https://drive.google.com/file/d/1QFE0zuTj-K_C6-BQbh0WxtcbWbLReWHI/view?usp=sharing

In [1]:
```
!gdown 1QFE0zuTj-K_C6-BQbh0WxtcbWbLReWHI
```

```
Downloading...
From: https://drive.google.com/uc?id=1QFE0zuTj-K_C6-BQbh0WxtcbWbLReWHI
To: C:\Users\91944\d2beiqkhq929f0.cloudfront.net_public_assets_assets_000_001_551_ori
ginal_delhivery_data.csv

  0%|          | 0.00/55.6M [00:00<?, ?B/s]
  1%|          | 524k/55.6M [00:00<00:15, 3.64MB/s]
  5%|4         | 2.62M/55.6M [00:00<00:04, 11.7MB/s]
  8%|8         | 4.72M/55.6M [00:00<00:03, 14.3MB/s]
 11%|#1        | 6.29M/55.6M [00:00<00:03, 13.6MB/s]
 14%|#4        | 7.86M/55.6M [00:00<00:03, 13.4MB/s]
 17%|#6        | 9.44M/55.6M [00:00<00:03, 13.1MB/s]
 20%|#9        | 11.0M/55.6M [00:00<00:03, 12.9MB/s]
 23%|##2       | 12.6M/55.6M [00:00<00:03, 12.9MB/s]
 25%|##5       | 14.2M/55.6M [00:01<00:03, 12.8MB/s]
 28%|##8       | 15.7M/55.6M [00:01<00:03, 12.9MB/s]
 31%|###1      | 17.3M/55.6M [00:01<00:03, 12.7MB/s]
 34%|###3      | 18.9M/55.6M [00:01<00:02, 12.8MB/s]
 37%|###6      | 20.4M/55.6M [00:01<00:02, 12.8MB/s]
 40%|###9      | 22.0M/55.6M [00:01<00:02, 12.7MB/s]
 42%|####2     | 23.6M/55.6M [00:01<00:02, 12.8MB/s]
 45%|####5     | 25.2M/55.6M [00:01<00:02, 12.7MB/s]
 48%|####8     | 26.7M/55.6M [00:02<00:02, 12.9MB/s]
 51%|#####     | 28.3M/55.6M [00:02<00:02, 12.7MB/s]
 54%|#####3    | 29.9M/55.6M [00:02<00:02, 12.7MB/s]
 57%|#####6    | 31.5M/55.6M [00:02<00:01, 12.8MB/s]
 59%|#####9    | 33.0M/55.6M [00:02<00:01, 12.8MB/s]
 62%|######2   | 34.6M/55.6M [00:02<00:01, 12.8MB/s]
 65%|######5   | 36.2M/55.6M [00:02<00:01, 12.7MB/s]
 68%|######7   | 37.7M/55.6M [00:02<00:01, 12.8MB/s]
 71%|#######   | 39.3M/55.6M [00:03<00:01, 12.7MB/s]
 74%|#######3  | 40.9M/55.6M [00:03<00:01, 12.8MB/s]
 76%|#######6  | 42.5M/55.6M [00:03<00:01, 12.7MB/s]
 79%|#######9  | 44.0M/55.6M [00:03<00:00, 12.9MB/s]
 82%|########2 | 45.6M/55.6M [00:03<00:00, 12.7MB/s]
 85%|########4 | 47.2M/55.6M [00:03<00:00, 12.7MB/s]
 88%|########7 | 48.8M/55.6M [00:03<00:00, 12.7MB/s]
 90%|######### | 50.3M/55.6M [00:03<00:00, 12.7MB/s]
 93%|#########3| 51.9M/55.6M [00:04<00:00, 12.8MB/s]
 96%|#########6| 53.5M/55.6M [00:04<00:00, 12.7MB/s]
 99%|#########8| 55.1M/55.6M [00:04<00:00, 12.9MB/s]
100%|##########| 55.6M/55.6M [00:04<00:00, 12.7MB/s]
```

In [45]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv("d2beiqkhq929f0.cloudfront.net_public_assets_assets_000_001_551_origina
df
```

Out[45]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND3881
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND3881
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND3881
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND3881
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND3881
...	...	...	...	...	...	
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND1310
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND1310
144864	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND1310
144865	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND1310
144866	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND1310

144867 rows × 24 columns

In [76]:
```
pip install nbconvert
```

