Coursework 1

Author: candidate number: GTDS3

Contents:

1. Explanation of code preparation & understanding:

1.1 Introduction of the natural of dataset:

The dataset preprocessed is Human Activity Recognition from mobile phone sensors. The dataset is collected for a Machine Learning model to predict whether a person is moving or standing. The data is collected from an IMU using mobile phone. The head 5 rows of data of the dataset are shown in Fig.1. The dataset consists of 4 types of data. The columns [accX, accY and accZ] mean acceleration collected in 3D dimension (x-direction, y-direction, and z-direction (vertically upward and downward). The datatype of the acceleration is floating number. Similarly, the columns [gyroX, gyroY and gyroZ] mean the angular velocity collected by gyroscope sensor in 3D dimension. The timestamp represents the time difference between when the mobile phone was open and when the sensor data was recorded. The format of timestamp is [minutes: seconds. Microseconds]. The timestamps generally are continuous. Every row of data is collected in a frequency of 10 Hz, which means the normal time difference between 2 data point is 0.1s. However, there are breaking points (the time difference between 2 data points is longer than 0.1s) due to different reasons. The long difference of time between 2 breaking points is caused by that the collectors collected data in different periods. However, the short difference within half second between 2 breaking points is usually caused by delay of signal transmission. The activity means whether a person is moving or stopping. 1 means that person is moving and 0 means that person stops. Therefore, for a machine learning model, acceleration and angular velocity will become as features [X] and activity will become as label [y]. The timestamp will be used to preprocess the acceleration and angular velocity of the person. Also, the timestamp, acceleration and angular velocity will be combined with a motion machining learning model to predict the trajectory of a moving person.

The thing is important to mention is that acceleration and angular velocity are continuously changing with timestamp. In this case, statical methods to preprocess the data is unsuitable for the dataset.

	accX	accY	accZ	gyroX	gyroY	gyroZ	timestamp	Activity
2	-0.496517	3.785628	8.954828	-0.142849	-0.126159	-0.022539	34:22.9	1
3	-0.462388	3.869603	9.281898	0.084349	0.096695	0.09213	34:23.0	1
4	-0.296084	3.820505	8.930728	0.061763	0.051543	0.071287	34:23.1	1
5	-0.469723	3.89011	8.744067	0.007641	0.028679	0.109433	34:23.2	1
6	-0.472418	4.109105	8.941207	-0.12364	0.099057	0.051943	34:23.3	1

Fig.1 top 5 rows of data in the dataset

1.2 Explanation of the flow of preprocessing and detailed logic of functions:

The whole data preprocessing steps are demonstrated in the Fig.2. I am going to introduce step by step.

a) Read the csv and create a data frame.
b) Detect Null value in the data frame.

c) Change the datatype of timestamp.
d) Detect the breaking points.
e) Delete the data row with the same timestamp caused.
f) Interpolate the data points to eliminate the breaking points caused by delay of signal.
g) Use rolling window to detect the outliers.
h) Smooth the data series.
i) Virtualize the results and export the data frame into csv file.

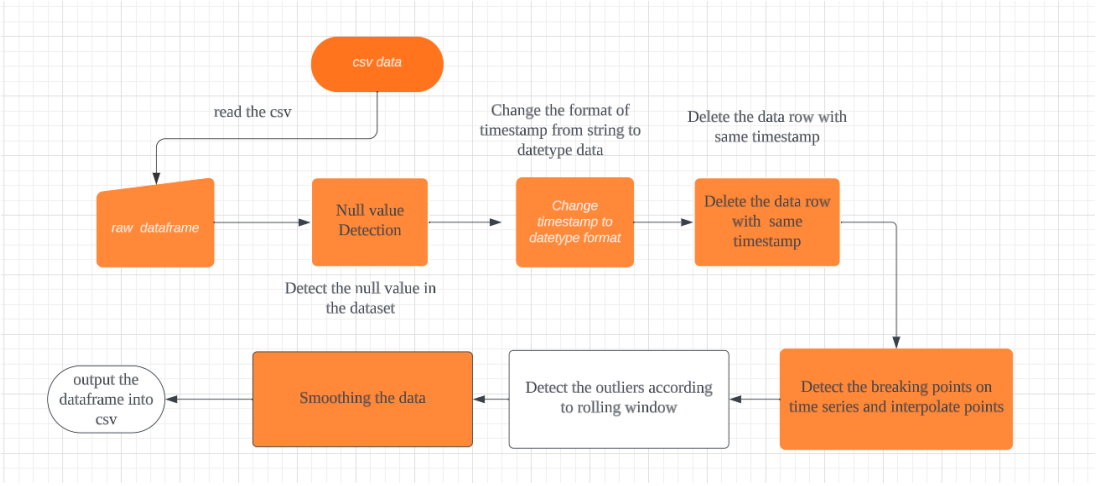

Fig.2 the flow chart of whole data preprocessing and preparation

1.2 a) Read the csv and create a data frame:

