**Just Taxi Software Application**

# ST1508: PRACTICAL AI

# CA1 Assignment

CHONG JUN HUI (P2021094)

JUMANA HASEEN (P2021627)

TUSHITA GOVINDARAJ (P2012155)

DAAA/FT/2B/02

# **Data Engineering (SQL database)**

## **Entity Relationship Diagram**

Diagram

Description automatically generated

There are 2 tables named Features and Labels.

In features table, there are 12 columns. ID will be the primary key and the bookingID will be the foreign key referencing the bookingID column in the labels table.

In the labels table, there are 2 columns and bookingID is the primary key.

The features table and the labels table have a many to one relationship. Many records of a bookingID can only have 1 label but a bookingID can have many records. Each record must have a bookingID and each record must have a label.

## **SQL Database Setup**

### **Data Description**

There are 5 features CSV files. All the CSV files have the 12 columns, and they provide data recorded by the GPS sensor for each trip during a specific period. Data recorded includes the accelerometer and gyroscope readings, speed of taxi as well as accuracy and bearing of GPS sensor. The table below shows the description of the dataset:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Feature Name | Description | Data Type |
| 1 | featuresID | Identifies each record in column | Int |
| 2 | bookingID | Identifies each trip by taxi | Float |
| 3 | Accuracy | Accuracy inferred by GPS (in metres) | Float |
| 4 | Bearing | GPS bearing (in metres) | Float |
| 5 | acceleration\_x | Accelerometer reading at x axis (m/s2) | Float |
| 6 | acceleration\_y | Accelerometer reading at y axis (m/s2) | Float |
| 7 | acceleration\_z | Accelerometer reading at z axis (m/s2) | Float |
| 8 | gyro\_x | Gyroscope reading in x axis (rad/s) | Float |
| 9 | gryo\_y | Gyroscope reading in y axis (rad/s) | Float |
| 10 | gryo\_z | Gyroscope reading in z axis (rad/s) | Float |
| 11 | second | Time of the record by number of seconds | Float |
| 12 | Speed | Speed measured by GPS in m/s | Float |

The total number of records in all 5 CSV files apps up to 7,469,656 records.

Whereas the labels table 2 columns and 20018 records and it provides information on whether a driving trip is dangerous or safe. The 2 columns in the table are bookingID and label. The bookingID column represents each bookingID in features table. The labels column indicates the safety of trip where 0 being safe trip and 1 indicating dangerous trip.

### **Data Preparation**

All the 5 features CSV files along with the label CSV file were imported into a database named JustTaxiDB inside the Microsoft Manager Server Management Studio. Hence, 5 different features tables and 1 labels table were created

All 5 features tables were merged to create the final features table. The featuresID column was dropped and replaced with a new identity column named ‘ID’ which is made the primary key for the features table.

For the labels table, we realised that there were duplicate records for 18 bookingIDs indicating the trips were both safe and dangerous, which was not appropriate. Hence, we decided to remove all duplicated records and thus, a total of 36 records were removed. These 36 records were only 0.2% of the whole labels dataset which meant that the removal of these records would not have a big impact in later stages of the assignment.

Since, the duplicated records for bookingIDs were removed, the bookingID column had only unique values and therefore, it was made the primary key for the labels table. This column was then referenced to the bookingID column in the features table as foreign key.

Since the 18 bookingsIDs were removed in the labels table, there should not be any existing bookingIDs in the features table thus we removed the 8,742 records of the same bookingIDs.