```
Requirement already satisfied: nbconvert in c:\users\91944\anaconda3\lib\site-package
s (6.5.4)
Requirement already satisfied: tinycss2 in c:\users\91944\anaconda3\lib\site-packages
(from nbconvert) (1.2.1)
Requirement already satisfied: packaging in c:\users\91944\anaconda3\lib\site-package
s (from nbconvert) (22.0)
Requirement already satisfied: entrypoints>=0.2.2 in c:\users\91944\anaconda3\lib\sit
e-packages (from nbconvert) (0.4)
Requirement already satisfied: beautifulsoup4 in c:\users\91944\anaconda3\lib\site-pa
ckages (from nbconvert) (4.11.1)
Requirement already satisfied: pygments>=2.4.1 in c:\users\91944\anaconda3\lib\site-p
ackages (from nbconvert) (2.11.2)
Requirement already satisfied: lxml in c:\users\91944\anaconda3\lib\site-packages (fr
om nbconvert) (4.9.1)
Requirement already satisfied: mistune<2,>=0.8.1 in c:\users\91944\anaconda3\lib\site
-packages (from nbconvert) (0.8.4)
Requirement already satisfied: nbclient>=0.5.0 in c:\users\91944\anaconda3\lib\site-p
ackages (from nbconvert) (0.5.13)
Requirement already satisfied: jupyterlab-pygments in c:\users\91944\anaconda3\lib\si
te-packages (from nbconvert) (0.1.2)
Requirement already satisfied: pandocfilters>=1.4.1 in c:\users\91944\anaconda3\lib\s
ite-packages (from nbconvert) (1.5.0)
Requirement already satisfied: jinja2>=3.0 in c:\users\91944\anaconda3\lib\site-packa
ges (from nbconvert) (3.1.2)
Requirement already satisfied: nbformat>=5.1 in c:\users\91944\anaconda3\lib\site-pac
kages (from nbconvert) (5.7.0)
Requirement already satisfied: traitlets>=5.0 in c:\users\91944\anaconda3\lib\site-pa
ckages (from nbconvert) (5.7.1)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\91944\anaconda3\lib\site-p
ackages (from nbconvert) (2.1.1)
Requirement already satisfied: defusedxml in c:\users\91944\anaconda3\lib\site-packag
es (from nbconvert) (0.7.1)
Requirement already satisfied: bleach in c:\users\91944\anaconda3\lib\site-packages
(from nbconvert) (4.1.0)
Requirement already satisfied: jupyter-core>=4.7 in c:\users\91944\anaconda3\lib\site
-packages (from nbconvert) (5.2.0)
Requirement already satisfied: pywin32>=1.0 in c:\users\91944\anaconda3\lib\site-pack
ages (from jupyter-core>=4.7->nbconvert) (305.1)
Requirement already satisfied: platformdirs>=2.5 in c:\users\91944\anaconda3\lib\site
-packages (from jupyter-core>=4.7->nbconvert) (2.5.2)
Requirement already satisfied: nest-asyncio in c:\users\91944\anaconda3\lib\site-pack
ages (from nbclient>=0.5.0->nbconvert) (1.5.6)
Requirement already satisfied: jupyter-client>=6.1.5 in c:\users\91944\anaconda3\lib
\site-packages (from nbclient>=0.5.0->nbconvert) (7.3.4)
Requirement already satisfied: jsonschema>=2.6 in c:\users\91944\anaconda3\lib\site-p
ackages (from nbformat>=5.1->nbconvert) (4.17.3)
Requirement already satisfied: fastjsonschema in c:\users\91944\anaconda3\lib\site-pa
ckages (from nbformat>=5.1->nbconvert) (2.16.2)
Requirement already satisfied: soupsieve>1.2 in c:\users\91944\anaconda3\lib\site-pac
kages (from beautifulsoup4->nbconvert) (2.3.2.post1)
Requirement already satisfied: webencodings in c:\users\91944\anaconda3\lib\site-pack
ages (from bleach->nbconvert) (0.5.1)
Requirement already satisfied: six>=1.9.0 in c:\users\91944\anaconda3\lib\site-packag
es (from bleach->nbconvert) (1.16.0)
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in c:\us
ers\91944\anaconda3\lib\site-packages (from jsonschema>=2.6->nbformat>=5.1->nbconver
t) (0.18.0)
Requirement already satisfied: attrs>=17.4.0 in c:\users\91944\anaconda3\lib\site-pac
kages (from jsonschema>=2.6->nbformat>=5.1->nbconvert) (22.1.0)
Requirement already satisfied: tornado>=6.0 in c:\users\91944\anaconda3\lib\site-pack
```

