Discuss the dataset

Introduction

There are two main types of the dataset provided; train_revised.csv and test_question.csv datasets. The train_revised.csv contains dataset of tickets purchased from Mobiticket for fourteen (14) routes from "up country" into the city of Nairobi. The dataset runs for a period between 17 October 2017 and 20 April 2018. The variables for this dataset are;

a. ride_id – this is a unique identification representing trip made from an up country,
b. seat_number – this stores the seat numbers for a specific car,
c. payment_method – stores type of payment method used e.g. Mpesa or Cash,
d. payment_receipt – store the receipt number as a prove of payment and it's unique,
e. travel_date – stores the commuting date when the ride happened,
f. travel_time – stores commuting time for the ride,
g. travel_from – store the start or originating town for the ride,
h. travel_to – stores the destination of the ride i.e. Nairobi,
i. car_type – stores the type of car used for the ride, Bus or shuttle,
j. max_capacity – stores the maximum number of passengers for each car type.

The train or training data is data for building a model, it's used in a supervised learning as it contains outcomes to train the machine/model.

Test_question.csv dataset is the testing dataset, the outcomes are not known, it depends on the model created from the training (train_revised.csv) dataset to predict its outcomes. In this case it contains most of the variables from the training dataset with an exception of seat_number, payment_method and payment_receipt.

Observations

The training dataset (train_revised.csv), the variable ride_id appears multiple times in the dataset which is different from the testing dataset (test_question.csv).

The training dataset has more columns compared to the testing dataset.

In both dataset there is no variable which stores the total number of tickets sold for each ride from all routes. This build the question and defines what the model should do, predict the number of tickets for each ride.

Assumptions in creating the model

To get the number of tickets for each ride in the training dataset, we count/aggregate how many times the ride_id has been repeated in the dataset. To achieve this, we consolidate the ride by count of the ride_id and store them in a variable no_of_tickets. Then, merge the aggregate to the training dataset and remove the duplicates.

Name the new dataset as train_aggregate.csv, which has an addition variable which answers our question on number_of_tickets sold per ride. The new train dataset should now have the following variables; ride_id, travel_date, travel_time, travel_from, travel_to, car_type, max_capacity, number_of_tickets. The new dataset will be used to create a predictive model to predict the outcomes of the testing dataset.

Conclusions

From the problem description, to build a model to predict the number of seats that Mobitickets can expect to sell for each ride for a specific route on a specific date and time, the aggregated training dataset with already determined or known outcomes will be ideal training dataset to solve this problem.

Model using regression

Regression analysis – is a set of statistical processes for estimating the relationship among variables and understanding which among the independent variables are related to the dependent variables and explore the nature of the identified relationships.

Regression model – it's used to investigate the relationship between two or more variables and use it to estimate one variable based on the others.

Solution

From the aggregated training dataset (*train_aggregate.csv*), identify the independent and dependent variables. The dependent variables in this case will be the number of tickets for each ride, this is because it's dependent to the route on a specific date and time. This makes route as an independent variable.

The summary output between routes against the number of tickets

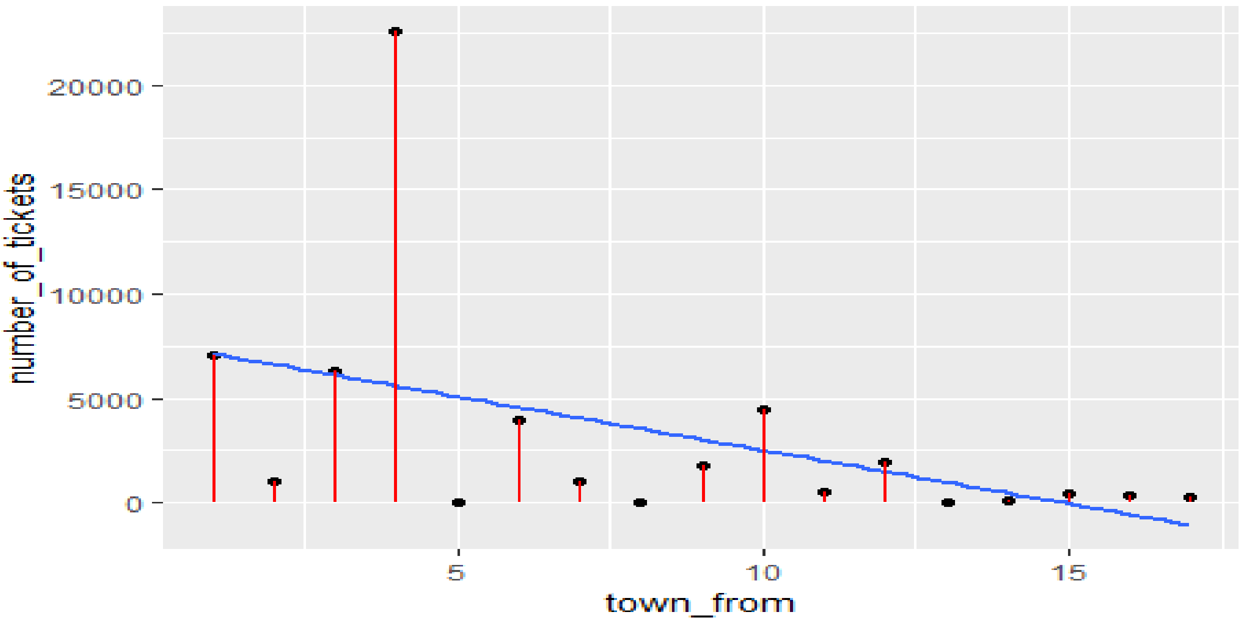
SUMMARY OUTPUT								
Regression Statistics								
Multiple R	0.237961691							
R Square	0.056625766							
Adjusted R Square	-0.005874234							
Standard Error	10.25890774							
Observations	17							
ANOVA								
	df	**SS**	**MS**	**F**	**Significance F**			
Regression	1	101.0769929	101.0769929	0.960395386	0.342629973			
Residual	16	1683.923007	105.2451879					
Total	17	1785						
	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%
Intercept	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
number_of_tickets	0.000395887	0.000403967	0.979997646	0.34167412	-0.000460485	0.001252258	-0.000460485	0.001252258

RESIDUAL OUTPUT			
Observation	Predicted sn(town)	Residuals	Standard Residuals
1	2.781895634	-1.781895634	-0.179038161
2	0.39113603	1.60886397	0.161652591
3	2.495669571	0.504330429	0.050673222
4	8.949809961	-4.949809961	-0.49733826
5	0.008709507	4.991290493	0.501506068
6	1.555042842	4.444957158	0.446612552
7	0.408159157	6.591840843	0.662323338
8	0.001979433	7.998020567	0.80361098
9	0.707845367	8.292154633	0.833164463
10	1.761299797	8.238700203	0.827793563
11	0.202693975	10.79730602	1.084872638
12	0.755747654	11.24425235	1.129780121
13	0.000395887	12.99960411	1.306151255
14	0.021773767	13.97822623	1.404479519
15	0.160334102	14.8396659	1.491033732
16	0.149645161	15.85035484	1.59258395
17	0.093429254	16.90657075	1.698708545

PROBABILITY OUTPUT	
Percentile	sn(route/town)
2.941176471	1
8.823529412	2
14.70588235	3
20.58823529	4
26.47058824	5
32.35294118	6
38.23529412	7
44.11764706	8
50	9
55.88235294	10
61.76470588	11
67.64705882	12
73.52941176	13
79.41176471	14
85.29411765	15
91.17647059	16
97.05882353	17