The final features dataset is of shape (7460914, 12) and final labels dataset is of shape (19982,2). The first 5 records of the final features and labels tables is shown below:

Graphical user interface, application

Description automatically generated

## **SQL complex queries**

### **Query 1**

*SELECT bookingID, ROUND(Speed,2) 'Speed (m/s)', ROUND(acceleration\_x,2) 'acceleration\_x', ROUND(acceleration\_y,2) 'acceleration\_y', ROUND(acceleration\_z,2) 'acceleration\_z', ROUND(gyro\_x,2) 'gyro\_x', ROUND(gyro\_y,2) 'gyro\_y', ROUND(gyro\_z,2) 'gyro\_z', label*

*FROM(SELECT f.bookingID, Speed, acceleration\_x, acceleration\_y, acceleration\_z, gyro\_x, gyro\_y, gyro\_z, l.label*

*RANK() OVER (PARTITION BY f.bookingID ORDER BY Speed DESC) AS ranking*

*FROM features f, labels l*

*WHERE f.bookingID=l.bookingID AND l.label=1*

*GROUP BY f.bookingID, Speed, acceleration\_x, acceleration\_y, acceleration\_z, gyro\_x, gyro\_y, gyro\_z, l.label) AS subquery*

*WHERE subquery.ranking=1*

Table

Description automatically generated**Insights:** National studies suggest that speeding is a factor of at least 25% of traffic accidents. Hence, the changes in the readings of accelerometer and gyroscope when speed is the highest was analysed. The acceleration results above shows that when a vehicle’s speed is at the highest, acceleration x have both negative and positive values while acceleration y and z has mostly positive values. Thus, suggesting acceleration x does not affect speed but acceleration in the forward and upward direction does. For gyro, at high speed, the taxis seem to rotate a lot.

**Recommendation:** When there is a lot of fluctuation in the gyroscope readings and when acceleration y and z is high, the taxi might have chances of leading to an unsafe trip. Therefore, the company could take this into account when classifying a taxi trip as dangerous or safe.

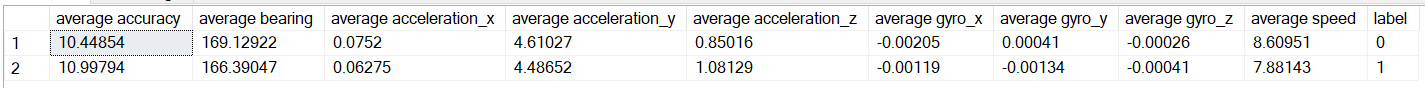
### **Query 2**

*SELECT ROUND(AVG(averageaccuracy),5) as 'average accuracy', ROUND(AVG(averagebearing),5) as 'average bearing', ROUND(AVG(averageacceleration\_x),5) as 'average acceleration\_x', ROUND(AVG(averageacceleration\_y),5) as 'average acceleration\_y', ROUND(AVG(averageacceleration\_z),5) as 'average acceleration\_z', ROUND(AVG(averagegyro\_x),5) as 'average gyro\_x', ROUND(AVG(averagegyro\_y),5) as 'average gyro\_y', ROUND(AVG(averagegyro\_z),5) as 'average gyro\_z', ROUND(AVG(averagespeed),5) as 'average speed', subquery1.label FROM (*

*SELECT c.bookingID, AVG(Accuracy) as 'averageaccuracy', AVG(Bearing) as 'averagebearing', AVG(acceleration\_x) as 'averageacceleration\_x', AVG(acceleration\_y) as 'averageacceleration\_y', AVG(acceleration\_z) as 'averageacceleration\_z', AVG(gyro\_x) as 'averagegyro\_x', AVG(gyro\_y) as 'averagegyro\_y', AVG(gyro\_z) as 'averagegyro\_z', AVG(speed) as 'averagespeed', c.label*

*FROM features f INNER JOIN (SELECT e1.bookingID, e1.label FROM labels e1*

*INNER JOIN (SELECT e3.bookingID FROM labels e3 GROUP BY e3.bookingID HAVING COUNT(e3.bookingID) = 1) e2 ON e1.bookingID = e2.bookingID) c ON f.bookingID = c.bookingID GROUP BY c.bookingID, c.label) subquery1 GROUP BY subquery1.label*

**

**Insights:** We analysed the overall difference for all the readings for safe and dangerous trips. There seems to be no significant difference in any of the columns in comparison. The accuracy of safe trips is lower than that of dangerous trips which is expected since the lower the GPS accuracy, the better. Acceleration x does not seem to have any impact on safe or dangerous trips, but acceleration y and z do have some impact. Dangerous trips have less forward but more upward acceleration than safe ones. As expected, there is a lot of fluctuation in the gyroscope readings for all axes. Surprisingly, the average speed for safe trips is greater than that for dangerous trips.