```
ages (from jupyter-client>=6.1.5->nbclient>=0.5.0->nbconvert) (6.1)
Requirement already satisfied: pyzmq>=23.0 in c:\users\91944\anaconda3\lib\site-packa
ges (from jupyter-client>=6.1.5->nbclient>=0.5.0->nbconvert) (23.2.0)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\91944\anaconda3\lib
\site-packages (from jupyter-client>=6.1.5->nbclient>=0.5.0->nbconvert) (2.8.2)
Note: you may need to restart the kernel to use updated packages.
```

Expand columns

In [49]:
```
pd.set_option('display.max_columns',100)
df.head(2)
```

Out[49]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA

EDA

In [6]:
```
df.shape
```

Out[6]:
```
(144867, 24)
```

In [7]:
```
len(df.groupby(by=["trip_uuid","source_name","destination_center"]))
```

Out[7]:
```
26368
```

In [2]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 144867 entries, 0 to 144866
Data columns (total 24 columns):
 #   Column                           Non-Null Count   Dtype  
---  ------                           --------------   -----  
 0   data                             144867 non-null  object 
 1   trip_creation_time               144867 non-null  object 
 2   route_schedule_uuid              144867 non-null  object 
 3   route_type                       144867 non-null  object 
 4   trip_uuid                        144867 non-null  object 
 5   source_center                    144867 non-null  object 
 6   source_name                      144574 non-null  object 
 7   destination_center               144867 non-null  object 
 8   destination_name                 144606 non-null  object 
 9   od_start_time                    144867 non-null  object 
 10  od_end_time                      144867 non-null  object 
 11  start_scan_to_end_scan           144867 non-null  float64
 12  is_cutoff                        144867 non-null  bool   
 13  cutoff_factor                    144867 non-null  int64  
 14  cutoff_timestamp                 144867 non-null  object 
 15  actual_distance_to_destination   144867 non-null  float64
 16  actual_time                      144867 non-null  float64
 17  osrm_time                        144867 non-null  float64
 18  osrm_distance                    144867 non-null  float64
 19  factor                           144867 non-null  float64
 20  segment_actual_time              144867 non-null  float64
 21  segment_osrm_time                144867 non-null  float64
 22  segment_osrm_distance            144867 non-null  float64
 23  segment_factor                   144867 non-null  float64
dtypes: bool(1), float64(10), int64(1), object(12)
memory usage: 25.6+ MB
```

In [64]:
```
df["data"].value_counts()
```

Out[64]:
```
training    104858
test         40009
Name: data, dtype: int64
```

In [65]:
```
df["route_type"].value_counts()
```

Out[65]:
```
FTL        99660
Carting    45207
Name: route_type, dtype: int64
```

Drop unknown fields

In [47]:
```
columns_to_drop = ['is_cutoff', 'cutoff_factor', 'cutoff_timestamp',"factor","segment_
df=df.drop(columns=columns_to_drop)
```

Datatypes Conversions

In [67]:
```
df.dtypes
```

Out[67]:
```
data                                  object
trip_creation_time                    object
route_schedule_uuid                   object
route_type                            object
trip_uuid                             object
source_center                         object
source_name                           object
destination_center                    object
destination_name                      object
od_start_time                         object
od_end_time                           object
start_scan_to_end_scan               float64
is_cutoff                               bool
cutoff_factor                          int64
cutoff_timestamp                      object
actual_distance_to_destination       float64
actual_time                          float64
osrm_time                            float64
osrm_distance                        float64
factor                               float64
segment_actual_time                  float64
segment_osrm_time                    float64
segment_osrm_distance                float64
segment_factor                       float64
dtype: object
```

In [46]:
```
df["trip_creation_time"] = pd.to_datetime(df["trip_creation_time"])
df["od_start_time"] = pd.to_datetime(df["od_start_time"])
df["od_end_time"] = pd.to_datetime(df["od_end_time"])
```

5) Missing values detection & Treatment

In [41]:
```
df.duplicated().sum()
```

Out[41]:
```
0
```

In [100…
```
df.describe()
```

Out[100]:

	start_scan_to_end_scan	actual_distance_to_destination	actual_time	osrm_time	osrm_dista
count	144867.000000	144867.000000	144867.000000	144867.000000	144867.000
mean	961.262986	234.073372	416.927527	213.868272	284.771
std	1037.012769	344.990009	598.103621	308.011085	421.119
min	20.000000	9.000045	9.000000	6.000000	9.008
25%	161.000000	23.355874	51.000000	27.000000	29.914
50%	449.000000	66.126571	132.000000	64.000000	78.525
75%	1634.000000	286.708875	513.000000	257.000000	343.193
max	7898.000000	1927.447705	4532.000000	1686.000000	2326.199

In [77]:
```
df.isnull().sum()
```

Out[77]:
```
data                                0
trip_creation_time                  0
route_schedule_uuid                 0
route_type                          0
trip_uuid                           0
source_center                       0
source_name                       293
destination_center                  0
destination_name                  261
od_start_time                       0
od_end_time                         0
start_scan_to_end_scan              0
actual_distance_to_destination      0
actual_time                         0
osrm_time                           0
osrm_distance                       0
factor                              0
segment_actual_time                 0
segment_osrm_time                   0
segment_osrm_distance               0
dtype: int64
```

In [43]:
```
df["source_name"].value_counts()
```

Out[43]:
```
Gurgaon_Bilaspur_HB (Haryana)            23347
Bangalore_Nelmngla_H (Karnataka)          9975
Bhiwandi_Mankoli_HB (Maharashtra)         9088
Pune_Tathawde_H (Maharashtra)             4061
Hyderabad_Shamshbd_H (Telangana)          3340
                                         ...
Shahjhnpur_NavdaCln_D (Uttar Pradesh)        1
Soro_UttarDPP_D (Orissa)                     1
Kayamkulam_Bhrnikvu_D (Kerala)               1
Krishnanagar_AnadiDPP_D (West Bengal)        1
Faridabad_Old (Haryana)                      1
Name: source_name, Length: 1498, dtype: int64
```

In [6]:
```
df["source_name"].fillna(df["source_name"].mode()[0],inplace=True)
```

In [9]:
```
df["destination_name"].value_counts()
```

Out[9]:
```
Gurgaon_Bilaspur_HB (Haryana)        15192
Bangalore_Nelmngla_H (Karnataka)     11019
Bhiwandi_Mankoli_HB (Maharashtra)     5492
Hyderabad_Shamshbd_H (Telangana)      5142
Kolkata_Dankuni_HB (West Bengal)      4892
                                     ...
Hyd_Trimulgherry_Dc (Telangana)          1
Vijayawada (Andhra Pradesh)              1
Baghpat_Barout_D (Uttar Pradesh)         1
Mumbai_Sanpada_CP (Maharashtra)          1
Basta_Central_DPP_1 (Orissa)             1
Name: destination_name, Length: 1468, dtype: int64
```

In [7]:
```
df["destination_name"].fillna(df["destination_name"].mode()[0],inplace=True)
```

In [80]:
```
df.isnull().sum().sum()
```

Out[80]:
```
0
```

In [103…
```
df.isnull().sum()
```

Out[103]:
```
data                                    0
trip_creation_time                      0
route_schedule_uuid                     0
route_type                              0
trip_uuid                               0
source_center                           0
source_name                             0
destination_center                      0
destination_name                        0
od_start_time                           0
od_end_time                             0
start_scan_to_end_scan                  0
actual_distance_to_destination          0
actual_time                             0
osrm_time                               0
osrm_distance                           0
segment_actual_time                     0
segment_osrm_time                       0
segment_osrm_distance                   0
dtype: int64
```

In [8]:
```
df.dtypes
```

Out[8]:
```
data                                    object
trip_creation_time                datetime64[ns]
route_schedule_uuid                     object
route_type                              object
trip_uuid                               object
source_center                           object
source_name                             object
destination_center                      object
destination_name                        object
od_start_time                     datetime64[ns]
od_end_time                       datetime64[ns]
start_scan_to_end_scan                   float64
is_cutoff                                   bool
cutoff_factor                              int64
cutoff_timestamp                  datetime64[ns]
actual_distance_to_destination           float64
actual_time                              float64
osrm_time                                float64
osrm_distance                            float64
factor                                   float64
segment_actual_time                      float64
segment_osrm_time                        float64
segment_osrm_distance                    float64
segment_factor                           float64
dtype: object
```

