PUBLIC TRANSPORT EFFICENCY ANALYSIS

Content For Project Phase 2:

Data Source:

When conducting a public transport efficiency analysis, it's crucial to ensure data accuracy, consistency, and relevance to your specific analysis goals. You may need to collaborate with local transportation authorities, use data from multiple sources, and employ data cleansing and integration techniques to make the most of these data sources.

Dataset Link:

https://www.kaggle.com/datasets/rednivrug/unisys?select=20140711.CSV

TripID	RouteID	StopID	StopName	WeekBeginning	NumberOfBo...
23631	100	14156	181 Cross Rd	2013-06-30 00:00:00	1
23631	100	14144	177 Cross Rd	2013-06-30 00:00:00	1
23632	100	14132	175 Cross Rd	2013-06-30 00:00:00	1
23633	100	12266	Zone A Arndale Interchange	2013-06-30 00:00:00	2
23633	100	14147	178 Cross Rd	2013-06-30 00:00:00	1
23634	100	13907	9A Marion Rd	2013-06-30 00:00:00	1
23634	100	14132	175 Cross Rd	2013-06-30 00:00:00	1
23634	100	13335	9A Holbrooks Rd	2013-06-30 00:00:00	1
23634	100	13875	9 Marion Rd	2013-06-30 00:00:00	1
23634	100	13045	206 Holbrooks Rd	2013-06-30 00:00:00	1
23635	100	13335	9A Holbrooks Rd	2013-06-30 00:00:00	1
23635	100	13383	8A Marion Rd	2013-06-30 00:00:00	1
23635	100	13586	8D Marion Rd	2013-06-30 00:00:00	2
23635	100	12726	23 Findon Rd	2013-06-30	1

DATA COLLECTION:

When conducting a public transport efficiency analysis, it's crucial to ensure data accuracy, consistency, and relevance to your specific analysis goals. You may need to collaborate with local transportation authorities, use data from multiple sources, and employ data cleansing and integration techniques to make the most of these data sources.

PROGRAM:

PUBLIC TRANSPORT EFFICENCY ANALYSIS

Data aggregation:

the process of collecting and summarizing data from various sources or individual data points into a more concise and useful format. It's often done to analyze trends, make decisions, or generate reports. Common aggregation methods include averaging, summing, counting, and grouping data. Aggregated data is easier to work with and can provide valuable insights for businesses, researchers, and analysts.

Data visualization:

Data visualization is the graphical representation of data to help people understand and interpret information more easily. It involves creating visual elements like charts, graphs, maps, and dashboards to present data in a visually appealing and comprehensible way.

Data exploration:

Data exploration is the initial phase of data analysis where you investigate and familiarize yourself with a dataset to gain a better understanding of its characteristics.

PROGRAM

```
    %matplotlib inline
    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
    import datetime
    import os
    from math import sqrt
    import warnings
    from IPython.core.interactiveshell import InteractiveShell
    InteractiveShell.ast_node_interactivity = "all"
    warnings.filterwarnings('ignore')
In [2]:
    linkcode
    data = pd.read_csv('../input/unisys/ptsboardingsummary/20140711.CSV')
    data.shape
    data.head(10)
Out[2]:
     (10857234, 6)
```

Data Visualization

```
In [27]:

fig,axrr=plt.subplots(2,2,figsize=(15,15))

ax=axrr[0][0]
ax.set_title("No of Boardings")
data['NumberOfBoardings'].value_counts().sort_index().head(20).plot.bar(ax=ax
rr[0][0])
```

```
ax=axrr[0][1]
ax.set_title("WeekBeginning")
data['WeekBeginning'].value_counts().plot.area(ax=axrr[0][1])

ax=axrr[1][0]
ax.set_title("most Busiest Route")
data['RouteID'].value_counts().head(10).plot.bar(ax=axrr[1][0])

ax=axrr[1][1]
ax.set_title("least Busiest Route")
data['RouteID'].value_counts().tail(10).plot.bar(ax=axrr[1][1])
```

Out[27]:

```
Text(0.5,1,'No of Boardings')
```

Out[27]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff880af0940>
```

Out[27]:

```
Text(0.5,1,'WeekBeginning')
```

Out[27]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff709a6bb38>
```

Out[27]:

```
Text(0.5,1,'most Busiest Route')
```

Out[27]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff709a48e10>
```

Out[27]:

```
Text(0.5,1,'least Busiest Route')
```

Out[27]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff736bbafd0>
```


No of Boardings
4000000
3500000
3000000
2500000
2000000
1500000
1000000
500000
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
WeekBeginning
200000
175000
150000
125000
100000
75000
50000
25000
0
2013-08
2013-10
2013-12
2014-02
2014-04
2014-06
most Busiest Route
350000
300000
250000
200000
150000
100000
50000
0
G10
B10
M44
H30
300
H20
174
506
W90
H22
least Busiest Route
1.0
0.8
0.6
0.4
0.2
0.0
FX2
LS
FX10
FX3
480
632
659
FX9
FX1
494T

In[28]:

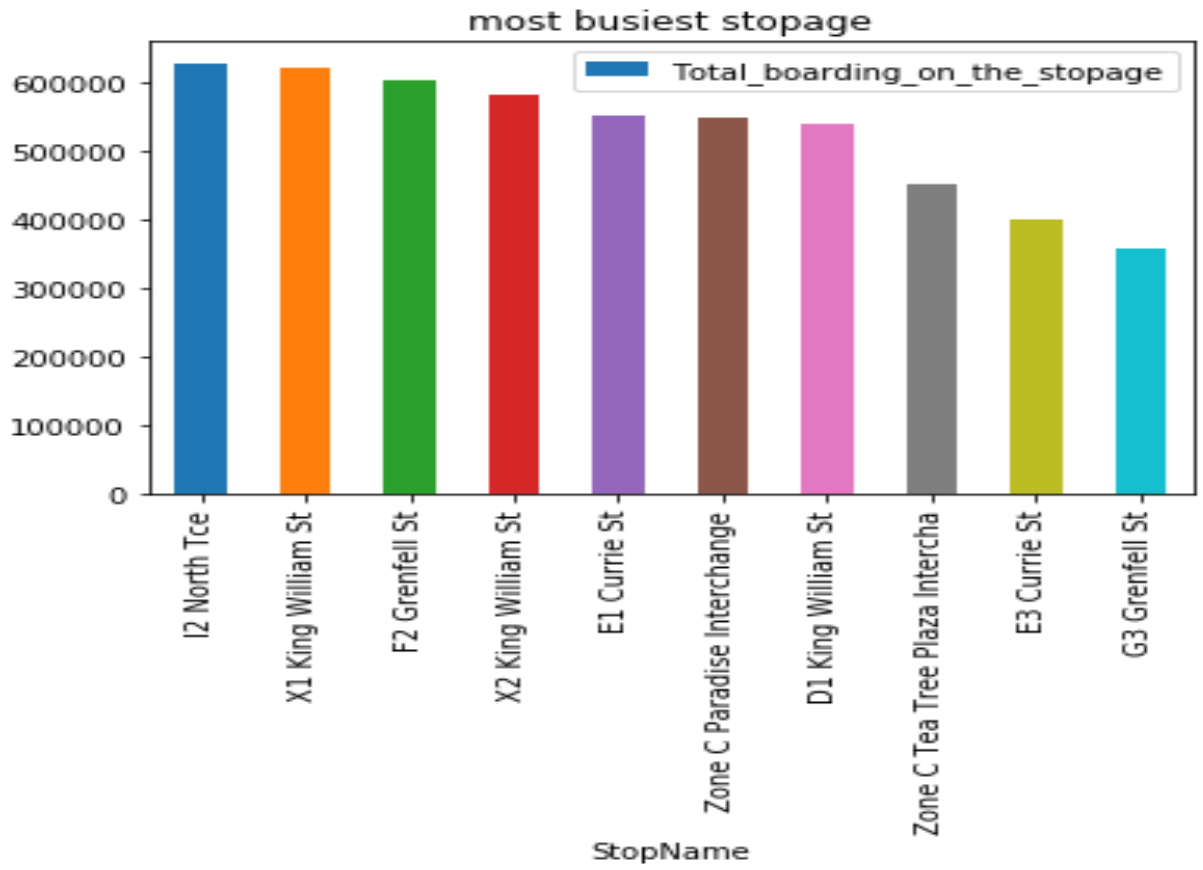
```
 stopageName_with_boarding = stopageName_with_boarding.sort_values('Total_
boarding_on_the_stopage', ascending = False)
stopageName_with_boarding.head(10)
```