**Recommendation:** The taxi company managers could use these values as the boundary for each feature in order to consider a taxi trip either safe or dangerous.

### **Query 3**

*SELECT subquery1.bookingID, [The biggest difference in time (s)], [Avg no of seconds taken for the sensor to record] FROM*

*(SELECT difft.bookingID, max(difft.diff) AS 'The biggest difference in time (s)' FROM*

*(SELECT bookingID, second, second - lag(second, 1) OVER (PARTITION BY bookingid ORDER BY second) AS diff FROM features) difft*

*GROUP BY difft.bookingID) AS subquery1,*

*(SELECT difft.bookingID, ROUND(AVG(difft.diff) OVER(PARTITION BY difft.bookingID),2) AS 'Avg no of seconds taken for the sensor to record' FROM*

*(SELECT bookingID, second, second - lag(second, 1) OVER (PARTITION BY bookingid ORDER BY second) AS diff FROM features) difft) AS subquery2*

*WHERE subquery1.bookingID=subquery2.bookingID*

*GROUP BY subquery1.bookingID, subquery1.[The biggest difference in time (s)], subquery2.[Avg no of seconds taken for the sensor to record]*

*ORDER BY 2 DESC*

*Table

Description automatically generated***Insights:** The second column shows the greatest difference between the 2 consecutive records. The first 4 bookingIDs have a jump of more than 600000000 seconds to the next record. This means that the sensor did not record down any records for more than 600000000 seconds (6944 days) and it leads to irregularity of the data. Looking at the rest of the records, we can see that the sensor has only recorded down the next record after the corresponding number of seconds in the column. The data that is not recorded by the sensor may be crucial for identifying the type of trip. The sensor should be recording the data at a fixed or regular pace. The second column shows the average number of seconds it took for the sensor to record down. The first 4 bookingIDs have a big value due to the 600000000 seconds in the data. We can see that for the 5th row, the sensor has an average of 128 seconds to take down a record. This means that for every 128 seconds, the sensor takes down a record. This number is too big as the sensor may have not recorded important points of the trip causing an inaccurate result for the type of trip.

**Recommendation:** The taxi company should get a better sensor so that the sensor can record down more records to get a better understanding of the trips and lead to better accuracy to the result of the bookingID.

# **ETL pipeline**

## **Extract:**

### **Connect to SQL Server database system**

1. Connection was first established from the jupyter notebook to the SQL Server Database System.
2. The method used was create\_engine which is defined under SQLAlchemy which takes arguments (connect string) to make connection the Database.

### **Extract Data from Database System**

1. The engine.connect() method was used to return the connection object.
2. The connection.execute() method was used to create a database cursor that enables transversal over the rows of a result dataset.

## **Transform (Data Pre-processing):**

Only the features dataset had a lot of cleaning to be done. Since the labels dataset only has unique bookingIDs with a label indicating whether each trip is dangerous or safe, there is nothing to clean for the dataset.

### **Identifying and Removing Outliers**

We are going to remove all the outliers in the dataset so that the model would be more accurate. However, when we plotted the boxplots, we realise that there were a lot of outliers and removing all of them would lead to loss of data. Thus, we decided on the threshold for the outliers and remove them. We researched on all the columns, and we reached an agreement on the threshold for the outliers.

Chart

Description automatically generatedChart, box and whisker chart

Description automatically generatedChart

Description automatically generated with medium confidence

For column ‘acceleration\_x’, we decided that values under and above would be considered as an outlier as internet sources showed that moving vehicles do not exceed -10 or 10m/s2. Thus, any values outside of that range would seem to be inaccurate data as it seems impossible for a car to accelerate that fast.

For column ‘bearing’, there were no outliers to remove as all the values were between 0 to 359 and boxplot shows a normal distribution.

