



Uber Data Analysis project Report

Academic year: Fall 2021-2022

Under the guidance of:
Prof. S.Gopikrishna

Submitted by:

19BCD7006 - KODURI GOKUL
19BCD7014 - KONGARA SAI HAREESH KUMAR
19BCD7019 - MUPPURI SIVAKUMAR
19BCD7025 - BUSSA SAIBIPIN

Content:

Summary:	3
Introduction:.....	4
DATA ANALYST:.....	5
Data Scientist:	
Dataset:	...11
Cleaning the Dataset:	
Code:.....	13
Conclusion:.....	17
Reference:.....	1

Summary:

The working of an Uber dataset includes primary Data on Uber pickups with details including the date, time of the ride as well as longitude-latitude information. Uber data analysis project, data storytelling is an important component of **Machine Learning** through which companies are able to understand the background of various operations. With the help of visualization, companies can avail the benefit of understanding the complex data and gain insights that would help them to craft decisions. The interesting insights that can be derived from a detailed analysis of the dataset. The ggplot2 on the Uber Pickups dataset and at the end, master the art of data visualization in R. the dataset and to know the effect of each field on price with every other field of the dataset. Then The objective is to first explore hidden or previously unknown information by applying exploratory data analytic on y different R models to complete the analysis. The number of pickups is more during weekends. To alleviate the dynamic price surge, we need to manage the 'supply and demand' of cabs through these events of high demand situations. Based on these results, we can expect that the demand will be high as the temperature drops or after business hours or on weekends, so that: As a Customer: We can plan our trips in advance to avoid paying extra money because of this dynamic price surge. As an Uber Driver: We can maximize profits by choosing to go on trips when these situations occur.

Introduction:

Uber launched in NYC in May of 2011, the first city outside of its San Francisco headquarters. NYC is probably the largest and most lucrative rideshare market in the world, with a total demand (for taxis and for-hire vehicles) in 2017 of more than 240 million trips per year. The number of Uber trips per day in NYC is still growing significantly. In 2017 so far, this number has often surpassed 200,000, but the plot below shows that by mid-2015 it was hovering around 120,000. The data also allows us to visualize other interesting trends over time. In the bar charts below, we can see that the demand for Uber is higher. Saturday has the highest demand. Interestingly, Sunday shows a level of demand similar to Wednesday, which is higher than Monday or Tuesday. When looking at the total demand per month along the period of time analyzed, seasonal effects are masked by the consistent month-to-month growth.

DATA ANALYST:

```
library(readr)
uber <- read_csv("C:/Users/siva kumar/Downloads/uber-rides-dataset- updated.csv")

## Rows: 678 Columns: 37

## Column specification
--
## Delimiter: ","
## chr (20): trip_status, trip_uid, driver_uid, rider_uid, customer, trip_star...
## dbl (12): pickup_lat, pickup_long, dropoff_lat, dropoff_long,
rub_usd_excha...
## dtm (2): trip_completed_at, temperature_time ##
time (3): trip_time, total_time, wait_time

##
## i Use `spec()` to retrieve the full column specification for this data. ## i Specify
the column types or set `show_col_types = FALSE` to quiet this message.

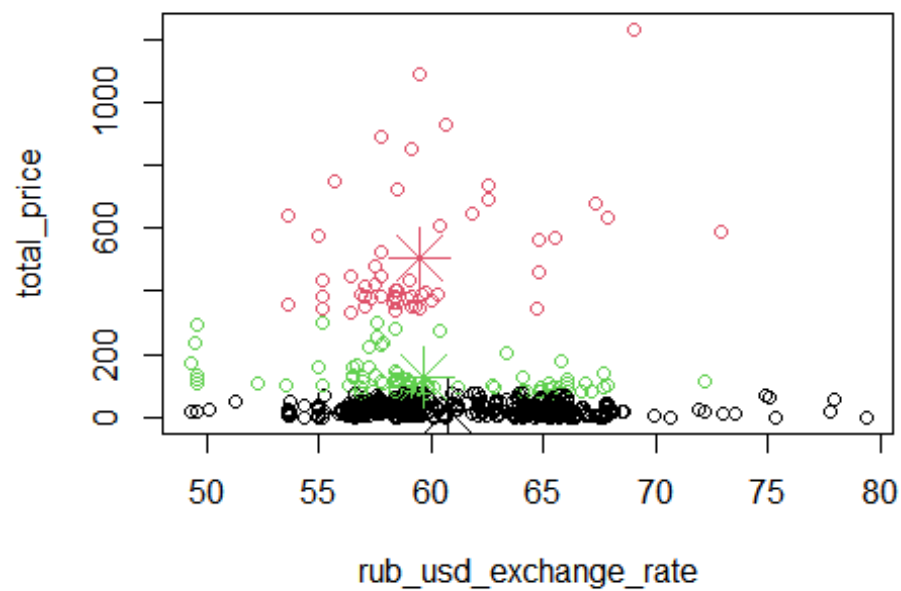
df <- uber[,c(7:19,22,24:37)]

# clustering
library(gtools)
library(ClusterR)
library(cluster)
library(ggplot2)

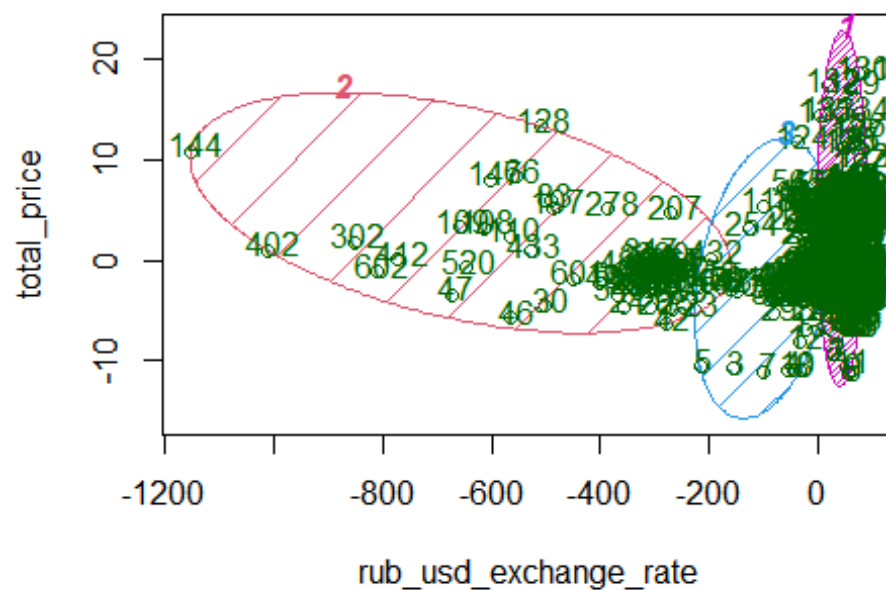
ClusterFunction <- function(clusterDF){

  kmeans.re <- kmeans(clusterDF, centers = 3)
  plot(clusterDF[,c(1,2)], col = kmeans.re$cluster)
  points(kmeans.re$centers, col = 1:3, pch = 8, cex = 3)
  y_kmeans <- kmeans.re$cluster
  clusplot(clusterDF[,c(1,2)],
            y_kmeans, lines
            = 0, shade =
            TRUE, color =
            TRUE, labels =
            2,
```

```
    plotchar = FALSE,  
    span = TRUE,  
    main = paste("Cluster iris"), xlab =  
    'rub_usd_exchange_rate', ylab =  
    'total_price')  
}  
  
# clustering Prize  
ClusterFunction(df[,c(17,18)])
```



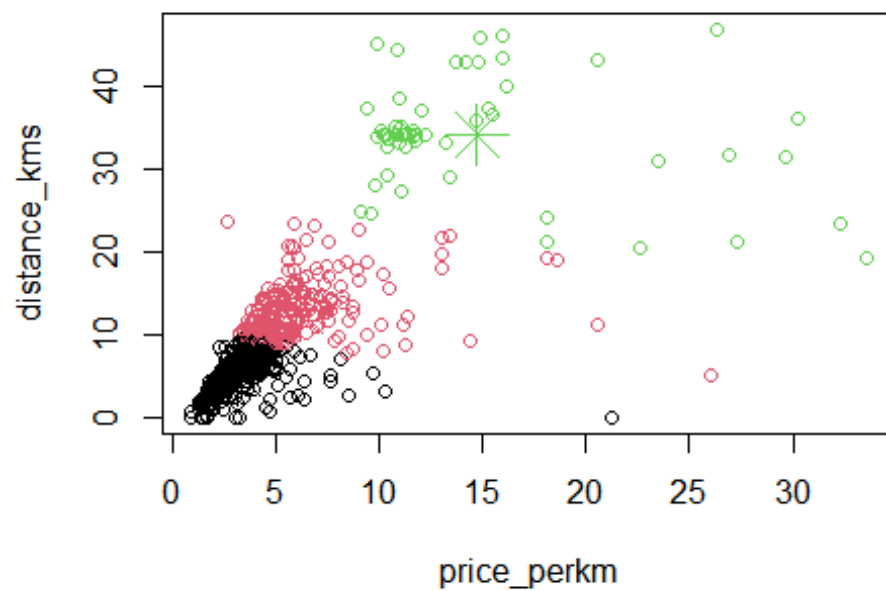
Cluster iris



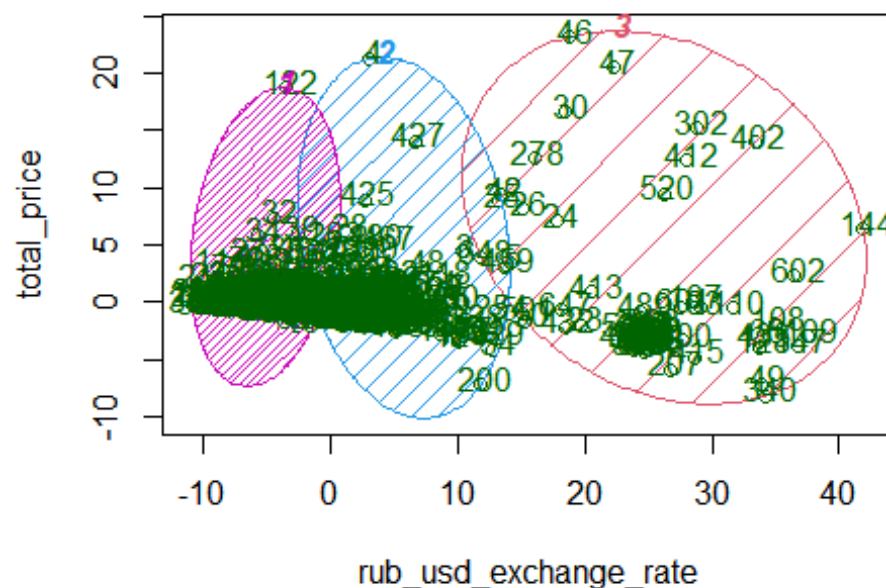
These two components explain 100 % of the point variability

```
# Clustering Km
ClusterFunction(df[,c(19,20)])
```

```
ClusterFunction(df[,c(19,20)])
```



Cluster iris



These two components explain 100 % of the point variability

```
# Hierarchical clustering
Hcluster <- df[,c(22,24:26)]

distance_mat <- dist(Hcluster, method = 'euclidean')
```



```
## Warning in dist(Hcluster, method = "euclidean"): NAs introduced by coercion
```

```
Hierar_cl <- hclust(distance_mat, method = "average")  
Hierar_cl
```

```
##
```

```
## Call:
```

```
## hclust(d = distance_mat, method = "average") ##
```

```
## Cluster method : average ##
```

```
Distance : euclidean ## Number
```

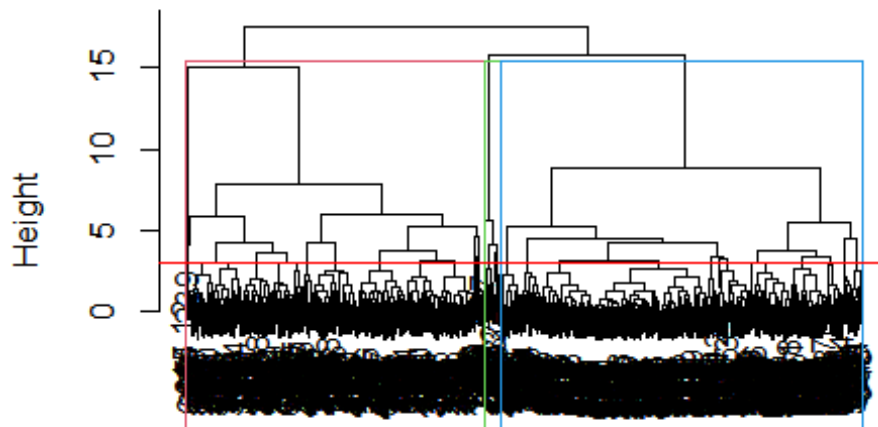
```
of objects: 678
```

```
cut_avg <- cutree(Hierar_cl, k = 3)
```

```
plot(Hierar_cl)
```

```
rect.hclust(Hierar_cl , k = 3, border = 2:6) abline(h = 3,  
col = 'red')
```

Cluster Dendrogram

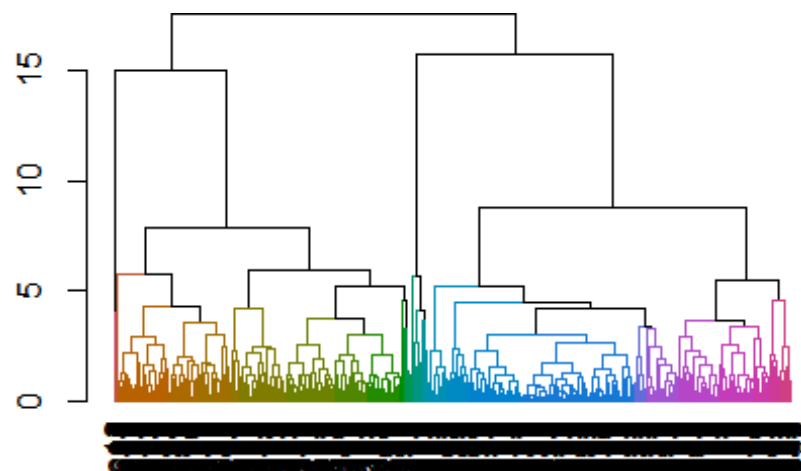


distance_mat
hclust (*, "average")

```
suppressPackageStartupMessages(library(dendextend))
```

```
## Warning: package 'dendextend' was built under R version 4.1.2 avg_dend_obj <-  
as.dendrogram(Hierar_cl)
```

```
avg_col_dend <- color_branches(avg_dend_obj, h = 3) plot(avg_col_dend)
```



Data Scientist:

Dataset:

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	trip_comple	trip_status	trip_uid	driver_uid	ridr_uid	customer	trip_start_t	trip_end_t	trip_time	total_time	wait_time	vehicle_mal	driver_nam	vehicle_mal	driver_gend	pickup_lat	pickup_long	dropoff_lat	dropoff_lon	trip_map	in_trip_f
2	#####	Completed	ee89076d9	05feb269e	3ffa4a71a5;	stantyan	#####	#####	0:21:33	0:29:00	0:07:27	Ford Focus	Maksim	Ford	Male	60.031438	30.329826	59.963131	30.307655	[ANONYMIZ]	ANON
3	#####	Completed	518be51d4c	4a4e24874;	3ffa4a71a5;	stantyan	#####	#####	0:19:27	0:26:00	0:06:33	Hyundai Sol	Sergey	Hyundai	Male	59.963014	30.307313	60.031351	30.329495	[ANONYMIZ]	ANON
4	#####	Completed	6e460cc8a1	cb249a2bd8;	3ffa4a71a5;	stantyan	13-05-2015	13-05-2015	1:06:53	1:23:00	0:16:07	Renault Flux	Oleg	Renault	Male	60.031529	30.329416	59.924281	30.387561	[ANONYMIZ]	ANON
5	#####	Completed	49613a86ac	d3f73f8151;	3ffa4a71a5;	stantyan	16-05-2015	16-05-2015	0:13:37	0:20:00	0:06:23	Mercedes-B	Maksim	Mercedes-B	Male	59.959883	30.311159	59.93468	30.308489	[ANONYMIZ]	ANON
6	#####	Completed	9896148de6	1287d21e6;	3ffa4a71a5;	stantyan	16-05-2015	16-05-2015	0:38:54	0:49:00	0:10:06	Hyundai Sol	Eduard	Hyundai	Male	59.934813	30.308553	60.03147	30.329402	[ANONYMIZ]	ANON
7	#####	Completed	5c0312a92f	1c6b151637	3ffa4a71a5;	stantyan	18-05-2015	18-05-2015	0:16:38	0:34:00	0:17:22	Hyundai Sol	Andrey	Hyundai	Male	59.925603	30.321773	59.928813	30.388147	[ANONYMIZ]	ANON
8	#####	Completed	4ad2e90581	1b926e88a;	3ffa4a71a5;	stantyan	18-05-2015	18-05-2015	0:40:24	0:44:00	0:03:36	Volkswagen	Igor	Volkswagen	Male	59.9287	30.387829	60.031336	30.329518	[ANONYMIZ]	ANON
9	#####	Completed	1e3935b05e	439ae2cf8a	3ffa4a71a5;	stantyan	19-05-2015	19-05-2015	0:41:56	0:58:00	0:16:04	Hyundai sol	Muhammed	Hyundai	Male	60.031592	30.330248	59.944279	30.359076	[ANONYMIZ]	ANON
10	#####	Completed	0eb9a9f7a3	75a4c47c3;	3ffa4a71a5;	stantyan	19-05-2015	19-05-2015	0:10:06	0:15:00	0:04:54	Hyundai ix3	Vladimir	Hyundai	Male	59.945143	30.356079	59.929122	30.388656	[ANONYMIZ]	ANON
11	#####	Completed	b56495d1a1	276f50c424	3ffa4a71a5;	stantyan	19-05-2015	19-05-2015	0:25:30	0:33:00	0:07:30	Volkswagen	Roman	Volkswagen	Male	59.92865	30.388166	60.031366	30.329493	[ANONYMIZ]	ANON
12	#####	Completed	613f3deb51	e51623399;	3ffa4a71a5;	stantyan	20-05-2015	20-05-2015	0:17:47	0:26:00	0:08:13	Hyundai Sol	Aleksey	Hyundai	Male	59.925556	30.321134	59.929301	30.388775	[ANONYMIZ]	ANON
13	#####	Completed	0d486aced5	e819523458;	3ffa4a71a5;	stantyan	31-05-2015	31-05-2015	0:14:28	0:27:00	0:12:32	Hyundai Sol	Ekaterina	Hyundai	Female	56.752253	60.805972	56.794567	60.614053	[ANONYMIZ]	ANON
14	#####	Completed	d12da2c7ae	f065223fa8	3ffa4a71a5;	stantyan	#####	#####	0:15:19	0:28:00	0:12:41	Ford Monde	Oleg	Ford	Male	56.751753	60.803867	56.796019	60.614159	[ANONYMIZ]	ANON
15	#####	Completed	36695e908f	b897afbe68	3ffa4a71a5;	stantyan	#####	#####	0:50:13	1:04:00	0:13:47	Ford Focus	Aleksandr	Ford	Male	56.795387	60.612997	56.857545	60.60663	[ANONYMIZ]	ANON
16	#####	Completed	30bd4a26ce	e733153fbc	3ffa4a71a5;	stantyan	#####	#####	0:12:39	0:37:00	0:24:21	Hyundai Sol	Sergey	Hyundai	Male	56.857754	60.606077	56.857726	60.606094	[ANONYMIZ]	ANON
17	#####	Completed	3e9559661f	85084dd09;	3ffa4a71a5;	stantyan	#####	#####	0:16:18	0:34:00	0:17:42	Kia Rio	Aleksey	Kia	Male	56.795207	60.612907	56.857433	60.606187	[ANONYMIZ]	ANON
18	#####	Completed	4669e3d96f	85086781e	3ffa4a71a5;	stantyan	#####	#####	0:11:27	0:25:00	0:13:33	Nissan Almr	Andrey	Nissan	Male	56.795483	60.612903	56.8283	60.597542	[ANONYMIZ]	ANON
19	#####	Completed	7e321acbd	e050ef8cb3	3ffa4a71a5;	stantyan	#####	#####	0:16:39	0:30:00	0:13:21	Kia Rio	Ekaterina	Kia	Female	56.795296	60.612885	56.817396	60.634999	[ANONYMIZ]	ANON
20	#####	Completed	8e9e903d95	76e28e41c5	3ffa4a71a5;	stantyan	#####	#####	0:16:28	0:27:00	0:10:32	Hyundai Sol	Aleksandr	Hyundai	Male	56.795367	60.614292	56.750043	60.801553	[ANONYMIZ]	ANON
21	#####	Completed	e01e7067e3	767a9c87d;	3ffa4a71a5;	stantyan	#####	#####	0:14:13	0:17:00	0:02:47	Hyundai Sol	Sergey	Hyundai	Male	59.925502	30.339049	59.948717	30.303298	[ANONYMIZ]	ANON
22	#####	Completed	c9d3a9edbf	151944a8f9	3ffa4a71a5;	stantyan	#####	#####	0:21:12	0:36:00	0:14:48	Chevrolet C	Oleg	Chevrolet	Male	59.951232	30.310537	60.011847	30.434048	[ANONYMIZ]	ANON
23	#####	Cancelled	b970e99cf7	e89bc69e8;	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:01:00	0:46:00	0:45:00	Mercedes-B	Pavel	Mercedes-B	Male	59.799903	30.273215	59.799903	30.273236	[ANONYMIZ]	ANON
24	#####	Completed	945269c95c	25f40db71;	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:45:40	0:54:00	0:28:22	Kia Cee'd	Maksim	Kia	Male	59.929035	30.356715	60.056114	30.427891	[ANONYMIZ]	ANON
25	#####	Completed	04f14fe72a	6b3642ae75	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:30:44	0:40:00	0:09:16	Mitsubishi A	Sergey	Mitsubishi	Male	60.016182	30.409457	60.02793	60.634919	[ANONYMIZ]	ANON
26	#####	Completed	fbd229175	69f4e9a875	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:25:18	0:30:00	0:04:42	Nissan Almr	Artem	Nissan	Male	59.927837	30.337983	59.800195	30.274495	[ANONYMIZ]	ANON
27	#####	Completed	4d984dfe40	e1992ce90;	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:30:18	0:37:00	0:06:42	Honda Civic	Finat	Honda	Male	59.927697	30.338072	59.927216	30.33929	[ANONYMIZ]	ANON
28	#####	Completed	57b0d2881f	b2cadbd6ee	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:09:33	0:19:00	0:09:27	Mercedes-B	Gennadiy	Mercedes-B	Male	59.927608	30.338298	59.93286	30.345696	[ANONYMIZ]	ANON
29	#####	Completed	268145a88e	e45d6a746;	3ffa4a71a5;	stantyan	14-06-2015	14-06-2015	0:22:31	0:29:00	0:06:29	Audi A7	Ramil	Audi	Male	59.932907	30.345892	59.927743	30.337948	[ANONYMIZ]	ANON
30	#####	Completed	9622fd4e18	78aab4c1e;	3ffa4a71a5;	stantyan	15-06-2015	15-06-2015	0:45:00	0:57:00	0:12:00	Mitsubishi C	Evgeniy	Mitsubishi	Male	60.01593	30.407802	59.927279	30.335885	[ANONYMIZ]	ANON
31	#####	Completed	72801dd19;	3613586d8;	3ffa4a71a5;	stantyan	15-06-2015	15-06-2015	1:08:04	1:28:00	0:19:56	Kia Rio	Kryuchkov	Kia	Male	59.799339	30.274062	59.927757	30.337976	[ANONYMIZ]	ANON
32	#####	Completed	f374f0aedd	1e734b1a8;	3ffa4a71a5;	stantyan	15-06-2015	15-06-2015	0:13:28	0:27:00	0:14:32	BMW 5-seri	Andrey	BMW	Male	50.037516	30.338067	50.018768	30.285588	[ANONYMIZ]	ANON

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
41	#####	Completed	cb06523d1e	66c6bd76e5	3ffa4a71a5;	stantyan	16-06-2015	16-06-2015	0:07:30	0:21:00	0:13:30	BMW 5-seri	Yana	BMW	Female	59.943622	30.32664	59.927673	30.33796	[ANONYMIZ]	ANON
42	#####	Completed	94ff538a9d	9d9db840;	3ffa4a71a5;	stantyan	17-06-2015	17-06-2015	0:10:20	0:22:00	0:11:40	Opel Astra	Tatyana	Opel	Female	59.918799	30.285494	59.927643	30.338161	[ANONYMIZ]	ANON
43	#####	Completed	9e4c25f004	1ca0a7c5fb	3ffa4a71a5;	stantyan	17-06-2015	17-06-2015	0:31:09	0:38:00	0:06:51	Ford Focus	Vadim	Ford	Male	59.927612	30.338089	59.800281	30.274314	[ANONYMIZ]	ANON
44	#####	Completed	ad25b0145;	c7a9f78db6	3ffa4a71a5;	stantyan	17-06-2015	17-06-2015	0:28:17	0:30:00	0:01:43	Kia Optima	Nikolay	Kia	Male	59.92891	30.335499	59.914004	30.341825	[ANONYMIZ]	ANON
45	#####	Completed	03210d1c2e	6410050a3;	3ffa4a71a5;	stantyan	17-06-2015	17-06-2015	0:29:50	0:45:00	0:15:10	Audi Q3	Andrey	Audi	Male	59.913822	30.341858	59.927962	30.337645	[ANONYMIZ]	ANON
46	#####	Completed	e912743c9f	fc8157aa69	3ffa4a71a5;	stantyan	17-06-2015	17-06-2015	0:11:39	0:27:00	0:15:21	Citroen C4	Aleksandr	Citroen	Male	59.929053	30.356678	59.928041	30.336991	[ANONYMIZ]	ANON
47	#####	Completed	f981dcaa5c	d3f73f8151;	3ffa4a71a5;	stantyan	17-06-2015	17-06-2015	0:48:13	0:57:00	0:08:47	Mercedes-B	Maksim	Mercedes-B	Male	59.927689	30.337908	59.800026	30.274172	[ANONYMIZ]	ANON
48	#####	Completed	910b27c47f	e7d863211;	3ffa4a71a5;	stantyan	29-06-2015	29-06-2015	0:31:20	0:46:00	0:14:40	Mercedes-B	Aleksandr	Mercedes-B	Male	59.928982	30.356505	59.800015	30.27378	[ANONYMIZ]	ANON
49	#####	Completed	e4004a6b4f	bbe2b7c59;	3ffa4a71a5;	stantyan	#####	#####	1:02:01	1:17:00	0:14:59	Chevrolet C	Anton	Chevrolet	Male	59.938426	30.348279	60.015673	30.409926	[ANONYMIZ]	ANON
50	#####	Completed	e49979ce97	e44cceba3e	3ffa4a71a5;	stantyan	#####	#####	0:37:41	0:45:00	0:07:19	Ford Focus	Valeriy	Ford	Male	60.016093	30.40966	59.842317	30.319145	[ANONYMIZ]	ANON
51	#####	Completed	4d92f7fd2e	72d4ed2f0c	3ffa4a71a5;	stantyan	#####	#####	0:42:26	0:55:00	0:12:34	Honda CR-V	Aleksey	Honda	Male	60.01616	30.409194	59.84233	30.319067	[ANONYMIZ]	ANON
52	#####	Completed	0697b6dd1c	bb10f17be2	3ffa4a71a5;	stantyan	#####	#####	0:05:24	0:08:00	0:02:36	Hyundai Sol	Denis	Hyundai	Male	60.015113	30.388899	60.01593	30.407802	[ANONYMIZ]	ANON
53	#####	Completed	54b6a258de	6035b1f913	3ffa4a71a5;	stantyan	#####	#####	0:23:40	0:35:00	2:11:20	Hyundai Sol	Viktor	Hyundai	Male	56.795427	60.612876	56.842798	60.593533	[ANONYMIZ]	ANON
54	#####	Completed	2a545858dc	b314c4ed51	3ffa4a71a5;	stantyan	#####	#####	0:13:29	0:33:00	0:19:31	Suzuki Swift	Valentin	Suzuki	Male	56.835338	60.599533	56.795247	60.612822	[ANONYMIZ]	ANON
55	#####	Completed	2bf0f836d2	08e2b4b73;	3ffa4a71a5;	stantyan	#####	#####	0:30:25	0:42:00	0:11:35	Kia Optima	Konstantin	Kia	Male	56.818351	60.538783	56.754579	60.809916	[ANONYMIZ]	ANON
56	#####	Completed	a66a7fde3a	08e2b4b73;	3ffa4a71a5;	stantyan	#####	#####	0:24:35	0:25:00	0:00:25	Volkswagen	Konstantin	Volkswagen	Male	56.754657	60.809913	56.794925	60.609017	[ANONYMIZ]	ANON
57	#####	Completed	1ab9e6f3cb	acfdcb835;	3ffa4a71a5;	stantyan	#####	#####	0:11:27	0:19:00	0:07:33	Hyundai Sol	Dmitriy	Hyundai	Male	56.795531	60.612566	56.827389	60.598018	[ANONYMIZ]	ANON
58	#####	Completed	328a7ed09;	2d20c209a;	3ffa4a71a5;	stantyan	#####	#####	0:12:23	0:28:00	0:15:37	Nissan Almr	Konstantin	Nissan	Male	56.827501	60.597904	56.795447	60.612977	[ANONYMIZ]	ANON
59	#####	Completed	2f48c68a2d	b4617db77;	3ffa4a71a5;	stantyan	#####	#####	0:14:53	0:24:00	0:09:07	Renault Dus	Mihail	Renault	Male	56.803948	60.555167	56.795337	60.612807	[ANONYMIZ]	ANON
60	#####	Completed	444819db0c	2e214d1c9f	3ffa4a71a5;	stantyan	#####	#####	0:15:37	0:31:00	0:15:23	Mazda MAZ	Vladimir	Mazda	Male	56.795439	60.612664	56.843381	60.591195	[ANONYMIZ]	ANON
61	#####	Completed	0fffa07abb8	74042fe65;	3ffa4a71a5;	stantyan	#####	#####	0:15:41	0:22:00	0:06:19	Volkswagen	Anton	Volkswagen	Male	56.795458	60.612796	56.842877	60.594053	[ANONYMIZ]	ANON
62	#####	Completed	87539006f2	51d2fca895	3ffa4a71a5;	stantyan	#####	#####	0:13:52	0:18:00	0:04:08	Peugeot 20;	Danil	Peugeot	Male	56.837945	60.600501	56.796989	60.614437	[ANONYMIZ]	ANON
63	#####	Completed	7590c2e05;	8dd6dd1c2;	c3ffa4a71a5;	stantyan	13-09-2015	13-09-2015	0:18:46	0:26:00	0:07:14	Dacia Duster	Nikolay	Dacia	Male	60.015903	30.409211	60.040901	30.299059	[ANONYMIZ]	ANON
64	#####	Completed	15aae4b6e1	6c1a6a05a3	3ffa4a71a5;	stantyan	13-09-2015	13-09-2015	0:20:01	0:28:00	0:07:59	Hyundai Sol	Igor	Hyundai	Male	60.004807	30.299834	60.016069	30.409218	[ANONYMIZ]	ANON
65	#####	Completed	4e1c923546	1c2111f3f;	3ffa4a71a5;	stantyan	13-09-2015	13-09-2015	0:13:00	0:35:00	0:22:00	Chrysler Vo	Nikolay	Chrysler	Male	56.795383	60.611239	56.750034	60.802219	[ANONYMIZ]	ANON
66	#####	Completed	cd250e2f1c	0202322f;	3ffa4a71a5;	stantyan	13-09-2015	13-09-2015	0:29:37	2:48:00	2:18:23	Renault Ne	Vladimir	Renault	Male	56.750988	60.799555	56.829888	60.56663	[ANONYMIZ]	ANON
67	#####	Completed	c21b0a3e97	0927f61a3;	3ffa4a71a5;	stantyan	13-09-2015	13-09-2015	0:32:39	0:39:00	0:06:21	Nissan X-Tr	Yachey	Nissan	Male	59.79903	30.273571	60.0161	30.409211	[ANONYMIZ]	ANON
68	#####	Completed	5f930c5a9;	e9f2a70459	3ffa4a71a5;	stantyan	14-09-2015	14-09-2015	0:16:01	0:27:00	0:10:59	Renault Me	Sergey	Renault	Male	60.016317	30.395937	60.03358	30.410183	[ANONYMIZ]	ANON
69	#####	Completed	c9e7cf6ce2	dbb3e13fa;	3ffa4a71a5;	stantyan	16-09-2015	16-09-2015	0:07:45	0:19:00	0:11:15	Hyundai Sol	Andrey	Hyundai	Male	60.013661	30.409377	60.0033	30.367926	[ANONYMIZ]	ANON
70	#####	Completed	e8a5c850e8	166cf40ae	3ffa4a71a5;	stantyan	16-09-2015	16-09-2015	0:11:27	0:14:00	0:02:33	Mazda MAZ	Ilya	Mazda	Male	60.033322	30.366692	60.016172	30.409271	[ANONYMIZ]	ANON
71	#####	Completed	bfbadc7f1;	83a10a3fb;	3ffa4a71a5;	stantyan	17-09-2015	17-09-2015	0:10:47	0:18:00	0:07:13	BMW 7-seri	Konstantin	BMW	Male	60.018877	30.409196	60.01354	30.392946	[ANONYMIZ]	ANON
72	#####	Completed	0f6fa273c;	1552a7e65;	3ffa4a71a5;	stantyan	17-09-2015	17-09-2015	0:30:05	0:45:00	0:15:55	Opel Astra	Aleksandr	Opel	Male	60.016182	30.409367	59.921127	30.387815	[ANONYMIZ]	ANON
73	uberides-dataset-updated																				


```

b<-c()
for(k in c(1:length(data$trip_start_address))) {
  a<-""
  for(i in c(1:nchar(data$trip_start_address[k]))) {
    for(j in c(substr(data$trip_start_address[k],i,i))) {
      if(j %in% letters | j %in% LETTERS | j %in% c(0:9) | j %in% "," | j
%in% ".") {
        a<-paste(a,j)
        a<-str_replace_all(a," ","")
      }
    }
  }
  b<-append(b,a)
}
data$trip_start_address<-b

#cleaning trip_end_address column
library(stringr)
b<-c()
for(k in c(1:length(data$trip_end_address))) {
  a<-""
  for(i in c(1:nchar(data$trip_end_address[k]))) {
    for(j in c(substr(data$trip_end_address[k],i,i))) {
      if(j %in% letters | j %in% LETTERS | j %in% c(0:9) | j %in% "," | j
%in% ".") {
        a<-paste(a,j)
        a<-str_replace_all(a," ","")
      }
    }
  }
  b<-append(b,a)
}
data$trip_end_address<-b
#changing trip_completed_at to timestamp
library(lubridate)
b<-c()
for(i in c(1:length(data$trip_completed_at))) {
  a<-strsplit(data$trip_completed_at,"at")
  b<-append(b,mdy_hm(paste(a[[i]][1],a[[i]][2])))
}

```

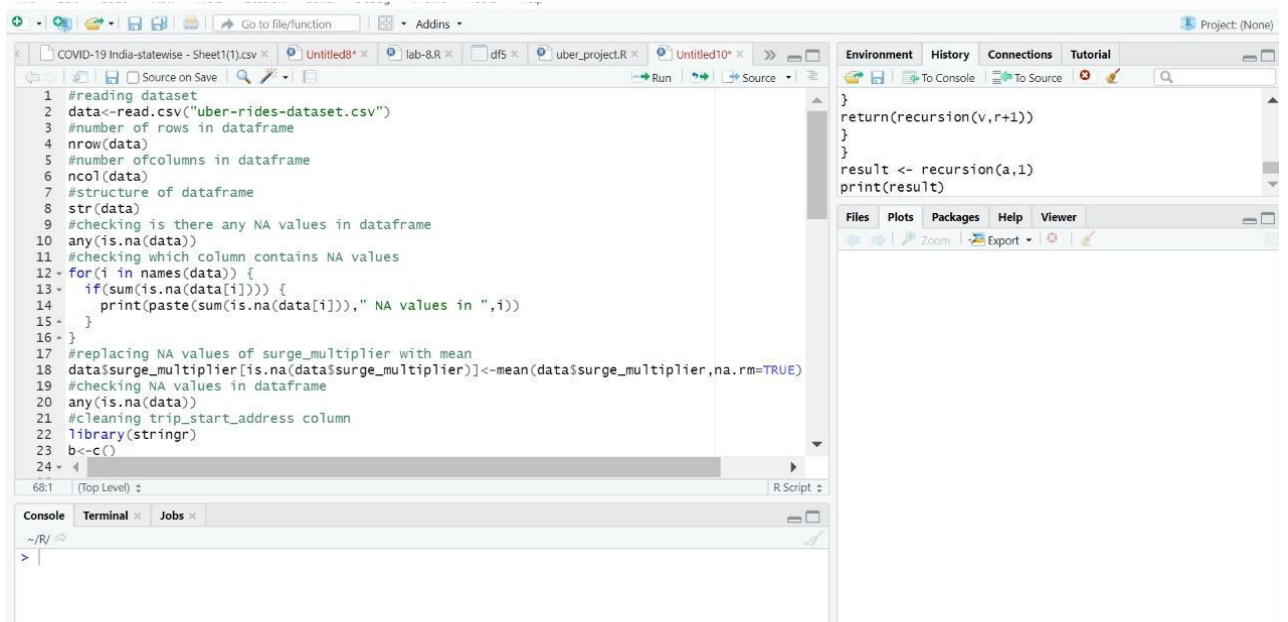
b

```
data$trip_completed_at<-b
```

```
#summary of data after cleaning
```

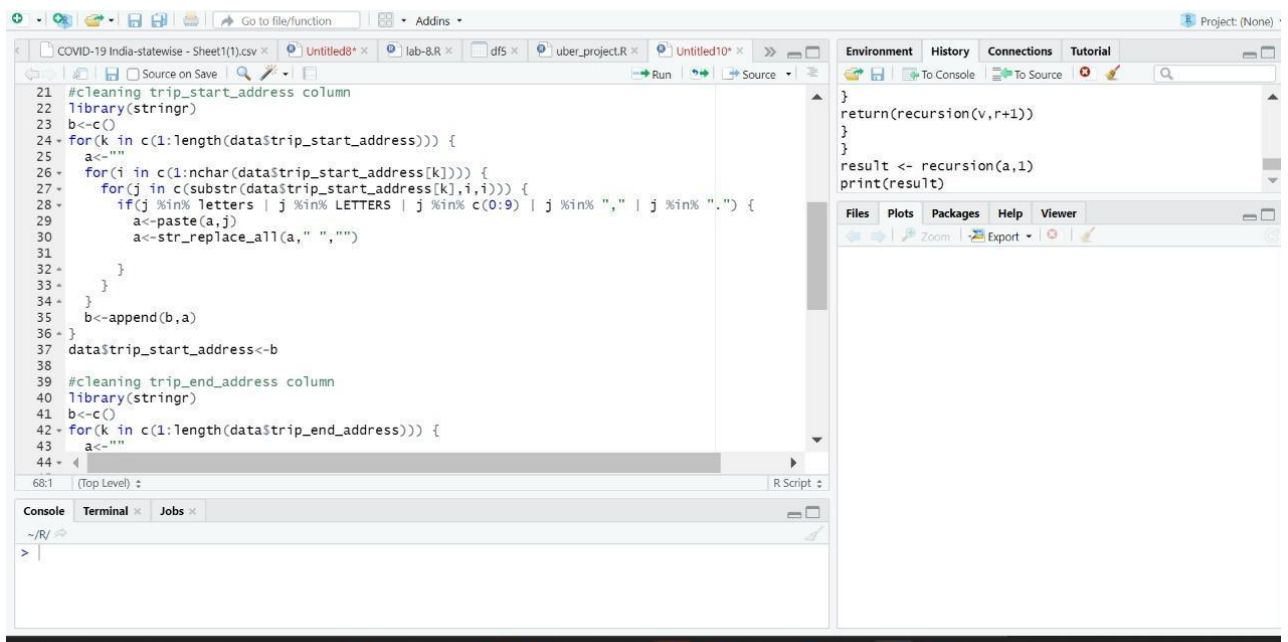
```
summary(data)
```

```
write.csv(data,"uber-rides-dataset-updated.csv")
```



The screenshot shows the RStudio IDE with a script editor on the left and a console at the bottom. The script editor contains R code for reading a dataset, checking for NA values, replacing them with the mean, and cleaning the trip_start_address column. The console shows the output of the code.

```
1 #reading dataset
2 data<-read.csv("uber-rides-dataset.csv")
3 #number of rows in dataframe
4 nrow(data)
5 #number of columns in dataframe
6 ncol(data)
7 #structure of dataframe
8 str(data)
9 #checking is there any NA values in dataframe
10 any(is.na(data))
11 #checking which column contains NA values
12 for(i in names(data)) {
13   if(sum(is.na(data[i]))) {
14     print(paste(sum(is.na(data[i])), " NA values in ",i))
15   }
16 }
17 #replacing NA values of surge_multiplier with mean
18 data$surge_multiplier[is.na(data$surge_multiplier)]<-mean(data$surge_multiplier,na.rm=TRUE)
19 #checking NA values in dataframe
20 any(is.na(data))
21 #cleaning trip_start_address column
22 library(stringr)
23 b<-c()
24
```



The screenshot shows the RStudio IDE with a script editor on the left and a console at the bottom. The script editor contains R code for cleaning the trip_start_address and trip_end_address columns. The console shows the output of the code.

```
21 #cleaning trip_start_address column
22 library(stringr)
23 b<-c()
24 for(k in c(1:length(data$trip_start_address))) {
25   a<-""
26   for(i in c(1:nchar(data$trip_start_address[k]))) {
27     for(j in c(substr(data$trip_start_address[k],i,i))) {
28       if(j %in% letters | j %in% LETTERS | j %in% c(0:9) | j %in% "," | j %in% ".") {
29         a<-paste(a,j)
30         a<-str_replace_all(a, " ", "")
31       }
32     }
33   }
34   b<-append(b,a)
35 }
36 data$trip_start_address<-b
37
38 #cleaning trip_end_address column
39 library(stringr)
40 b<-c()
41 for(k in c(1:length(data$trip_end_address))) {
42   a<-""
43   for(i in c(1:nchar(data$trip_end_address[k]))) {
44     for(j in c(substr(data$trip_end_address[k],i,i))) {
45       if(j %in% letters | j %in% LETTERS | j %in% c(0:9) | j %in% "," | j %in% ".") {
46         a<-paste(a,j)
47         a<-str_replace_all(a, " ", "")
48       }
49     }
50   }
51   b<-append(b,a)
52 }
53 data$trip_end_address<-b
54
```

```
46 - 1r(j %1n% letters | j %1n% LETTERS | j %1n% c(0:9) | j %1n% " " | j %1n% "-") {  
47 -   a<-paste(a,j)  
48 -   a<-str_replace_all(a, " ", "")  
49 -  
50 - }  
51 - }  
52 - }  
53 - b<-append(b,a)  
54 - }  
55 - data$trip_end_address<-b  
56 - #changing trip_completed_at to timestamp  
57 - library(lubridate)  
58 - b<-c()  
59 - for(i in c(1:length(data$trip_completed_at))) {  
60 -   a<-strsplit(data$trip_completed_at,"at")  
61 -   b<-append(b,mdy_hm(paste(a[[1]][1],a[[1]][2])))  
62 - }  
63 - b  
64 - data$trip_completed_at<-b  
65 - #summary of data after cleaning  
66 - summary(data)  
67 - write.csv(data,"uber-rides-dataset-updated.csv")  
68 -  
}  
return(recursion(v,r+1))  
}  
}  
result <- recursion(a,1)  
print(result)  
Files Plots Packages Help Viewer  
Zoom Export  
68:1 (Top Level) R Script  
Console Terminal Jobs  
~/R/ >
```


Conclusion:

There were very few clearly erroneous entries in the dataset and a small proportion of suspicious cases or *anomalies* that warrant further internal analysis. All taxi and for-hire-vehicles companies operating in the city, which include Uber, Lyft, and others release their data periodically. An update is published twice a year. The destination data were missing, and an extremely small number of cases had missing trip distance and destination. The imputation method chosen for the latter set was the mean distance and duration of their respective origin-destination pair. The entries with missing destination were left unchanged, although the information from the vast number of complete cases could potentially be used to determine the most probable destination.

The relation between a trip's duration and distance is not entirely linear. Rather, it approximates to a power function because shorter trips, occurring mostly within busy areas of traffic, tend to result in lower average trip speed. Analysis and undoubtedly highlighted the critical importance of a well-defined business problem, which directs all coding efforts to a particular purpose and reveals key details. This business case also attempted to demonstrate the basic use of python in everyday business activities, showing how fun, important, and fun it can be. Behavioural change of customers raising more complaints about taxi services.

Reference:

- I. <https://data-flair.training/blogs/r-data-science-project-uber-data-analysis/>
- II. <https://www.sciencedirect.com/science/article/abs/pii/S2214367X20302027>

Thank You