The program uses pd.read_csv() function to read the csv file and construct a data frame. The data frame consists of all the data in the excel file.

1.2 b) Detect Null value in the data frame:

The program uses a function called 'null_data_detection ()' to detect whether there is null value across all the columns in the data frame. It will return to the original data frame and print whether there is null value or not. The reason to just detect the null value without any methods to replace or delete the null values is that interpolation later will fill all the NaN data. In that case, here is no need to take actions on the null values at this stage.

1.2c) Change the datatype of timestamp:

To calculate the time difference between 2 time points, I need to convert timestamp datatype. In this case, 'time_stamp_format_convert ()' function convert the column ['timestamp'] from the string to the datetype. And then, the function appends a new column in the data frame called ['timestamp_datetype']to store the converted timestamp through the whole data frame. The conversion is necessary for the following functions such as breaking points detection, interpolation and smoothing.

Also, I manually change the content of data frame due to the wrong format of timestamp in the

function. The wrong format locates in the row number:20928. Because there is only one wrong format, I did not write a unique function for that. The wrong format of data row is shown in the fig.3.

20928	-0.413889	0.79829	9.709409	0.020137	0.019136	0.021809	05:47.8	0
20929	-0.445174	0.812959	9.782757	0.016075	-0.002261	0.023203	05:47.9	0
20930	-0.411345	0.726589	9.846674	0.010058	0.005086	0.012459	6/25/2022 14:05	0
20931	-0.490829	0.790057	9.663455	-0.042095	-0.010738	0.007971	05:48.1	0

Fig.3 Wrong format of timestamp in the excel file.

As fig.3, I just changed the timestamp to 05:48.0 to make timestamp continuously.

1.2d) Breaking points detection:

The function 'breaking_point_detection' calculates the time difference between 2 data points and detect any discontinuities. The 'breaking points' defined is the time difference between 2 data points is not equal to 0.1 seconds because the normal time difference between 2 continuous time points is 0.1 second according to the description in the introduction essay of the dataset. The sensor data is collected at 10Hz frequency. The function calculates the time difference between 2 consecutive timestamps. The function will return 2 lists. One list is called 'breaking_point_index' and another is called 'list_of_difference'. 'breaking_point_index' contains the index of breaking points and 'list_of_difference' contains the time difference between 'breaking points'. In this case, 2 lists contain all the abnormal points with time difference unequal to 0.1 seconds. The lists will contain the normal breaking points caused by that the collectors collect data in different periods and contain data points with repeated timestamp and timestamp out of sequence.

1.2e) Delete the data row with the same timestamp caused:

The function 'timestamp_delete' is used to delete the data points with repeated timestamp. The function will call the function 'breaking_point_detection' to get 2 lists: 'list_of_difference' and 'breaking_point_index'. The function scans all the time difference which the difference equals to zero. When the time difference is identified, the corresponding data row will be deleted according to its index. The function is used to keep the time stamp is unique in the time series.

1.2f) Interpolate the data points to eliminate the breaking points caused by delay of signal:

To keep the continuity of the timestamp, interpolation is necessary to handle discrepancies in the time series. The function 'interpolation' solves situations where data points have an interval of 0.2 seconds between each other. The function interpolates a row of new data between the original data points to keep the time series continuously. Similarly with 1.2e), the 'interpolation' will also get 2 lists: 'list_of_difference' and 'breaking_point_index' through the function 'breaking_point_detection'. The function will define the interpolation points through the 2 lists and put the indices of the points into the 'list_of_interpolation'. Then, the function segments the original data frame and creates a new row with 0.1 second time difference before the next timestamp. The new row consists of NaN values and an interpolated timestamp in the row of 'timestamp datetype'. The insertion is finished by concatenated all the segmented data frame with new rows to keep a continuously sequence and reset the new index of data frame. After the concatenation,

interpolation is provided with a method PCHIP (Piecewise Cubic Hermite Interpolating Polynomial). The method PCHIP saves the shape of the data points and provides an accurate approximation on the value of the acceleration and angular velocity. After the interpolation, the disposal data frame is returned. If the original data frame has a NaN value, the data with NaN value will be interpolated automatically by interpolation.