For the column ‘accuracy’, we chose to delete accuracy that is above 1500 because there was not too much data above accuracy of 1500 and an accuracy above 1500 was considered as an irregularity.

Chart, scatter chart

Description automatically generatedChart, box and whisker chart

Description automatically generatedChart, box and whisker chart

Description automatically generated

For column ‘gyro\_x’, we removed records that had below -1rad/s or above 1rad/s because for a car to have a rate of rotation of 57 degrees in a second seemed impossible thus we removed values outside of that range.

For column ‘acceleration \_z’, records that were below or above because internet sources showed that moving vehicles do not exceed -15m/s2 or 15m/s2. Thus, any values outside of that range would seem to be inaccurate data as it seems impossible for a car to accelerate that fast.

For column ‘acceleration\_y’, we considered records that had below or above because internet sources showed that moving vehicles do not exceed -206m/s2 or 256m/s2. Thus, any values beyond that range would be considered irregularity.

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generatedChart

Description automatically generated

For column ‘seconds’, we removed records that showed that the taxis were travelling for more than a day as it seems impossible for cars to be travelling for a day for a trip. Box Plot also showed that outliers were around 6,000,000 seconds thus giving the idea that these values are wrongly inserted.

For column ‘gyro\_z’, we removed records that had below -7rad/s or above 7rad/s because for a car to have a rate of rotation of 401.07 degrees in a second seemed impossible thus we removed values outside of that range.

For column ‘gyro\_y’, we removed records that had below -6rad/s or above 6rad/s because for a car to have a rate of rotation of 343.775 degrees in a second seemed impossible thus we removed values outside of that range.

### **How we managed the large dataset**

As the dataset was too large, it took a long time for us to load the dataset into our computer thus we decided to split the dataset into 3 parts and each part will be given to each of our team members.

We spilt the dataset by bookingID. The first set contains 2,900,342 records. The second sets contain 2,515,201 records. The third set contains 1,691,277 records. In total, these 3 sets have a total of 7,106,820 records. This was after the removal of outliers. Hence, with the respective set of data given to each of us, we proceeded to handle the null records in each of the dataset.

### **Handling Null values**

#### **Accuracy column**

For the ‘accuracy’ column, we decided to replace the null values with the mode for each bookingID. Our assumption is that the best accuracy value for a trip represents the overall GPS accuracy for the trip

#### **Bearing column**

For the ‘Bearing’ column, we decided to replace the null values with 0. Since we were unsure of which direction the vehicle was heading towards, we decided to keep the missing value to a neutral value of 0 which indicates the North direction.

#### **Speed, Acceleration and Gyro columns**

For the acceleration and gyro columns for all axis, we decided to replace the null values with the median based on the value before and after the null record. When grouped by ‘bookingID’ column and ordered by the ‘seconds’ column, we realised there were some null values for the very first record or the last record.

Since we cannot possibly find the median for these records, the standard deviation for the respective columns for each bookingID was found first. Then, for null values at the start, they were replaced with value obtained from subtracting the standard deviation from the next value in the respective column. Whereas null values at the end were replaced with the value obtained from adding the standard deviation from previous value in the respective column.

Similarly, for consecutive null values, they were replaced with values by simply adding the standard deviation with the previous records.

#### **Seconds column**

For the ‘seconds’ column, it did not seem appropriate for us to replace null values with average, mode, median, the previous or the following value. Hence, since the number of null values in seconds column is 96,882 which is not significantly a lot, we decided to remove all the null values.

## **Loading**

After removing the respective outliers as well as null values, we were left with 7,194,253 records out of the initial 7,469,656 records. Therefore, overall, after the whole data cleaning process we were able to still retain approximately 96% of the original data.

All 3 sets of cleaned data were concatenated to form the final features dataset and loaded in a CSV file named ‘final\_features.csv’.

# **Data Visualization**

For visualization purposes, the final features and labels datasets were merged to form a new dataset. The final dataset has 7,194,253 records and 12 columns. A preview of the dataset is shown below:

Table

Description automatically generated

Below shows the information about the dataset as well as that there are no null values.

Table

Description automatically generatedTable

Description automatically generated

Chart, pie chart

Description automatically generated

### **Pie Chart**

The pie chart shows that ¾ of the trips are safe and ¼ of it are dangerous. This suggests that there is a 25% chance of a trip being dangerous and 25% probability is a very high chance for a Taxi company. The company should decrease their chances of a dangerous trip to maximum 5% so that the passengers feel safer in the taxis.

### **Correlation Heatmap**

According to Our Kloud, most GPS tracking devices are accurate to within three meters. Hence, two heatmaps were done to visualize the correlation between all the variables for records with accuracy within 3.

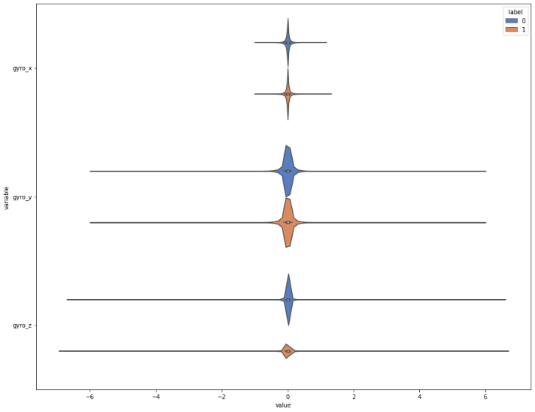
This chart shows the correlation for safe trips. Overall, the correlation between the variables is quite low. However, among all the correlation values, speed and acceleration y seem to be the most correlated with a value of 0.121. While correlation between second and gyro z seems to be the lowest with a value of 0.00036 which is very close to 0. There are some variables which are negatively correlated. Speed and acceleration x has the highest negative correlation of -0.567.

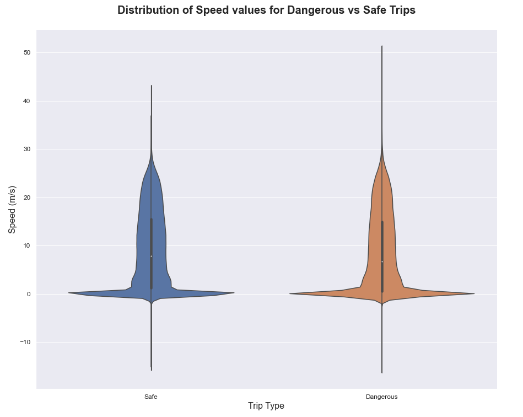
Chart, treemap chart

Description automatically generatedThis chart shows the correlation for dangerous trips. Like the chart for safe trips, the correlation between variable here is also quite low generally. Among all the correlation values, correlation between speed and second is the highest with a value of 0.129. While correlation between acceleration y and acceleration x is the lowest with correlation value of 0.00012. Among the variables which are negatively correlated, speed and acceleration y has the highest negative correlation value of -0.301.

All in all, there are no variables that seem to be showing a significant correlational relationship.

### **Violin Plots**

This is a violin plot showing the distribution of the gyro for safe and dangerous labels. The overall shape and distribution of each group of gyros are similar for both types of trips. The median, which is around 0 radian, is similar for all gyros. The interquartile range for the 2 labels is very small and similar for both types of trips. The interquartile range is small because the values for gyro are in radian, not degree. Radians are in smaller values meaning a radian of 1 is equivalent to 57.2958 degrees. Comparing the length of the line for each violin plot, we can see that the range for each gyro and type of trip is similar. The only difference that we can see is the line of dangerous trip for gyro\_z is slightly longer than the line of the safe trip. This would mean that there are larger range of values for dangerous trip in gyro\_z than the safe trips. The distribution of the safe trips and dangerous trip for gyro\_x are similar. There are many records that have values of gyro\_x around 0 radian. For gyro\_y, there are many values that also around 0 but as compared to gyro\_x, the values are more distributed. For gyro\_z, the shape of the distribution for both types of trips are pointed at the sides. This means that very high percentage of the records are close to 0.

This is a violin plot of speed for safe and dangerous trips.

This plot is done to visualize the distribution of speed for the different trip types. The interquartile range of speed for dangerous trips is around -18 to 51 m/s, which is bigger than the interquartile range of speed for safe trips which is around -18 to 42 m/s. The reason could be because dangerous trips might have taxis going in extreme speeds. As much as its dangerous to drive at extremely high speed it is as equally dangerous to drive too slowly. Hence, the median speed for safe trips which is around 19m/s seem to be slightly higher than that for dangerous trips which is around 18m/s.

Chart, histogram

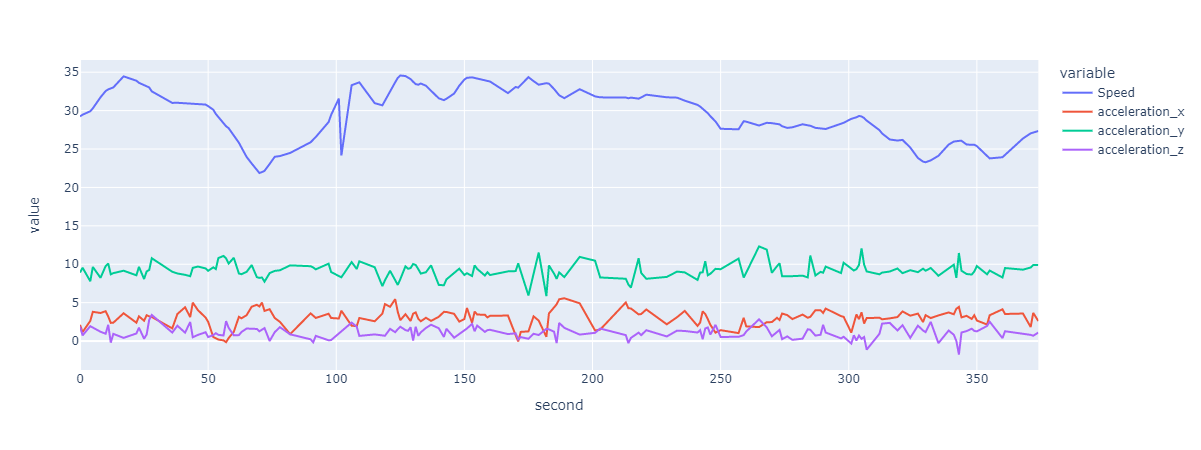
Description automatically generated

### **Histogram**

The histogram shows the distribution of the duration of trips made in this dataset. As you can see, the histogram is skewed to the right. Most of the trips last for around 1000 seconds. There are fewer longer trips that last up to 8000 seconds.

### **Line Chart**

This is a line chart that shows the changes in speed, acceleration\_x, acceleration\_y and acceleration\_z against time. This is only plotted with one bookingID which is derived from finding the median out of all the average speed for all bookingIDs. This line chart shows that there is not much changes in acceleration when the speed is changing. For example, at around 100 seconds of the trip, there is a huge drop in the speed and acceleration in all axis decreased slightly. But in overall, the changes in speed did not have a huge impact in the changes in the acceleration. Acceleration\_y has the biggest range out of all the acceleration. The acceleration in all axes do not change greatly according to each other. For example, at around 200 seconds, there is a decrease in the acceleration\_y and acceleration\_x also decreased.



# **Dashboard**

The dashboard contains 8 interactive charts, and it includes all the columns in the dataset.

### **Packed bubble Chart**

This packed bubble chart visualizes the number of records for each bookingID. There are many bookingIDs in the data as seen from the different colors of bubbles. The size of the bubbles show the number of records in each bookingID. All of the bookingIDs have similar number of records as the sizes of the bubbles are similar to each other and there is no bubble that is significantly bigger than others.

