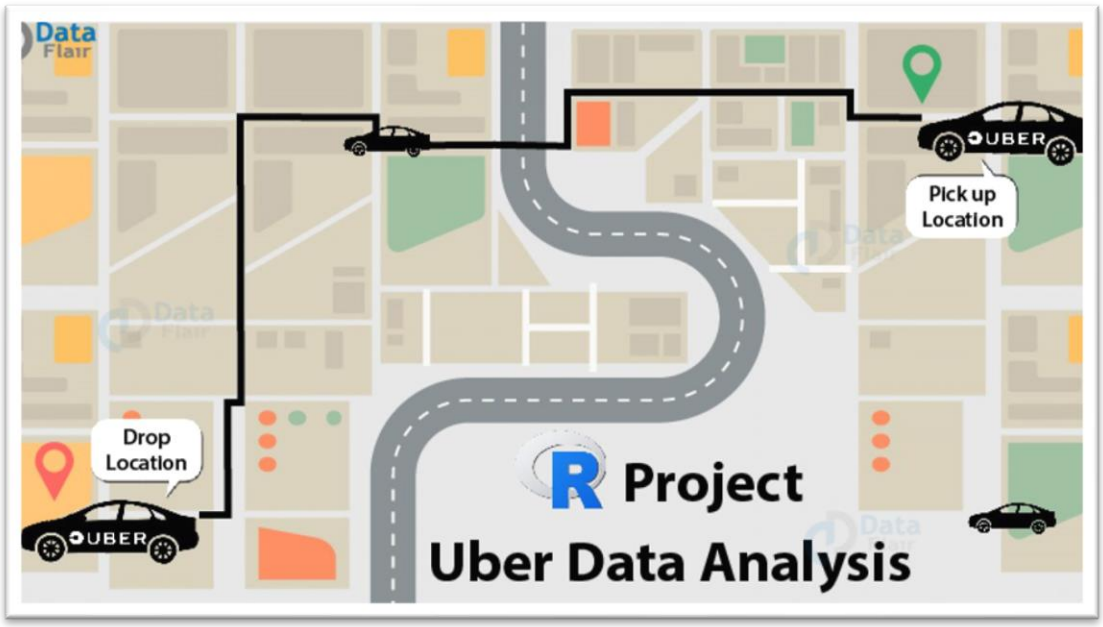
A Project Report On



Uber Data Analysis

Submitted as course project for

MACHINE LEARNING AND ARTIFICIAL INTELIGENCE

ML820

By
Swaib Ilias Mazumder
ASTU Roll No.: 172010007044

Under the Guidance of

Eckovation

Preface:

Uber Technologies, Inc., commonly known as Uber, offers vehicles for hire, food delivery (Uber Eats), package delivery, couriers, freight transportation, and, through a partnership with Lime, electric bicycle and motorized scooter rental. The company is based in San Francisco and has operations in over 900 metropolitan areas worldwide. It is one of the largest providers in the gig economy and is also a pioneer in the development of self-driving cars.

Acknowledgement

It’s a matter of great pleasure to present this progress report on “Uber Data Analysis”. I am very grateful to Eckovation to give me the opportunity to build this project.
I am also grateful to our ASTU coordinator along with Eckovation Faculty Members for their support and helping hand. Without them it was not possible to build.

Thanking you
Swaib Ilias Mazumder

Content:

Abstract

Talking about our 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.

Introduction

Uber has a massive database of drivers, so as soon as you request a car, Uber's algorithm goes right to work – in 15 seconds or less, it matches you with the driver closest to you. In the background Uber is storing data for every trip taken — even when the driver has no passengers. All of this data is stored and leveraged to predict supply and demand, as well as setting fares. Uber also looks at how transportation is handled across cities and tries to adjust for bottlenecks and other common issues.

Uber also gathers data on its drivers. In addition to collecting non-identifiable information about their vehicle and their location, Uber also monitors their speed and acceleration, and checks to see if they are working for a competing company as well (such as Lyft).

If you're reading this wondering if it's a gross invasion of privacy – you aren't the first. But Uber is very clear about how it uses the data gathered on its platform. A section of its privacy policy for U.S. customers and drivers reads:

Uber uses your personal data in an anonymised and aggregated form to closely monitor which features of the Service are used most, to analyze usage patterns and to determine where we should offer or focus our Service. We may share this information with third parties for industry analysis and statistics.

Objectives of this project:

- Finding traveling time and Calculating the average speed of the trip.

- Visualizing the data in terms of trips per hour of the day, per day of the week, and per month of the year.

- From the above step finding out in which month highest trips are made.

Implementation

1. Importing necessary libraries:

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

import matplotlib.pyplot as plt
from builtins import list
import matplotlib
matplotlib.style.use('ggplot')

import math

import datetime
```

We are using the following libraries in this project:

- Numpy
- Pandas
- Matplotlib
- List
- Ggplot
- Math
- Datetime

2. Reading the data into their designated variables:

```
In [2]: uber_df=pd.read_csv("datasets_1026_1855_My Uber Drives - 2016.csv")
```

```
In [3]: uber_df.head()
```

Out[3]:

	START_DATE*	END_DATE*	CATEGORY*	START*	STOP*	MILES*	PURPOSE*
0	1/1/2016 21:11	1/1/2016 21:17	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain
1	1/2/2016 1:25	1/2/2016 1:37	Business	Fort Pierce	Fort Pierce	5.0	NaN
2	1/2/2016 20:25	1/2/2016 20:38	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies
3	1/5/2016 17:31	1/5/2016 17:45	Business	Fort Pierce	Fort Pierce	4.7	Meeting
4	1/6/2016 14:42	1/6/2016 15:49	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit

```
In [4]: uber_df.tail()
```

Out[4]:

	START_DATE*	END_DATE*	CATEGORY*	START*	STOP*	MILES*	PURPOSE*
1151	12/31/2016 13:24	12/31/2016 13:42	Business	Kar?chi	Unknown Location	3.9	Temporary Site
1152	12/31/2016 15:03	12/31/2016 15:38	Business	Unknown Location	Unknown Location	16.2	Meeting
1153	12/31/2016 21:32	12/31/2016 21:50	Business	Katunayake	Gampaha	6.4	Temporary Site
1154	12/31/2016 22:08	12/31/2016 23:51	Business	Gampaha	Ilukwatta	48.2	Temporary Site
1155	Totals	NaN	NaN	NaN	NaN	12204.7	NaN

Here the dataset is read using pd.read_csv and saved in “uber_df”.
uber_df.head() displays the dataset from the start to some numbers.
uber_df.tail() displays the dataset from the end to some numbers.

3. Cleaning the dataset:

In [5]:
```
# Remove uncessary data
uber_df = uber_df[:-1]
```

In [6]:
```
# fix data types of data columns
def convert_time(column_name):
    y=[]
    for x in uber_df[column_name]:
        y.append(datetime.datetime.strptime(x, "%m/%d/%Y %H:%M"))

    uber_df[column_name] = y
```

In [7]:
```
column_date=uber_df[['START_DATE*','END_DATE*']]
for x in column_date:
    convert_time(x)
```

In [8]:
```
# check that all data is fixed and ready to work on it
uber_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1155 entries, 0 to 1154
Data columns (total 7 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   START_DATE*  1155 non-null   datetime64[ns]
 1   END_DATE*    1155 non-null   datetime64[ns]
 2   CATEGORY*    1155 non-null   object
 3   START*       1155 non-null   object
 4   STOP*        1155 non-null   object
 5   MILES*       1155 non-null   float64
 6   PURPOSE*     653 non-null    object
dtypes: datetime64[ns](2), float64(1), object(4)
memory usage: 63.3+ KB
```

First we are removing the unnecessary data.

Then we are fixing the date types of date column so that we can use this for calculating time for a trip.

4. Finding traveling time and calculating speed of the trip:

In [11]:
```
# calculate duration of each trip in minutes
minutes=[]
uber_df['Duration_Minutes'] = uber_df['END_DATE*'] - uber_df['START_DATE*']
uber_df['Duration_Minutes']
for x in uber_df['Duration_Minutes']:
    minutes.append(x.seconds / 60)

uber_df['Duration_Minutes'] = minutes
```

In [21]:
```
# calculate speed of each trip in kilometers/hour
count = 0
speed=[]
while count < len(uber_df):
    x=uber_df['MILES*'][count]
    x=x*1.60934
    y=uber_df['Duration_Minutes'][count]/60
    if(y!=0):
        speed.append(x / y)
    else:
        speed.append(0)
    count=count+1
uber_df['Speed in Kmph'] = speed
```

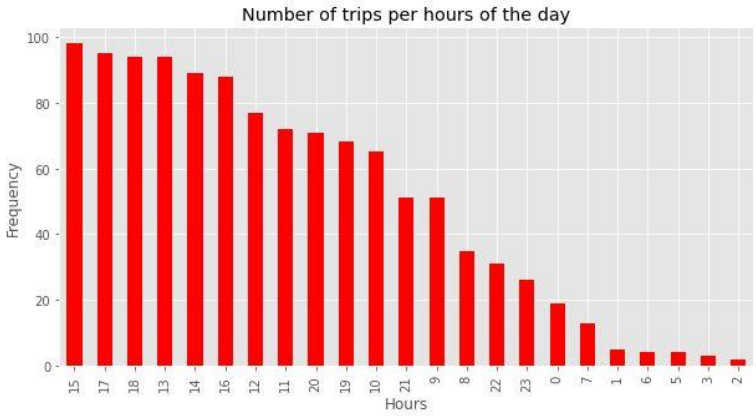
The above is how we are calculating the time and speed for each trip. Below is the Time and Speed for each trip:

In [29]:
```
print(uber_df)
```

```
            START_DATE*           END_DATE* CATEGORY*            START*  \
0    2016-01-01 21:11:00 2016-01-01 21:17:00  Business       Fort Pierce
1    2016-01-02 01:25:00 2016-01-02 01:37:00  Business       Fort Pierce
2    2016-01-02 20:25:00 2016-01-02 20:38:00  Business       Fort Pierce
3    2016-01-05 17:31:00 2016-01-05 17:45:00  Business       Fort Pierce
4    2016-01-06 14:42:00 2016-01-06 15:49:00  Business       Fort Pierce
...                  ...                 ...       ...              ...
1150 2016-12-31 01:07:00 2016-12-31 01:14:00  Business           Kar?chi
1151 2016-12-31 13:24:00 2016-12-31 13:42:00  Business           Kar?chi
1152 2016-12-31 15:03:00 2016-12-31 15:38:00  Business  Unknown Location
1153 2016-12-31 21:32:00 2016-12-31 21:50:00  Business       Katunayake
1154 2016-12-31 22:08:00 2016-12-31 23:51:00  Business           Gampaha

                STOP*  MILES*         PURPOSE*  Month  Week  \
0         Fort Pierce     5.1   Meal/Entertain      1     1
1         Fort Pierce     5.0              NaN      1     1
2         Fort Pierce     4.8  Errand/Supplies      1     1
3         Fort Pierce     4.7          Meeting      1     1
4      West Palm Beach    63.7   Customer Visit      1     1
...                ...     ...              ...    ...   ...
1150           Kar?chi     0.7          Meeting     12     4
1151  Unknown Location     3.9   Temporary Site     12     4
1152  Unknown Location    16.2          Meeting     12     4
1153           Gampaha     6.4   Temporary Site     12     4
1154         Ilukwatta    48.2   Temporary Site     12     4

      Duration_Minutes      Speed
0                  6.0  82.076340
1                 12.0  40.233500
2                 13.0  35.653071
3                 14.0  32.416706
4                 67.0  91.804440
...                ...        ...
1150               7.0   9.656040
1151              18.0  20.921420
1152              35.0  44.693671
1153              18.0  34.332587
1154             103.0  45.186517

[1155 rows x 11 columns]
```

5. Number of trips per hours of the day:

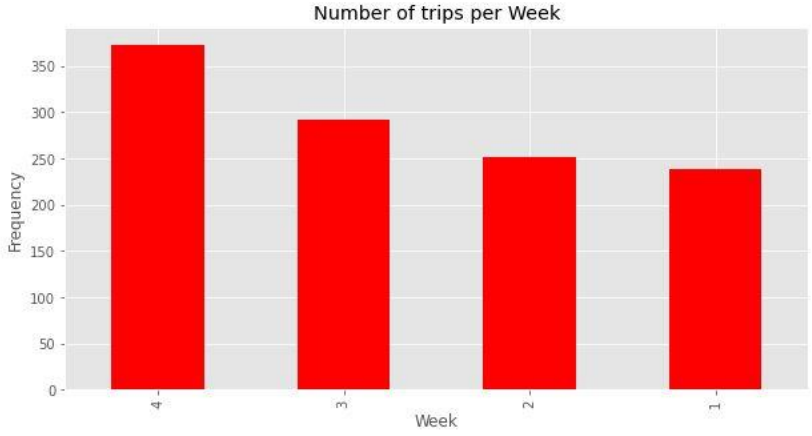
```
In [34]: # plot number of trips per hour of the day
         hours = uber_df['START_DATE*'].dt.hour.value_counts()
         hours.plot(kind='bar',color='red',figsize=(10,5))
         plt.xlabel('Hours')
         plt.ylabel('Frequency')
         plt.title('Number of trips per hours of the day')

Out[34]: Text(0.5, 1.0, 'Number of trips per hours of the day')
```



From the above plot, we can see the number of trips are highest at 3:00 pm of every day.

6. Number of trips per day of the week:

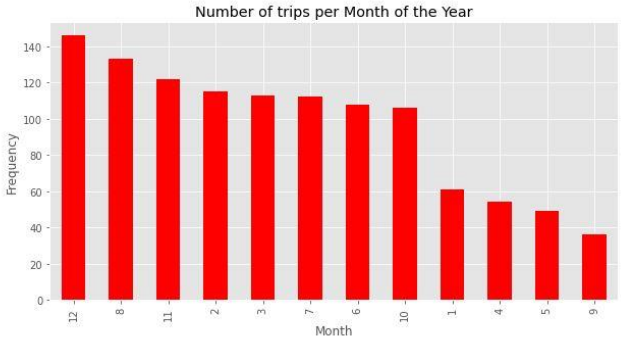
```
In [33]: # plot number of trips at each week
         x = uber_df['Week'].value_counts()
         x.plot(kind='bar',figsize=(10,5),color='red')
         plt.xlabel('Week')
         plt.ylabel('Frequency')
         plt.title('Number of trips per Week')

Out[33]: Text(0.5, 1.0, 'Number of trips per Week')
```



From the above plot, we can see the number of trips are highest in the last week of the month.

7. Number of trips per month of the year:

```
In [32]: # plot number of trips at each month of the year and we can see that maximum number of trips are done in the month of december
         x = uber_df['Month'].value_counts()
         x.plot(kind='bar',figsize=(10,5),color='red')
         plt.xlabel('Month')
         plt.ylabel('Frequency')
         plt.title('Number of trips per Month of the Year')

Out[32]: Text(0.5, 1.0, 'Number of trips per Month of the Year')
```



From the above plot, we can see the number of trips is highest in the month of december.

The Challenges:

• The data:

The quality of the data determines the outcome of our model. Cleaning and processing the data, understanding it, playing with it, plotting it, cuddling it. Make sure you explore every aspect of it.

• Perhaps the most frequent challenge in big data efforts is the inaccessibility of data sets from external sources.

• It is necessary for the data to be available in an accurate, complete and timely manner because if data in the companies information system is to be used to make accurate decisions in time then it becomes necessary for data to be available in this manner.