1.2 g) Use rolling window to detect the outliers:

'outliers_detection' is the function which uses rolling window to detect outliers. Outliers are normal in the time series and extremes are valuable to mark in the time series. Rolling window is a method used to detect outliers in a time series. The window takes a subset of data frame with a certain time interval defined by window size and detects the outliers according to the standard deviation and mean of the subset. The window proceeds with the time series and report the outliers which is higher than the upper bound or lower than the lower bound. The upper bound and lower bound are calculated by the mean of data in subset plus or reduce three times standard deviation of the subset. The data will be stored in a dictionary. In this case, I do not do anything on the outliers at this stage because the following smoothing steps will take a more reasonable behavior on the outliers.

1.2 h) Smooth the data series:

The function 'smoothing_all' is used to smooth the acceleration and angular velocity in the data frame according to breaking points. The reasons for smoothing are minimizing the effect of outliers, diminishing the noise of data collected from sensors and maintain the shape of the data. The function uses a sub-function 'smoothing' to smooth every segment of original data frame segmented by breaking points generated by the function 'breaking_point_detection'. Every segment of data frame is in a continuous timestamp. The smoothing method used in the function is ewm function (Exponential Weighted Moving). It works well on time series data and gives a more sophisticated view on disposal of outliers. The alpha in Exponential Weighted Moving indicates the difference between the weight of recent timestamp and the weight of older timestamps. If the value of it tends to be 1, it means that data collected in recent timestamps plays a much more important role that the data collected in older timestamps. If the alpha tends to be zero, it means the older timestamps become important on evaluation of smoothing. After smoothing, the data preprocessing stage is finished.

1.3 i) *Virtualize the results and export the data frame into csv file:*

Finally, we would like to divide the data frame into different segments according whether Activity is moving or not and virtualize the results. The reason of segmenting data frame according to Activity column is that Moving and Stopping have significantly difference in acceleration and angular speed. There is not any statistically meaning to analysis outliers and standard deviations in that case. The boxplots and histograms are plotted for the users to analysis the data. Virtualize results after data preprocessing are listed in the following page.

