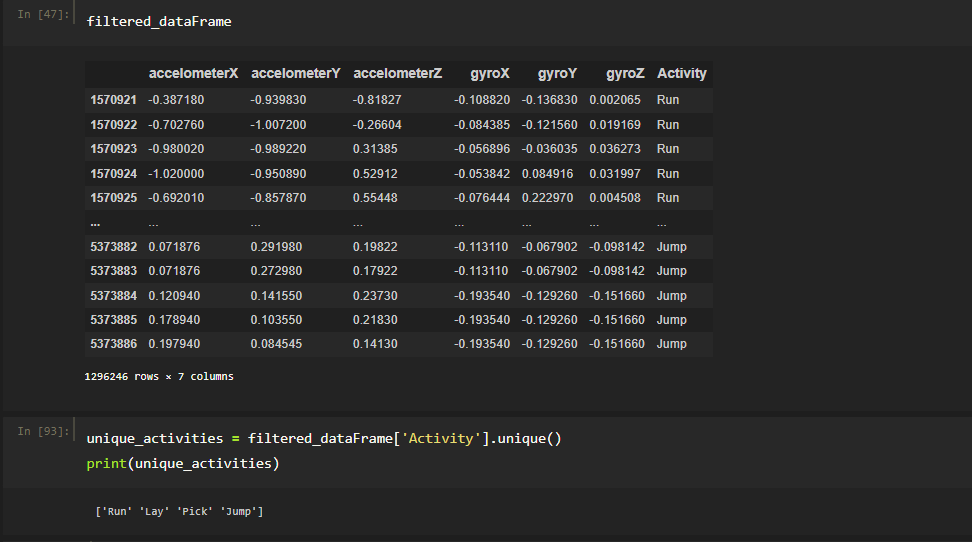
Data preprocessing involves cleaning, transforming, and organizing raw data into a usable format for analysis or machine learning.

***Importing the dataDropping autogenerated id column***

***Picking the four selected activities and merging them into one new data frame***

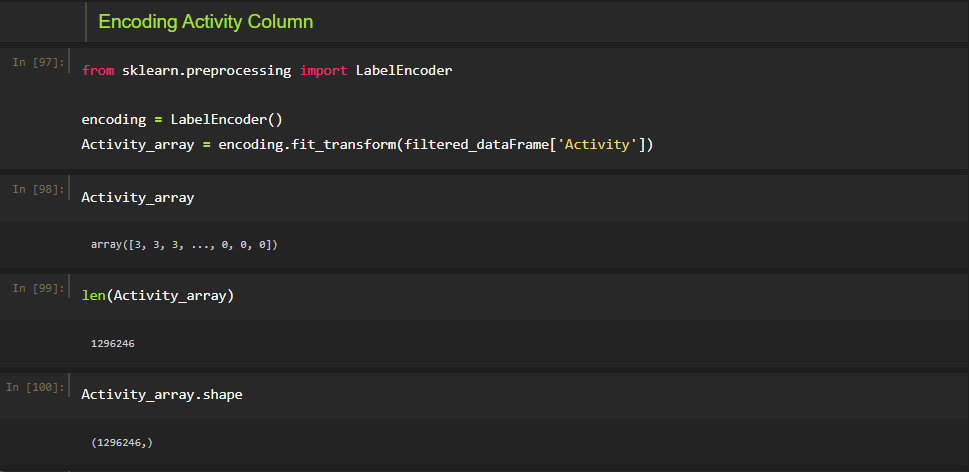
******

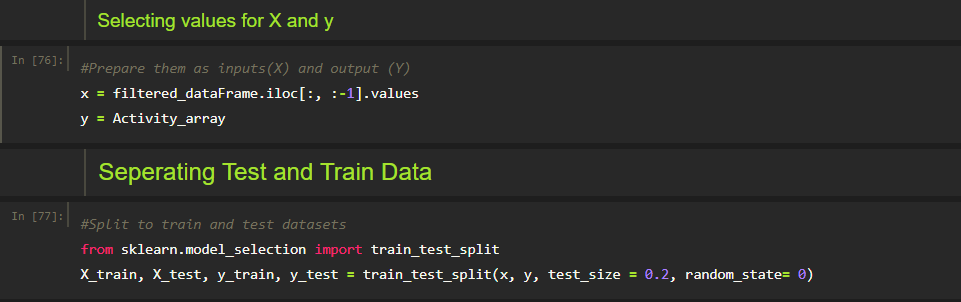
***Validating the new create data frame***

******

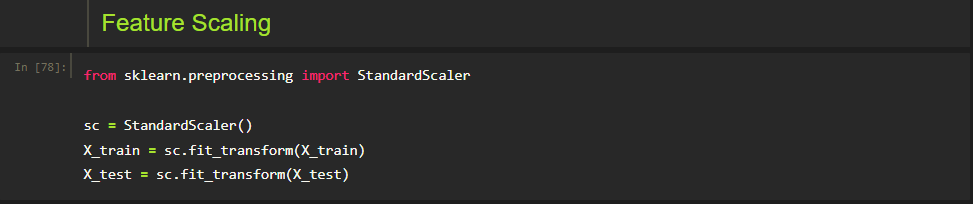
*Encoding Categorical Data*

**Converting categorical data into numerical format**

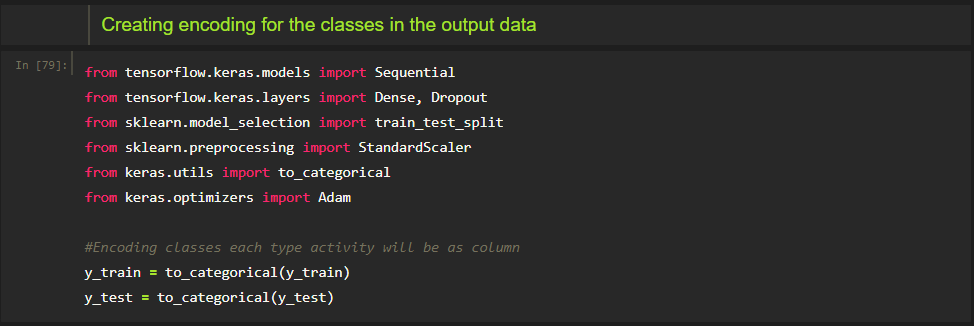
****

**Splitting the data into input (x) and output (y) variables, and further separating them into training and testing sets.**

*Future Scaling*

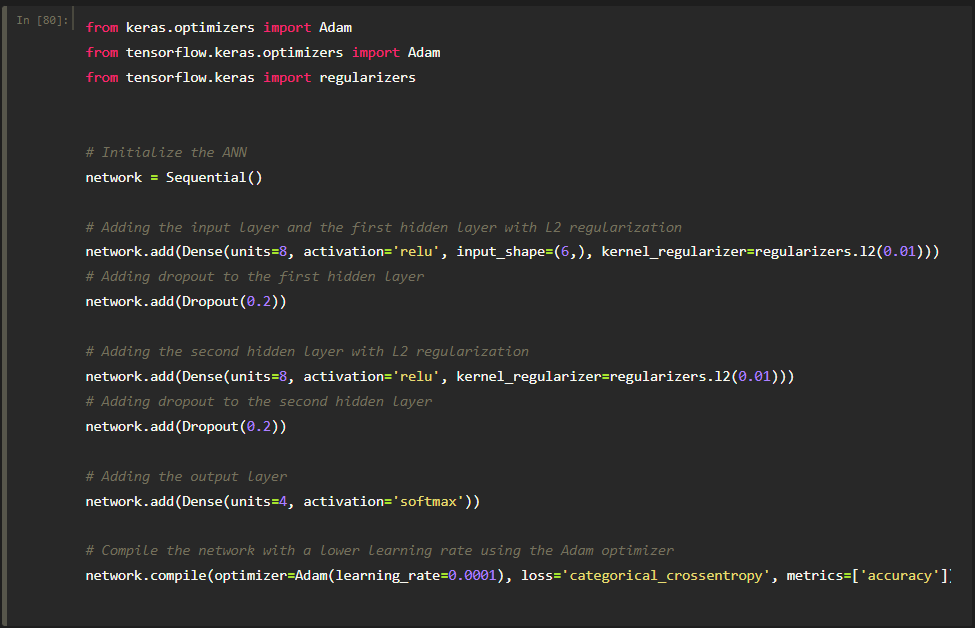
**Scaling the input features using Standard Scaler to ensure consistent scaling across the dataset**

**Converting categorical output classes into a one-hot encoded format suitable for neural network classification models using to\_categorical.**

****

*Building the model*

Building Neural network using TensorFlow's Keras API. It builds a multi-layered model with two hidden layers, applying L2 regularization to prevent overfitting. Input shape is 6 which represent our columns in the dataset and the output units 4 represents the classes(Activities).Dropout layers are inserted after each hidden layer for further regularization. The output layer uses the soft max activation function for multiclass classification. The network is compiled with the Adam optimizer and a lower learning rate (0.0001) along with categorical cross-entropy loss for training, evaluating accuracy as the performance metric.

****

*Training the model*

**Playing with the neural network units / batch size / epoch to reach better performance in our model**

*Batch size = 36*

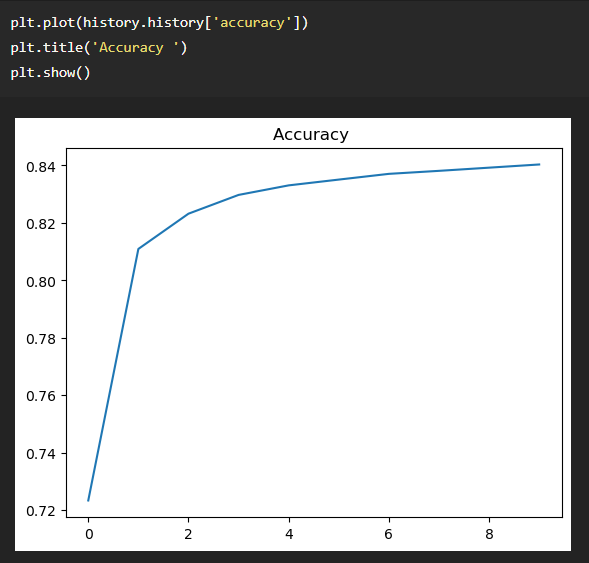
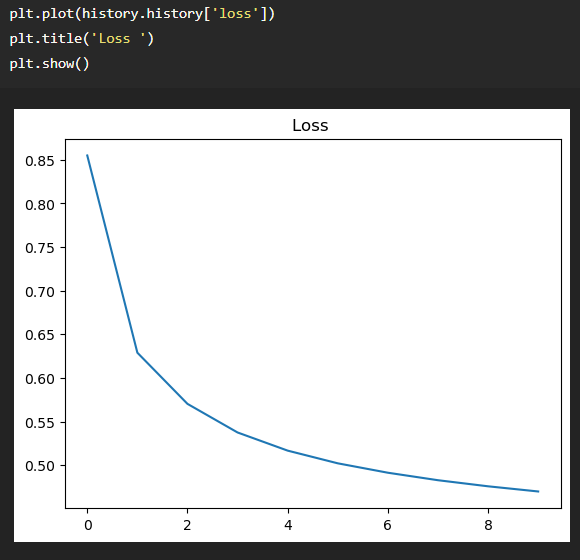
***10 Epoch: 4 -> 8 -> 8 -> 6 ==> 0.5127 Loss / 0.8228 Accuracy***



***10 Epoch: 4 -> 12 -> 12 -> 6 ==> 0.4698 Loss / 0. 8403 Accuracy***



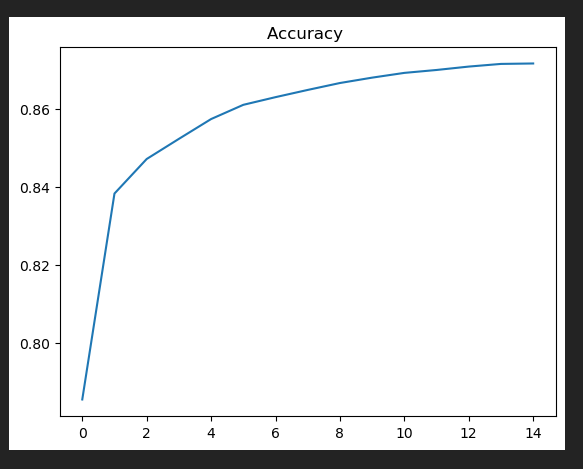
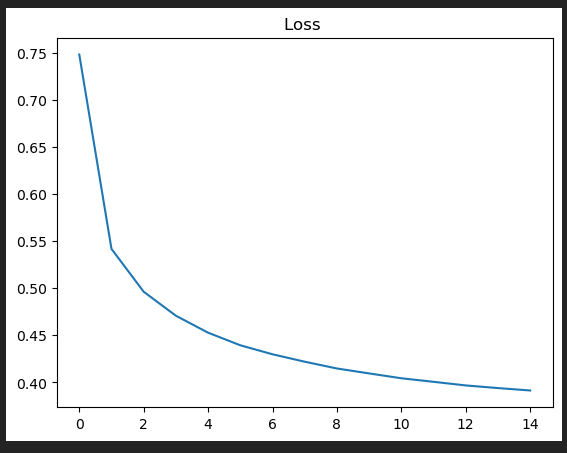
***Evaluation***



***15 Epoch: 4 -> 32 -> 32 -> 6 ==> 0.3914 Loss / 0. 8716 Accuracy***



***Evaluation***

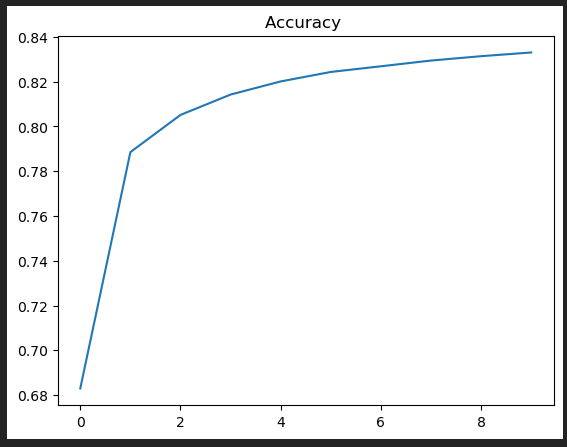
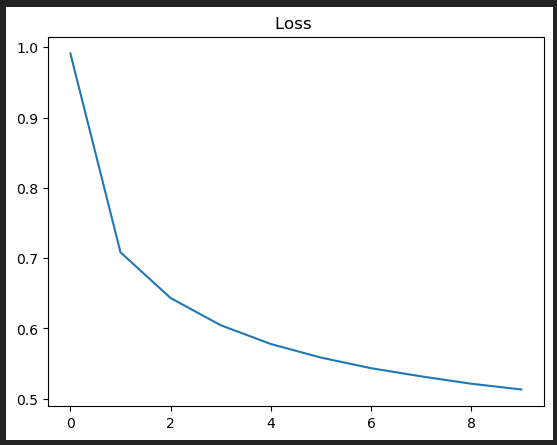


*Batch size = 64*

***10 Epoch: 4 -> 12 -> 12 -> 6 ==> 0.5131 Loss / 0.8331 Accuracy***

******

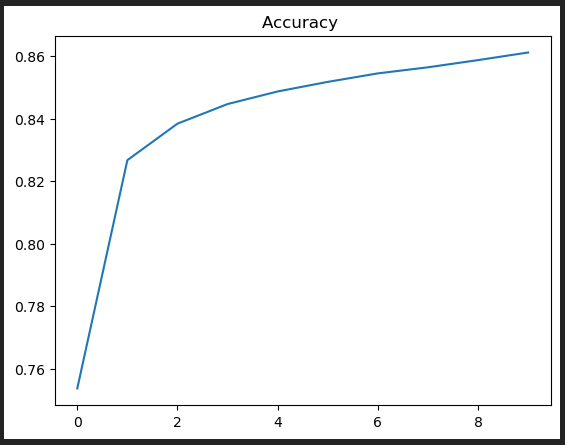
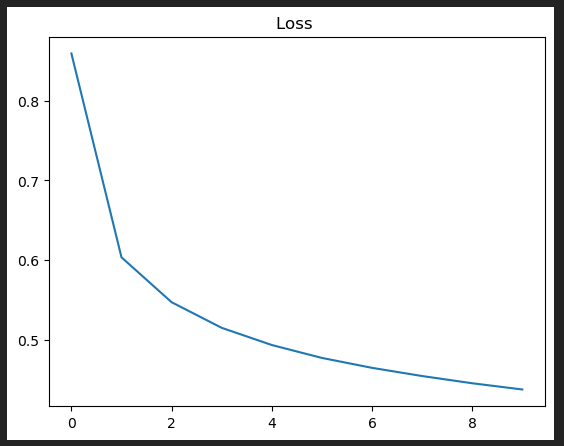
***Evaluation***



***10 Epoch: 4 -> 32 -> 32 -> 6 ==> 0.4373 Loss / 0.8612 Accuracy***

******

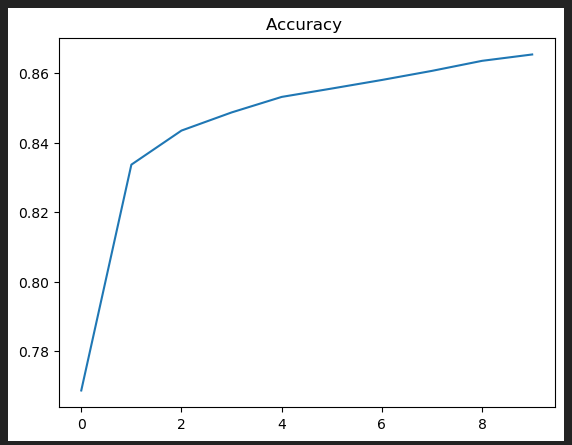
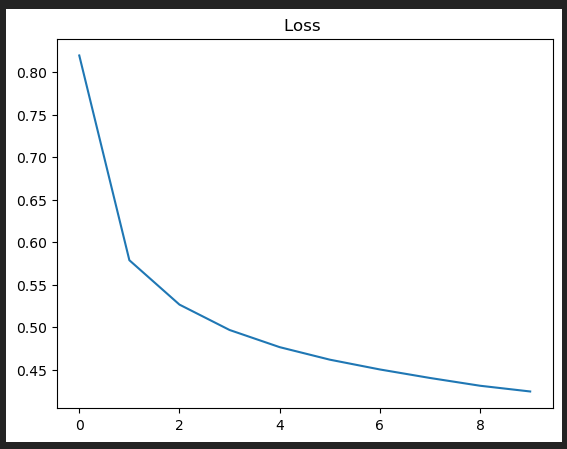
***Evaluation***

******

***10 Epoch: 4 -> 40 -> 40 -> 6 ==> 0.4243 Loss / 0.8654 Accuracy***

******

***Evaluation***

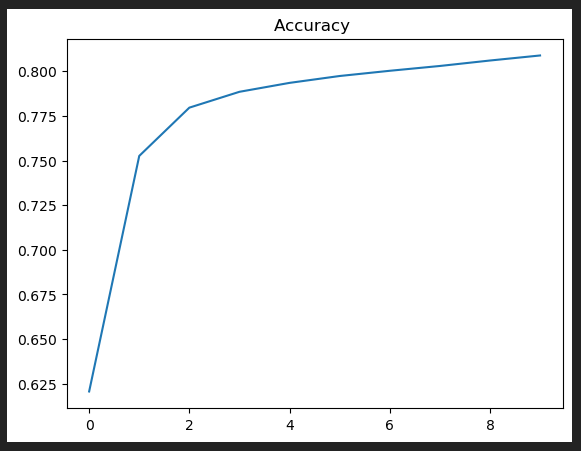
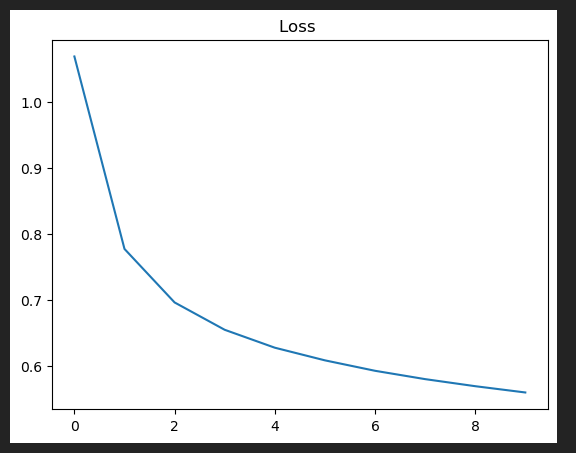
******

*Batch size = 72*

***10 Epoch: 4 -> 8 -> 8 -> 6 ==> 0.5603 Loss / 0.8088 Accuracy***

******

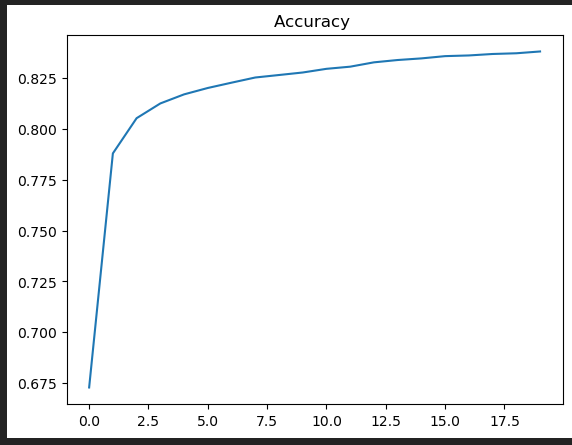
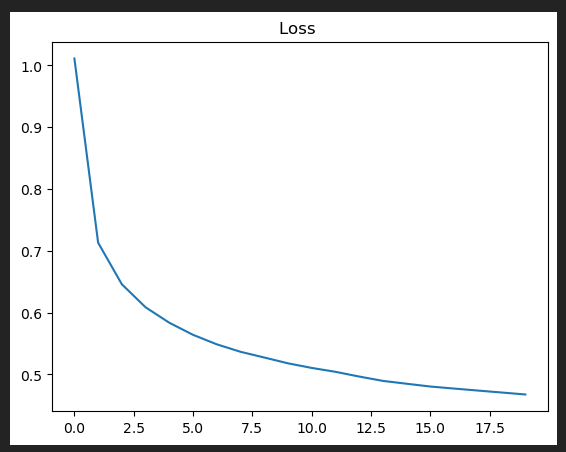
***Evaluation***

******

***20 Epoch: 4 -> 12 -> 12 -> 6 ==> 0.4677 Loss / 0.8382 Accuracy***

******

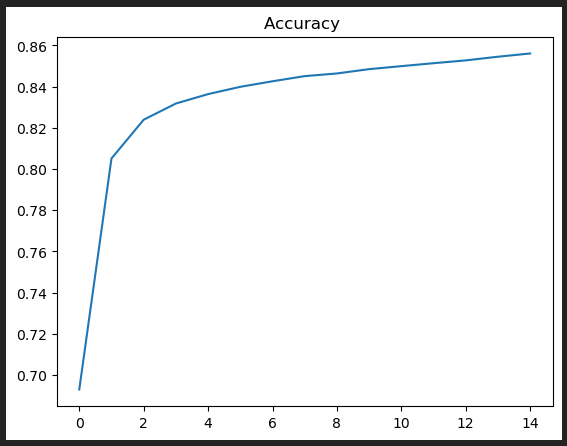
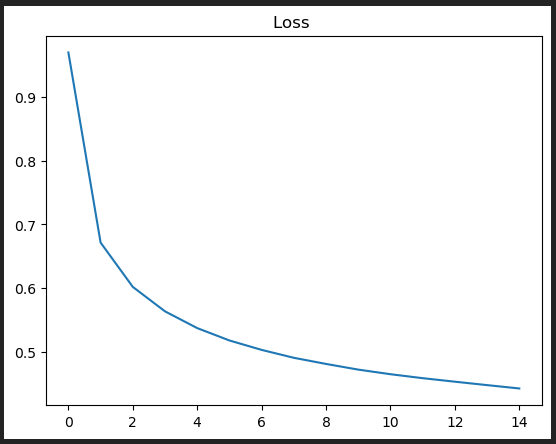
***Evaluation***

******

***15 Epoch: 4 -> 20 -> 20 -> 6 ==> 0.4677 Loss / 0.8382 Accuracy***

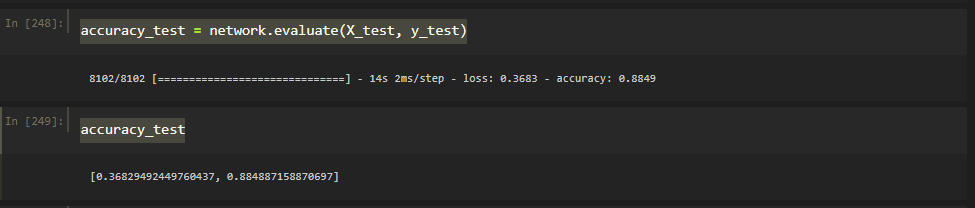
******

***Evaluation***

******

*Further Evaluation*

**Evaluation Results Analysis (for best fit model)**

****

The evaluation results, with a low loss value and a high accuracy score of 88.49% on the test dataset, indicate that the machine learning model performs well in making predictions. It demonstrates good generalization capabilities by accurately predicting labels on unseen data, providing confidence in its effectiveness for the given task

***Confusion matrix***

The confusion matrix is organized in a tabular format where rows represent the true labels from the test dataset, and columns represent the predicted labels by the machine learning model

*Breakdown of Confusion Matrix:*

***Jump:***

True Positives (Predicted correctly as Jump): 26,374 instances were correctly predicted as 'Jump.'

Misclassifications: 358 instances of 'Jump' were incorrectly predicted as 'Lay,' 8,985 as 'Pick,' and 5,255 as 'Run.'

***Lay:***

True Positives (Predicted correctly as Lay): 106,970 instances were correctly predicted as 'Lay.'

Misclassifications: 114 instances of 'Lay' were incorrectly predicted as 'Pick.'

***Pick:***

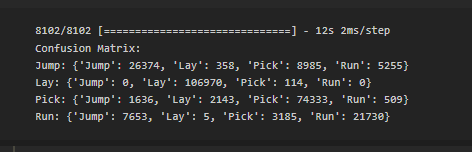
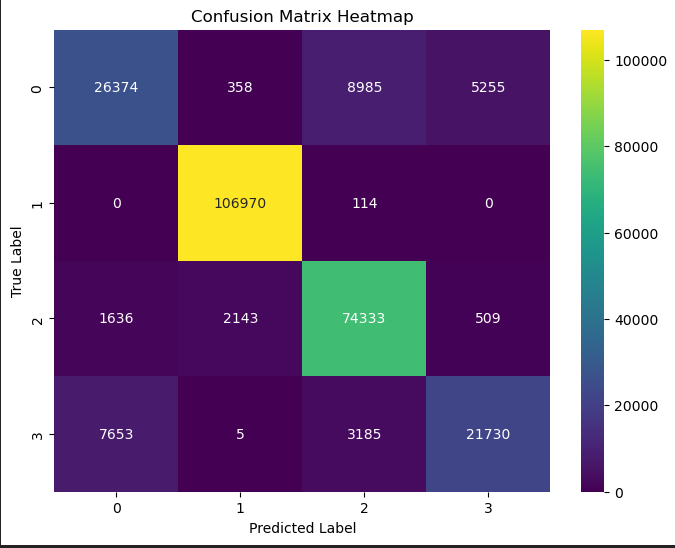
True Positives (Predicted correctly as Pick): 74,333 instances were correctly predicted as 'Pick.'