StopName	Total_boarding_on_the_stopage	
3054	I2 North Tce	628859
3125	X1 King William St	622099
3032	F2 Grenfell St	604149
3130	X2 King William St	583227
3021	E1 Currie St	550396
3207	Zone C Paradise Interchange	547709
3015	D1 King William St	541046
3211	Zone C Tea Tree Plaza Intercha	451960
3025	E3 Currie St	399351
3039	G3 Grenfell St	356518

```
ax = stopageName_with_boarding.head(10).plot.bar(x='StopName', y='Total_board
ing_on_the_stopage', rot=90)
ax.set_title("most busiest stopage")
```

Out[30]:

```
Text(0.5,1,'most busiest stopage')
```

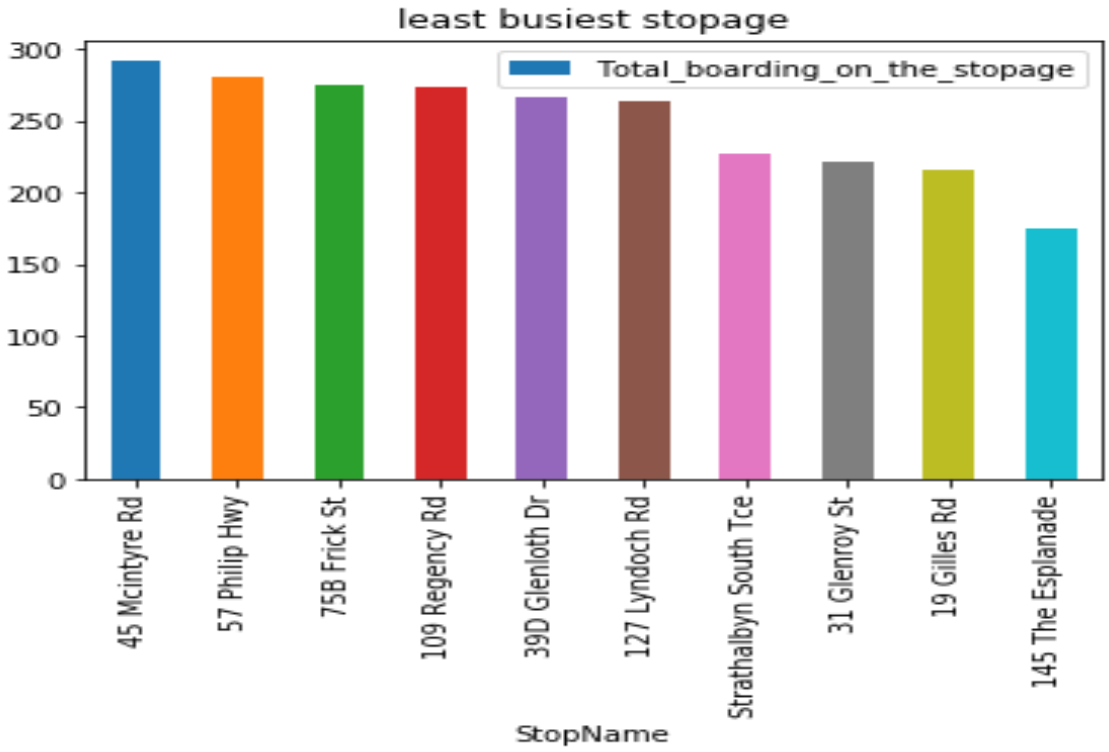


In [31]:

```
ax = stopageName_with_boarding.tail(10).plot.bar(x='StopName', y='Total_board
ing_on_the_stopage', rot=90)
ax.set_title("least busiest stopage")
```

Out[31]:

```
Text(0.5,1,'least busiest stopage')
```



In [32]:
```
data['WeekBeginning'].value_counts().mean()
```

Out[32]:
```
191508.66666666666
```

In [33]:
```
bb_grp = data.groupby(['dist_from_centre']).agg({'NumberOfBoardings': ['sum']
}).reset_index()
bb_grp.columns = bb_grp.columns.get_level_values(0)
bb_grp.head()
bb_grp.columns
bb_grp.tail()
```

Out[33]:

	dist_from_centre	NumberOfBoardings
0	0.000018	1892435

	dist_from_centre	NumberOfBoardings
1	0.131368	167535
2	0.309089	356518
3	0.314937	1484824
4	0.326005	120061

Out[33]:
```
Index(['dist_from_centre', 'NumberOfBoardings'], dtype='object')
```

Out[33]:

	dist_from_centre	NumberOfBoardings
2392	86.471064	18905
2393	94.826409	321
2394	99.625655	1101
2395	99.665190	4373
2396	99.748995	21216

```
import plotly.graph_objs as go
from plotly.offline import iplot

trace0 = go.Scatter(
    x = bb_grp['dist_from_centre'],
    y = bb_grp['NumberOfBoardings'],mode = 'lines+markers',name = 'X2 King Wi
lliam St')

data1 = [trace0]
layout = dict(title = 'Distance Vs Number of boarding',
              xaxis = dict(title = 'Distance from centre'),
              yaxis = dict(title = 'Number of Boardings'))
fig = dict(data=data1, layout=layout)
iplot(fig)
```

In [35]:

```
#
x = data["dist_from_centre"]
distance_10 = []
distance_10_50 = []
distance_50_100 = []
#distance_100_ = []
distance_100_more = []
total = 0
outlier = []
outlier_ = 0
for i in x:
    if(i<=10):
        distance_10.append(i)
        total += 1
    elif(i<=50):
        distance_10_50.append(i)
        total +=
    elif(i<=100):
        distance_50_100.append(i)
        total += 1
    #elif(i>100 and i< 2000):
        #distance_100_more.append(i)
        #total += 1
    #elif(i>2000):
        #outlier.append(i)
        #outlier_ += 1
```

In [36]:

```
print(outlier_)
```

```
0
```

```
In [37]:
y = len(distance_10)+len(distance_10_50)+len(distance_50_100)
#+len(distance_100_more)
#print(y)
#print(total)
```

```
In [38]:
    print(total)
    print("passangers, boarding the buses in the radious of 10Km from the cit
y center = ", (len(distance_10)/total)*100)

    print("passanger, boarding the buses from the distance of 10Km to 50Km fr
om the city center = ", (len(distance_10_50)/total)*100)

print("passanger, boarding the buses from the distance of 50Km to 100 from th
e city center = ", (len(distance_50_100)/total)*100)

#print("passanger, boarding the buses from the distance of 100Km and more from
the city center = ", (len(distance_100_more)/total)*100)
```

```
10341468

passangers, boarding the buses in the radious of 10Km from the city center
=  64.31275521038212

passanger, boarding the buses from the distance of 10Km to 50Km from the c
ity center =  33.16731241638035

passanger, boarding the buses from the distance of 50Km to 100 from the ci
ty center =  2.5199323732375327
```

CONCLUSION:

- ✓ the Phase 2 conclusion, we will summarize the key findings and in sights from the advanced regression techniques.
- ✓ We will reiterate the impact of these techniques on improving the accuracy.

. . .