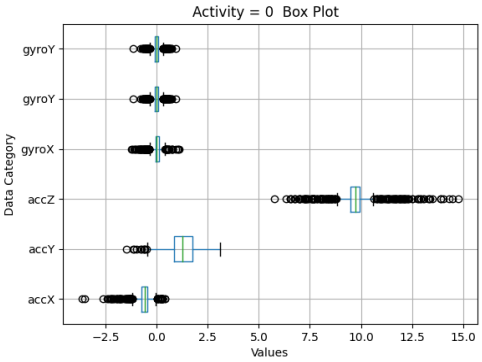

Fig.4 Boxplot of activity 0

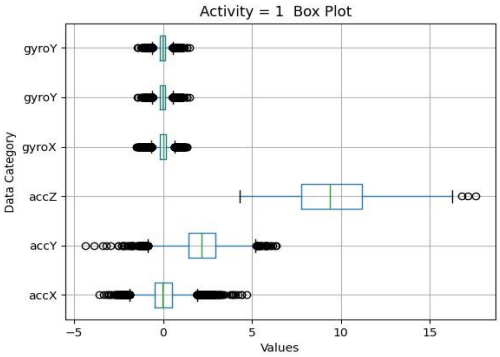

Fig.5 Boxplot of activity 1

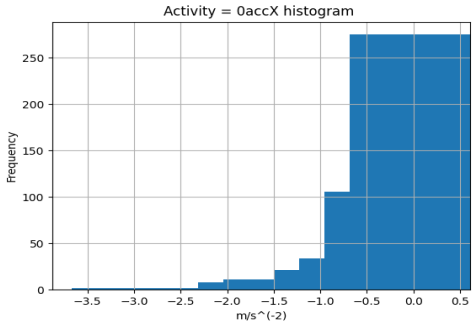

Fig.6 Histogram of accX when activity =0

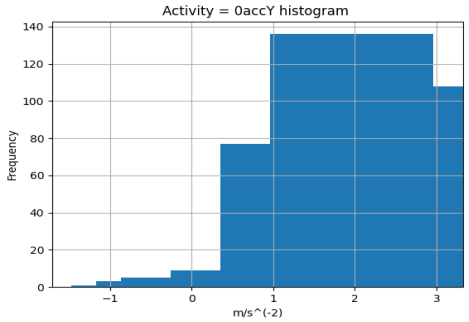

Fig.7 Histogram of accY when activity =0

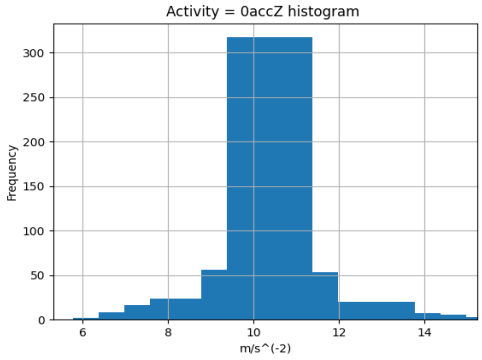

Fig.8 Histogram of accZ when activity =0

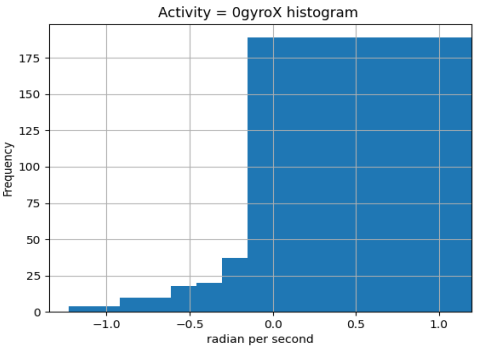

Fig.9 Histogram of gyroX when activity =0

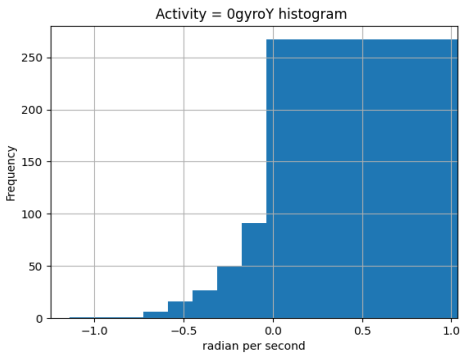

Fig.10 Histogram of gyroY when activity =0

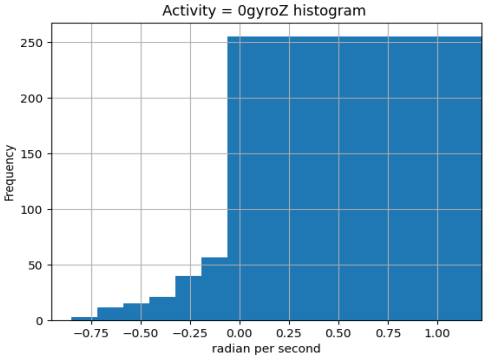

Fig.11 Histogram of gyroZ when activity =0

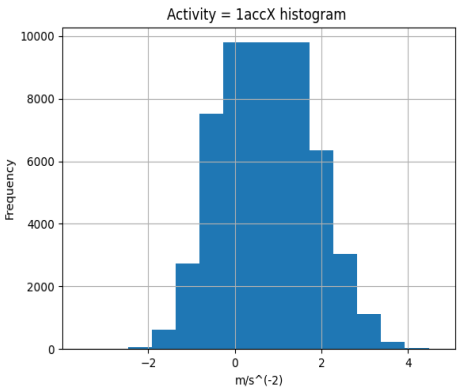

Fig.12 Histogram of accX when activity =1

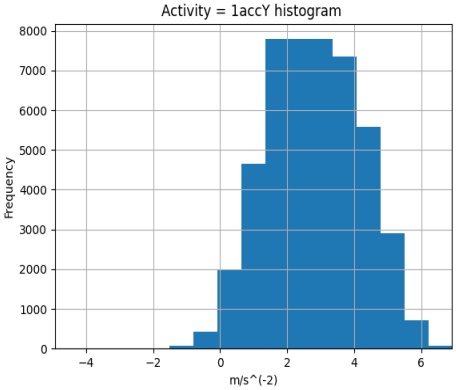

Fig.13 Histogram of accY when activity =1

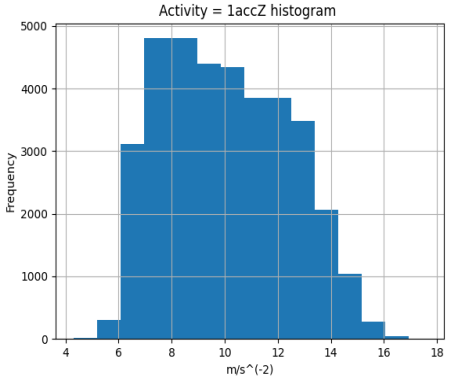

Fig.14 Histogram of accZ when activity = 1

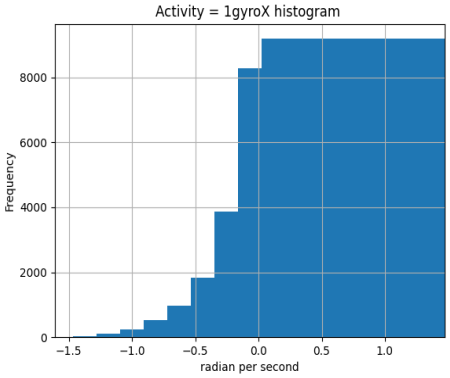

Fig.15 Histogram of gyroX when activity =1

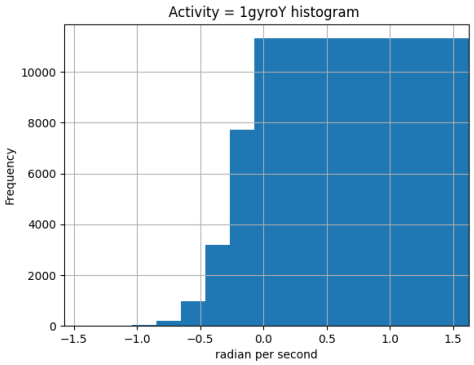

Fig.16 Histogram of gyroY when activity = 1

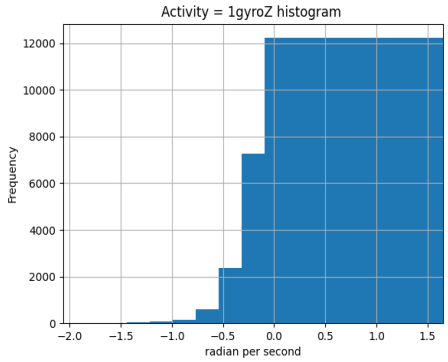

Fig.17 Histogram of gyroZ when activity =1

From the histograms shown, we would like to see my segmenting according to activities work. The Acceleration and angular velocity are fitted in normal distribution. According to the box plots of activity = 1 and activity = 0, there are many outliers analyzed by a traditional method to check the outlier. In this case, the dynamic methods such as rolling window and smoothing have more statistics meaning in time series. If we just replace all the outliers using traditional methods, the pattern and structure of data will be lost. The preprocessing method I used is to keep the true signal and characteristics of the data in the time series.

\

2. Product Overview

APP1:

The API is used for programmers to access the dataset which used the IMU (inertial measurement unit) to collect Acceleration and Angular velocity over a period. Developers could integrate the API into their own software. The API will allow developers to request and manipulate the dataset effectively. The product will concentrate on improving performance and allow users to update in real time. The API can play an important role in many applications like mapping software and health monitoring software on the mobile phone because it could provide the data basis to developers. The developers can predict the speed, routing, and location of the user over a certain time.

App2:

The App2 will deploy a machine learning model according to the acceleration and rotation speed in the dataset by App1 to predict whether a person is moving. The model will also predict the speed, rotation angel and moving routing of a person if they are walking. To give a detailed description of the routing, the moving trajectory of a person moving will be visualized on a 3-D spatial map. The predicted results (speed and rotation angle) will be exported as csv also. The App2 will concentrate on how to demonstrate these data to users with visual impact and design a user-friendly interface especially for somebody who does not have prior programming knowledge and physics knowledge. The App2 aims to assist individual athletes or sports teams to evaluate their performance when they are moving. The trainer can collect athletes' acceleration and angular speed by IMU to draw player performance statistics through the App2 and help athletes to train or perform better in the competition.

3. Persona: