GPS TRAJECTORY

ABSTRACT:

Aim of this project is to statistically analyse the dataset of GPS Trajectory. In this project I predicted rating of traffic using features like average speed, time, distance .Here we used go_track dataset.

Process:

1.Loading dataset

2.Null values check

3.duplicates checking

4.normality checking

5.outliers checking

6.Apply ordinal regression model

7. Feauture selection

8.Test of assumptions

Dataset description:

```
Attribute Information:

(1) go_track_tracks.csv: a list of trajectories
id_android - it represents the device used to capture the instance;
speed - it represents the average speed (Km/H)
distance - it represent the total distance (Km)
rating - it is an evaluation parameter. Evaluation the traffic is a way
to verify the volunteers perception about the traffic during the
travel, in other words,
if volunteers move to some place and face traffic jam, maybe they will
evaluate 'bad'. (3- good, 2- normal, 1-bad).
rating_bus - it is other evaluation parameter. (1 - The amount of
people inside the bus is little, 2 - The bus is not crowded, 3- The bus
is crowded.
rating_weather - it is another evaluation parameter. ( 2- sunny, 1-
raining).
car_or_bus - (1 - car, 2-bus)
linha - information about the bus that does the pathway


(2) go_track_trackspoints.csv: localization points of each trajectory
id: unique key to identify each point
latitude: latitude from where the point is
longitude: longitude from where the point is
track_id: identify the trajectory which the point belong
time: datetime when the point was collected (GMT-3)
```

ANALYSIS:

Dataset contains 2 files(go_track__tracks.csv,go_track_trackpoints.csv).Second file contains infromtaion about trackid,latitude,longitude.Using second file, speed,distance,time features of first file are calculated.

SAMPLE DATA:

:

	id	id_android	speed	time	distance	rating	rating_bus	rating_weather	car_or_bus	linha
0	1	0	19.210586	0.138049	2.652	3	0	0	1	NaN
1	2	0	30.848229	0.171485	5.290	3	0	0	1	NaN
2	3	1	13.560101	0.067699	0.918	3	0	0	2	NaN
3	4	1	19.766679	0.389544	7.700	3	0	0	2	NaN
4	8	0	25.807401	0.154801	3.995	2	0	0	1	NaN

NULL VALUES

Linha column has 80% null values.So removed the linha column.

Summary of data:

	id	id_android	speed	time	distance	rating	rating_bus	rating_weather	car_or_bus
count	163.000000	163.000000	163.000000	163.000000	163.000000	163.000000	163.000000	163.000000	163.000000
mean	15607.650307	7.386503	16.704738	0.264272	5.302411	2.515337	0.386503	0.515337	1.466258
std	18644.257138	7.348742	16.016168	0.292731	7.639011	0.679105	0.687859	0.841485	0.500397
min	1.000000	0.000000	0.009779	0.002175	0.001000	1.000000	0.000000	0.000000	1.000000
25%	48.500000	1.000000	1.591016	0.035978	0.034500	2.000000	0.000000	0.000000	1.000000
50%	158.000000	4.000000	16.685368	0.214466	3.995000	3.000000	0.000000	0.000000	1.000000
75%	37991.000000	10.000000	23.915760	0.390572	7.333000	3.000000	1.000000	1.000000	2.000000
max	38092.000000	27.000000	96.206029	1.942948	55.770000	3.000000	3.000000	2.000000	2.000000

Speed,distance,time:Continuous variables

Rating_bus,rating:ordinal

Rating_weather,car_or_bus:categorical

So,categorical is converted using get dummies of pandas.

So, dataset now becomes:

speed	time	distance	rating	rating_bus	rating_weather_0	rating_weather_1	rating_weather_2	car_or_bus_1	car_or_bus_
9.210586	0.138049	2.652	3	0	1	0	0	1	
0.848229	0.171485	5.290	3	0	1	0	0	1	
3.560101	0.067699	0.918	3	0	1	0	0	0	
9.766679	0.389544	7.700	3	0	1	0	0	0	
5.807401	0.154801	3.995	2	0	1	0	0	1	

Indpendantvariables:speed,distance,time,rating_bus,rating_weather_0,rating_weather_1,rating_weather_2,car_or_bus_1,car_or_bus_2

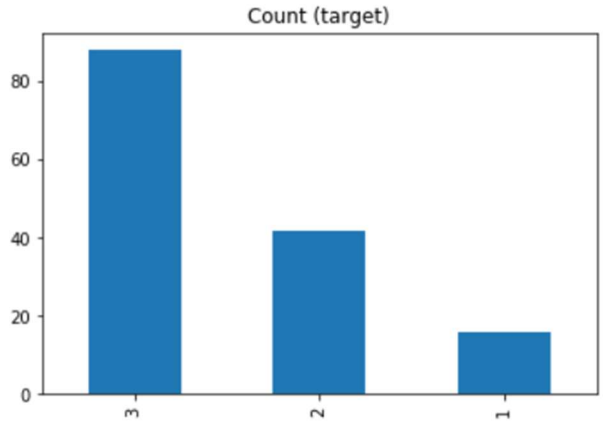
Dependant variables:rating

Drop duplicates:

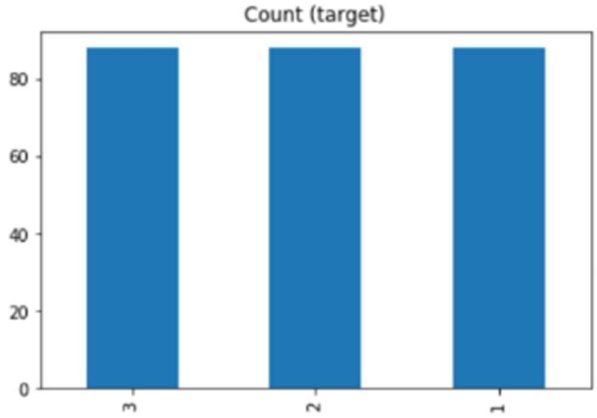
Dataset contains duplicates .This was removed using pandas.drop_duplicates() function.

Imbalanced dataset:

The target variables is imbalanced,That is less number of rating1 compared to other 2 as shown

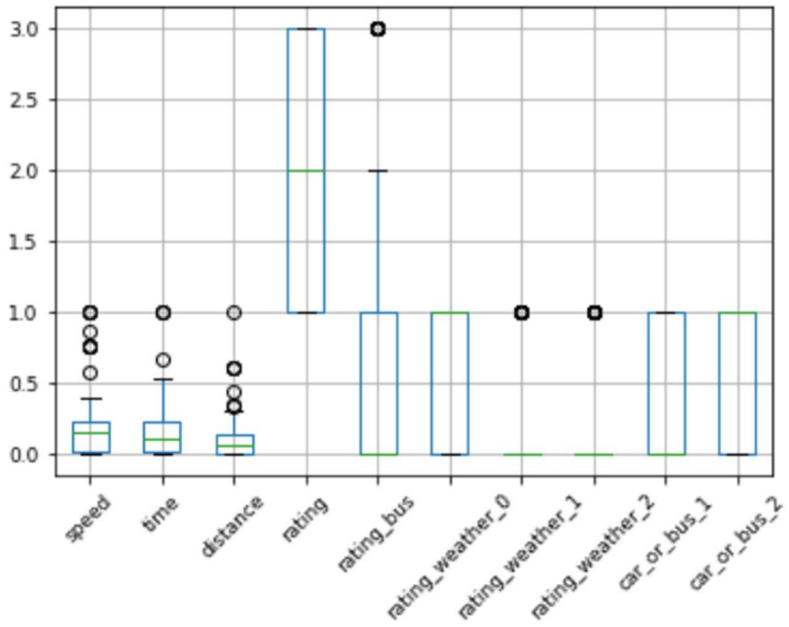


This problem is solved by oversampling of dataset using imbalanced library of python

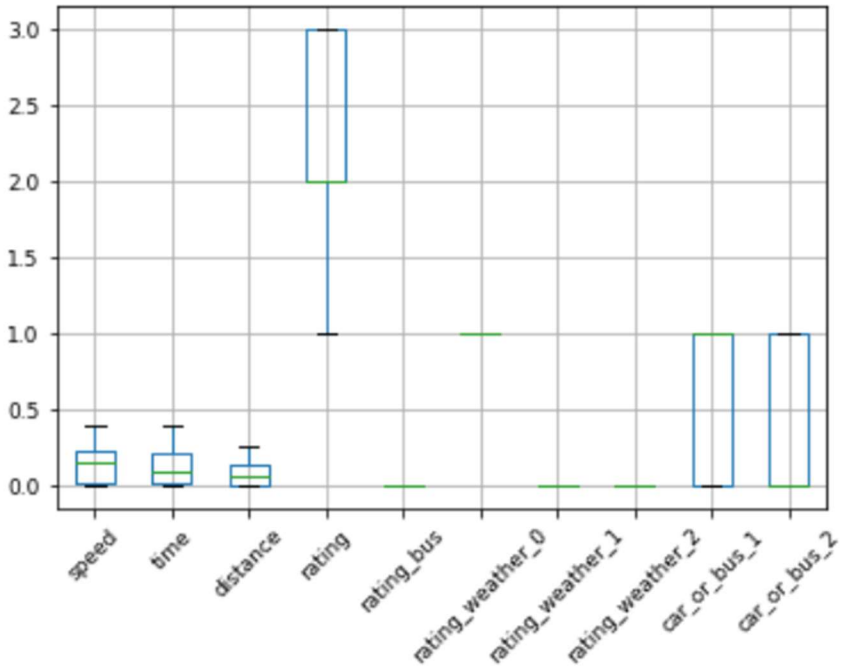


Outliers detection:

By following boxplot we can observer outliers

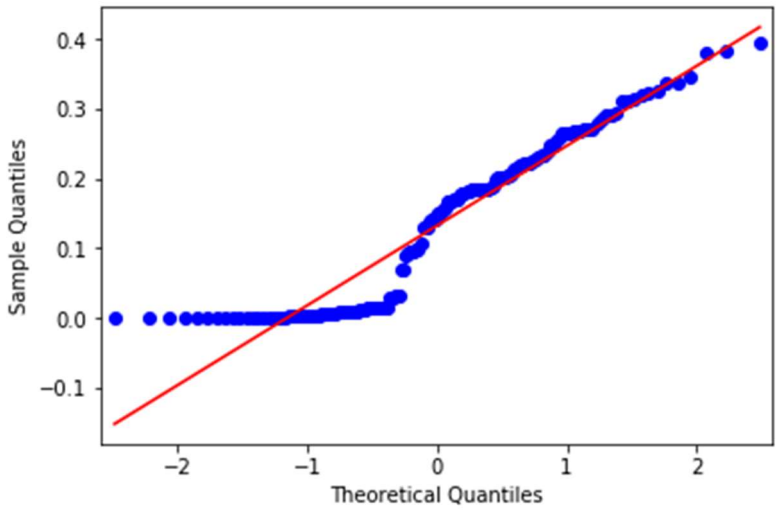


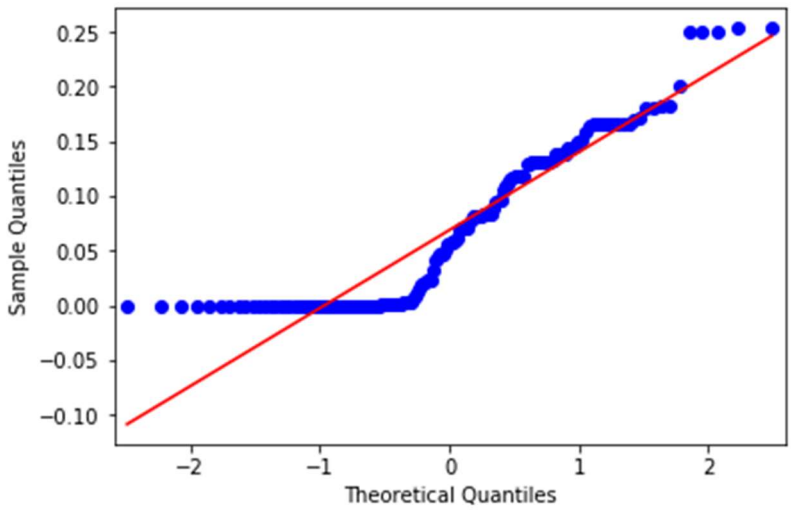
The outliers are removed by finding IQR range and removing points that are not in the IQR range



Normality of data:

Q-Q Plots for speed,time,distance respectively





From above graphs we can see that these are not linear.But ordinal regression doesn't require data to be normally distributed.

APPLYING ORDINAL MODEL:

Since rating is ordinal ,ordinal regression .This is done using mord ligisticat function in python.

Results are as follows:

Goodness of fit :measured in terms of accuracy -0.41935483870967744

Coefficients:

[2.13960580e+00 -1.02190818e+00 -1.37202194e-02 0.00000000e+00

 -1.43958306e-06 0.00000000e+00 0.00000000e+00 4.81319826e-01

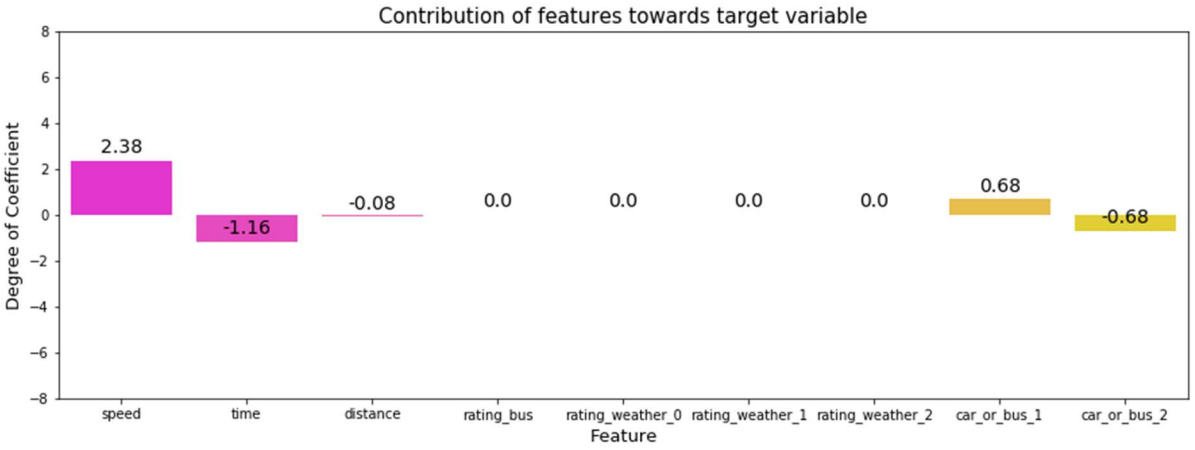
 -4.81321265e-01]

Confusion matrix:

[[0 0 0]

 [6 12 12]

[0 0 1]]



Feature selection:

From the above results,we see the coefficients of rating_bus,rating_weather_1,rating_weather_2 are zeros.

Used inbuit recursive feature elimination from sklearn which gives the results if the features are to selected or not.Results are as follows:

- Selected features:[True True True False False False False True]

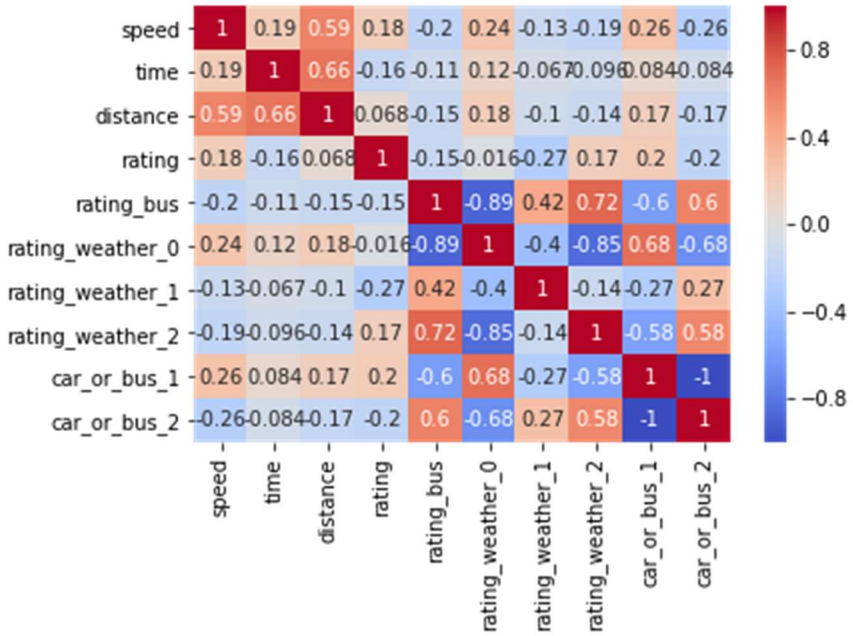
Order is
speed,time,distance,rating_bus,rating_weather_0,rating_weather_1,rating_wrather_2,car_or_bus_1,car_or_bus_2.

So from above results we can remove
rating_bus,rating_weather_0,rating_weather_1,rating_wrather_2

Test of assumptions:

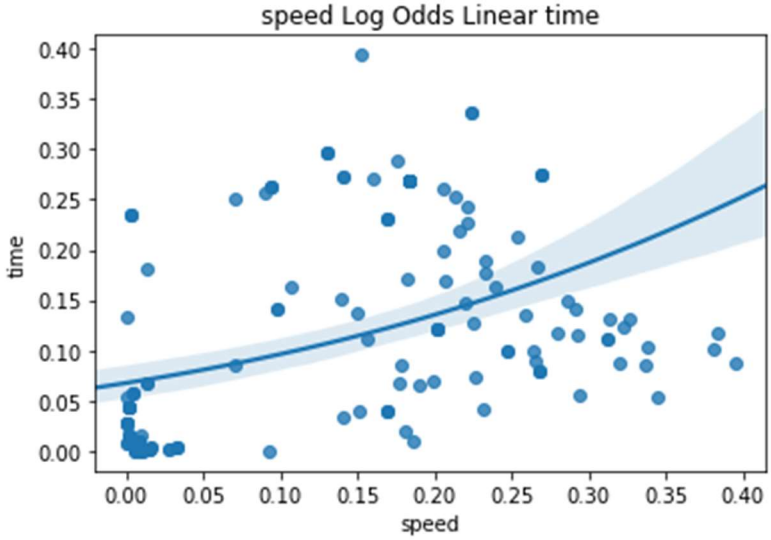
Note:logistic regression does not require a linear relationship between the dependent and independent variables. Second, the error terms (residuals) do not need to be normally distributed. Third, homoscedasticity is not required.

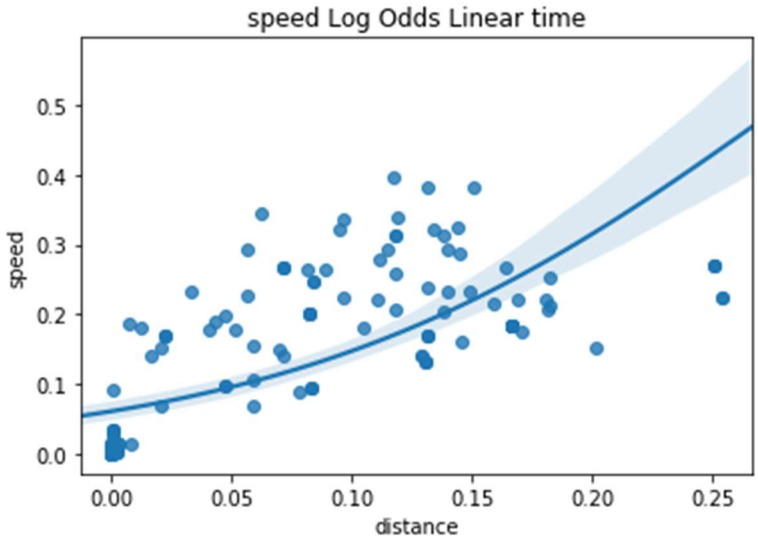
1. ordinal logistic regression requires the dependent variable to be ordinal.
   This is satisfied since rating is ordinal data.
2. It requires there to be little or no multicollinearity among the independent variables. This means that the independent variables should not be too highly correlated with each other.
   Correlation plot:

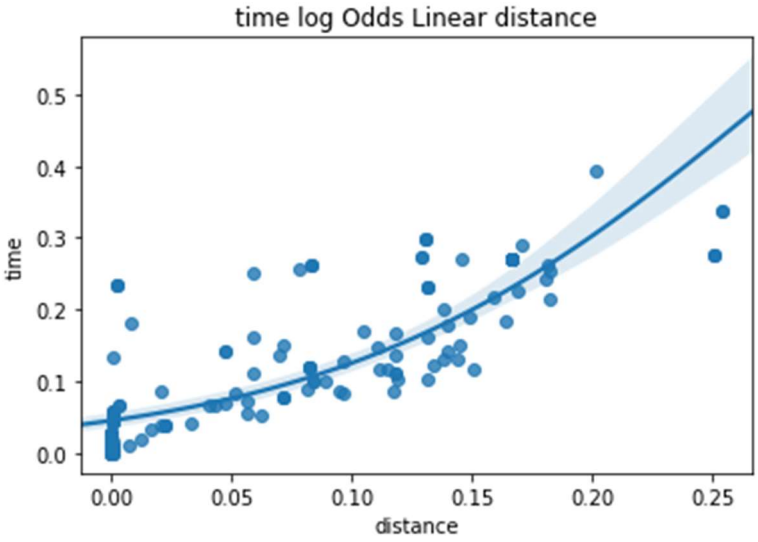


From the above correlation,we can see that car_or_bus_1,car_or_bus_2 are highly correlated.So remove car_or_bus_1.

3. It assumes linearity ofcontinuous independent variables and log odds. although this analysis does not require the dependent and independent variables to be related linearly, it requires that the independent variables are linearly related to the log odds. That is by plotting loglinear graphs if we observe some curve(S shape) this satisfies the assumption.







So,from above graphs we can see that logodd test is satisfied.So, we can say that after test of assumptions and feature selection we can remove rating_bus,rating_weather_1,rating_weather_2 ,car_or_bus_1.So,again applying ordinal regression, these are results:

Accuracuy:0.41935483870967744

Coefficients:[2.16441633e+00 -9.99903774e-01 7.18855331e-04 -1.79099231e-06

-9.02924895e-01]

Confusion matrix:[[0 0 0]

[6 12 12]

[0 0 1]]

We can see that even after doing test of assumptions we found that accuracy is very less. So applied different models like LogisticRegression CV,Random Forest classifier.

Results using logistic regression CV:

Accuracy:0.6580645161290323

Coefficients:[[-5.52822551e+00 3.50113465e+00 3.17647802e+00 0.00000000e+00
6.16715464e-04 0.00000000e+00 0.00000000e+00 -5.08959471e-01
5.09576187e-01]
[-2.09515539e+00 9.53578685e-02 -1.04287583e+00 0.00000000e+00
-3.86861143e-03 0.00000000e+00 0.00000000e+00 -2.26950331e-02
1.88264216e-02]
[7.62338090e+00 -3.59649252e+00 -2.13360218e+00 0.00000000e+00
3.25189597e-03 0.00000000e+00 0.00000000e+00 5.31654504e-01
-5.28402608e-01]]

Intecepts:

[-0.59932287 1.14069747 -0.54137459]

Again performing feature selection and removing features having coefficient 0.Again applying **logicregressioncv** we get following results.

Accuracy:0.6580645161290323

Cofficients:[[-5.53074622e+00 3.49323103e+00 3.17347507e+00 5.11084927e-03
1.01217584e+00]
[-2.09559423e+00 9.71864392e-02 -1.04234458e+00 -6.62311676e-06
4.17469559e-02]
[7.62634046e+00 -3.59041747e+00 -2.13113049e+00 -5.10422615e-03
-1.05392279e+00]]

Intercepts:[-1.1085018 1.11311259 -0.00461079]

Random forestclassifier:

Using random forest classifier,got the following results:

Coefficients:[0.36447909 0.24805381 0.24221074 0. 0. 0.
0. 0.08956213 0.05569424]

Accuracy:0.8064516129032258

Perfoming feature selection and again applying **randomizedsearchcv,**

got following results:

randomized cv

confusion matrix:[[3 0 0]

[0 19 1]

[0 3 5]]

Accuracy:0.8709677419354839

So, randomizedsearchcv ,We got better results.(87% accuracy)

Summary: From the above results, we can see that randomizedsearchcv performs best for this dataset.

MODEL	ACCURACY
ORDINAL REGRESSION	41%
LOGISTICREGRESSIONCV	65%
RANDOM FOREST	80%
RANDOMIZEDSEARCHCV	87%