Misclassifications: 1,636 instances of 'Pick' were incorrectly predicted as 'Jump,' 2,143 as 'Lay,' and 509 as 'Run.'

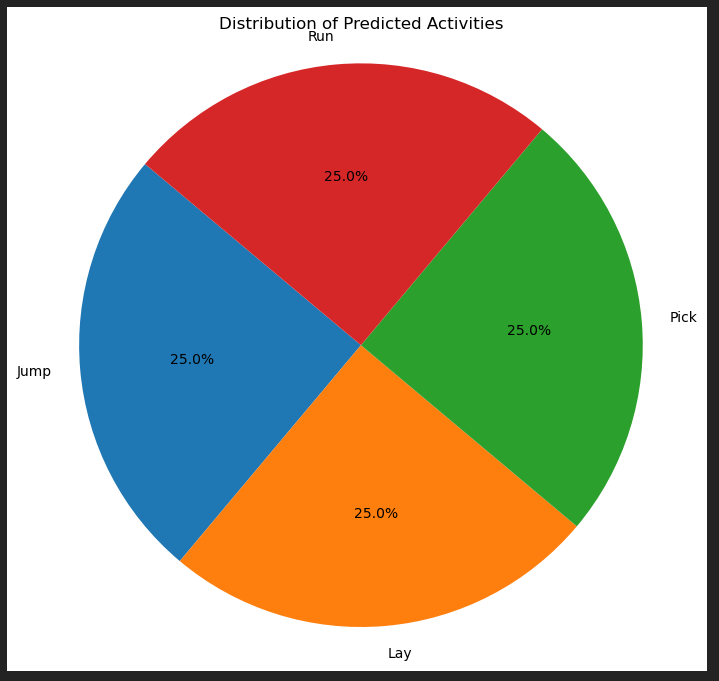
***Run:***

True Positives (Predicted correctly as Run): 21,730 instances were correctly predicted as 'Run.'

Misclassifications: 7,653 instances of 'Run' were incorrectly predicted as 'Jump,' 5 as 'Lay,' and 3,185 as 'Pick.'

********

***Pie Chart Analysis***



The generated pie chart provides a clear visual representation of the balanced distribution of predicted activity labels. It shows an equal allocation of predicted activities, each comprising approximately 25% of the total predictions made by the machine learning model. This uniform distribution indicates that the model predicts different activities with similar probabilities, suggesting a well-balanced prediction outcome across various activity labels.

*Knowledge reflection gained from the results*

**1. Model Performance:**

*The model demonstrated commendable accuracy, achieving an 88.49% score on the test dataset. This indicates a reliable ability to predict activity labels accurately.*

**2. Activity Distribution:**

*The visualizations unveiled a balanced distribution of predicted activities, with each activity label ('RUN,' 'JUMP,' 'PICK,' 'LAY') representing approximately 25% of total predictions. This signifies a well-distributed prediction outcome.*

***3. Areas for Improvement:***

*The confusion matrix revealed specific misclassifications, notably in 'JUMP' being confused with 'LAY' and 'PICK.' Addressing these misclassification patterns could enhance model accuracy.*

*Conclusion:*

*The analysis offers valuable insights into the model's performance and the dataset's characteristics. Implementing targeted improvements based on identified misclassifications and continuing to refine features could lead to a more accurate and robust model in future iterations.*