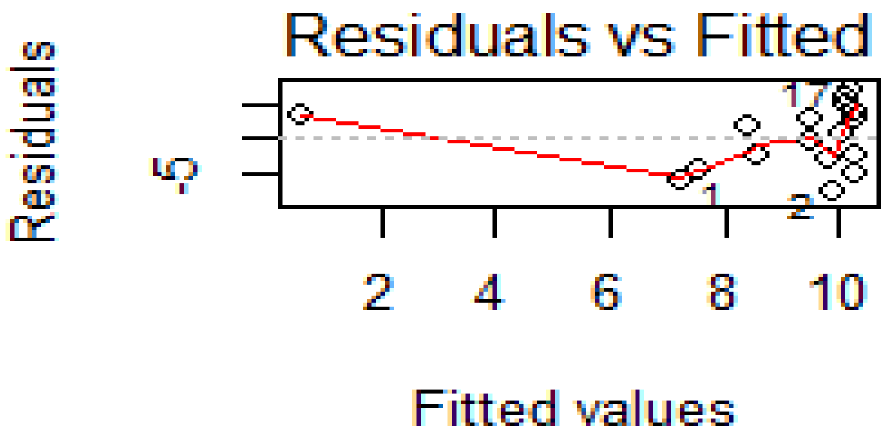
Model using R.

```
Coefficients:

      (Intercept)  number_of_tickets
       10.3040303         -0.0004292

# A tibble: 6 x 9
  town_from number_of_tickets .fitted .se.fit .resid   .hat .sigma .cooksd .std.resid
      <int>             <int>   <dbl>   <dbl>  <dbl>  <dbl>  <dbl>   <dbl>      <dbl>
1         1              7027    7.29    1.39  -6.29 0.0914   4.43  0.103       -1.43
2         2               988    9.88    1.20  -7.88 0.0674   4.24  0.114       -1.77
3         3              6304    7.60    1.31  -4.60 0.0807   4.59  0.0476      -1.04
4         4             22607    0.600   4.23   3.40 0.844    4.18  9.39         1.87
5         5                22   10.3     1.28  -5.29 0.0775   4.53  0.0602      -1.20
6         6              3928    8.62    1.13  -2.62 0.0604   4.71  0.0111      -0.587
```



Create a model and show regression modelling assumptions

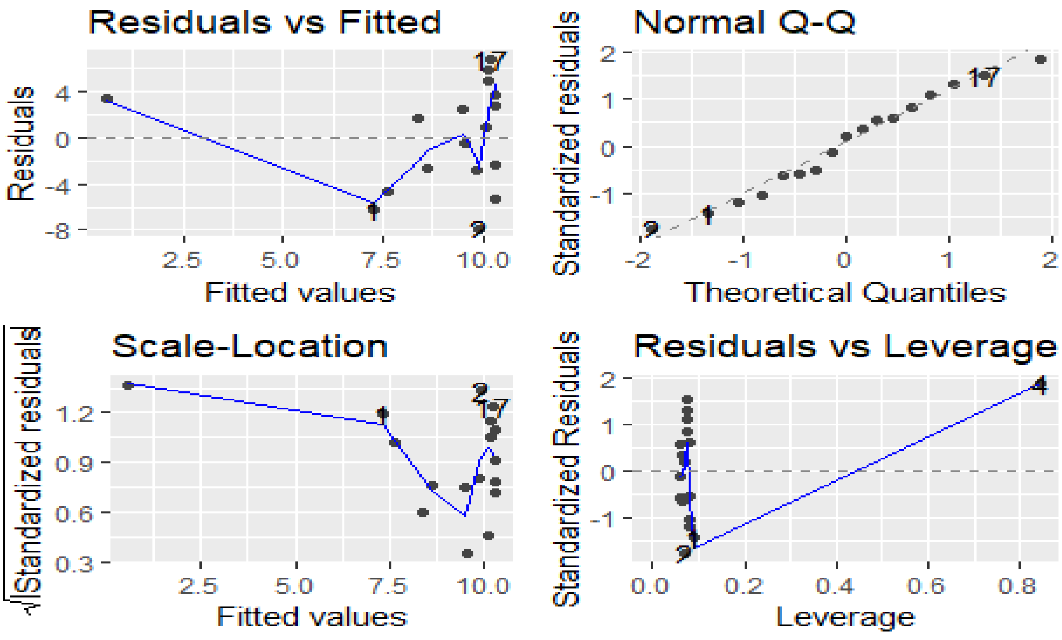
Linearity - The relationship between the predictor (x) and the outcome (y) is assumed to be linear. The linearity assumption can be checked by inspecting the *Residuals vs. fitted* plot (1st plot)



Normality of residuals. The residual errors are assumed to be normally distributed. Constant variance. The QQ plot of residuals can be used to visually check the normality assumption. The normal probability plot of residuals should approximately follow a straight line.

Homogeneity of residuals variance. The residuals are assumed to have a constant variance *(homoscedasticity)*

Independence of residuals error terms



Conclusion

The model can be used to predict the number of tickets in the testing dataset