### Chart Description automatically generated **Bar Plot**

This bar plot shows the number of records for each bookingID. The bars in blue are bookingIDs that are normal trips and the bars in orange are the bookingIDs that are dangerous trips. The higher the bar is, the more records the bookingID has. Number of records of a bookingID that is a safe trip have at most 1069 records while bookingIDs that are dangerous trips have at a maximum of 3267 records. BookingIDs that are dangerous trips have significantly more records than bookingIDs that are normal trips.

Chart, bar chart

Description automatically generatedThe graph below shows the maximum speed for each bookingID. Most bookingIDs have a maximum speed of below 60m/s. The fastest and slowest bookingID is a safe trip. There is no pattern or relationship between speed and the type of trip. Thus, speed is not a determining factor for the type of trip.

### **Scatter Plot**

Chart, scatter chart

Description automatically generatedThere are 3 scatterplots, and each scatterplot plots the each axis of acceleration against each other. The graphs below plot all the data points and a blue marker is a record of a normal trip and a orange marker is a record of a dangerous trip.

Chart, scatter chart

Description automatically generatedThis scatterplot plots acceleration\_x against acceleration\_y. We can see that there is no relationship between acceleration\_x and acceleration\_y as the data points do not show that acceleration\_x is increasing/decreasing while acceleration\_y is increasing/decreasing. Records that are in dangerous trips are appearing outside of the records that are in normal trips thus we can infer that there is a boundary for acceleration\_x and acceleration\_y for the trips to be considered dangerous.

Chart, scatter chart

Description automatically generatedThis scatterplot plots acceleration\_y against acceleration\_z. The data points do not form a diagonal line. Instead, the data points form a horizontal line which means there is no relationship between these 2 accelerations. Records that are in safe trips have a lesser range of -13m/s2 to 13m/s2 and the records that are in dangerous trips have a bigger range of -15m/s2 to 15m/s2 for acceleration\_z. There is no difference in the range for acceleration\_y thus we can conclude that acceleration\_z plays a bigger role as compared to acceleration\_y.

This scatterplot plots acceleration\_x against acceleration\_z. The data points do not form a diagonal line. Instead, the data points form a horizontal line which means there is no relationship between these 2 accelerations. Records in the safe trip have the same range, -10m/s2 to 10m/s2, as records in the dangerous trips for acceleration\_x. However, there is a difference in range for the 2 types of trips for acceleration\_z. The range for safe trips is from -12m/s2 to 13m/s2 while the range for dangerous trips is from -15m/s2 to 15m/s2. There is a smaller range for safe trips as compared to dangerous trips thus we can conclude that acceleration\_z plays a bigger role as compared to acceleration\_x.

The overall conclusion from these scatterplots of acceleration is that acceleration\_z has a more obvious boundary for the 2 types of trips, thus acceleration\_z plays a more important role for deciding whether a trip is normal or dangerous.

### Chart, histogram Description automatically generated **Histogram**

This histogram shows the distribution of accuracy for both types of trips. The left histogram is for normal trips and the right histogram is for dangerous trips. Both histograms are positively skewed. There is not much difference in the distribution of accuracy for both types of trips. Most accuracies reside between values of 0 to 50. However, there are extreme values for both trips that are more than 250.

### **Line Chart**

Chart

Description automatically generatedThis line chart shows the change in gyro for all axis for bookingID 0, which is a safe trip. We can see that the gyro for all axis fluctuates a lot and there is a period from around 300 seconds to 900 seconds when the change in gyro is stagnant. The change in an axis of gyro does not really affect the values of another axis. At around 200 seconds, there is a drop in gyro\_y but gyro\_z increased. At around 900 seconds, there is an increase in gyro\_z and gyro\_y and gyro\_x increased. At 1300 seconds, there is a drop in gyro\_x and gyro\_y also decreased. Overall, there is not much relationship to infer from these 3 axes.

# **References**

[1] <https://iopscience.iop.org/article/10.1088/1757-899X/100/1/012017/pdf>

[2] <https://www.researchgate.net/figure/Illustration-of-the-acceleration-on-the-vehicles-z-axis-and-the-corresponding_fig2_269215022>