3) Merging rows and aggregation of fields

In [34]:
```
data=df.groupby(["trip_uuid","source_center","destination_center"])[["actual_time","os
                        Total_seg_osrm_time=("segment_osrm_time","sum"),Total_osrm_dista
```

```
                        Total_seg_osrm_distance=("segment_osrm_distance","sum")).reset_i
data
```

Out[34]:

	trip_uuid	source_center	destination_center	Total_actualtime	Total_osrmtime	Tota
0	trip-153671041653548748	IND209304AAA	IND000000ACB	732.0	349.0	
1	trip-153671041653548748	IND462022AAA	IND209304AAA	830.0	394.0	
2	trip-153671042288605164	IND561203AAB	IND562101AAA	47.0	26.0	
3	trip-153671042288605164	IND572101AAA	IND561203AAB	96.0	42.0	
4	trip-153671043369099517	IND000000ACB	IND160002AAC	611.0	212.0	
...	...	...	...	...	...	
26363	trip-153861115439069069	IND628204AAA	IND627657AAA	51.0	41.0	
26364	trip-153861115439069069	IND628613AAA	IND627005AAA	90.0	48.0	
26365	trip-153861115439069069	IND628801AAA	IND628204AAA	30.0	14.0	
26366	trip-153861118270144424	IND583119AAA	IND583101AAA	233.0	42.0	
26367	trip-153861118270144424	IND583201AAA	IND583119AAA	42.0	26.0	

26368 rows × 9 columns

In [35]:

```
data1=df.groupby(["trip_uuid"])[["actual_time","osrm_time","segment_actual_time","segm
                        Total_seg_osrm_time=("segment_osrm_time","sum"),Total_osrm_dista
                        Total_seg_osrm_distance=("segment_osrm_distance","sum")).reset_i
data1
```

Out[35]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_tin
0	trip-153671041653548748	830.0	394.0	1548.0	1008
1	trip-153671042288605164	96.0	42.0	141.0	65
2	trip-153671043369099517	2736.0	1529.0	3308.0	1941
3	trip-153671046011330457	59.0	15.0	59.0	16
4	trip-153671052974046625	147.0	46.0	340.0	115
...	...	...	...	...	
14812	trip-153861095625827784	49.0	34.0	82.0	62
14813	trip-153861104386292051	21.0	12.0	21.0	11
14814	trip-153861106442901555	190.0	29.0	281.0	88
14815	trip-153861115439069069	90.0	50.0	258.0	221
14816	trip-153861118270144424	233.0	42.0	274.0	67

14817 rows × 7 columns

2 Building features [Destination Name,Source Name,Trip_creation_time]

In [125…

```
df.head()
```

Out[125]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA

Split and extract features into separate columns

In [33]:

```
df.insert(8,"City",df['destination_name'].str.split('_').str[0])
df.insert(9,"Place",df['destination_name'].str.split('_').str[1])
df.insert(10,"State",df['destination_name'].str.extract(r'\((.*?)\)',expand=False).fil

df.drop("destination_name",axis=1,inplace=True)

df.head()
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3802, in Index.get_loc
(self, key, method, tolerance)
   3801 try:
-> 3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:138, in pandas._libs.index.
IndexEngine.get_loc()

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:165, in pandas._libs.index.
IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:5745, in pandas._libs.hashtable.PyObject
HashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:5753, in pandas._libs.hashtable.PyObject
HashTable.get_item()

KeyError: 'destination_name'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[33], line 1
----> 1 df.insert(8,"City",df['destination_name'].str.split('_').str[0])
      2 df.insert(9,"Place",df['destination_name'].str.split('_').str[1])
      3 df.insert(10,"State",df['destination_name'].str.extract(r'\((.*?)\)',expand=F
alse).fillna("Unknown"))

File ~\anaconda3\lib\site-packages\pandas\core\frame.py:3807, in DataFrame.__getitem_
_(self, key)
   3805 if self.columns.nlevels > 1:
   3806     return self._getitem_multilevel(key)
-> 3807 indexer = self.columns.get_loc(key)
   3808 if is_integer(indexer):
   3809     indexer = [indexer]

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3804, in Index.get_loc
(self, key, method, tolerance)
   3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:
-> 3804     raise KeyError(key) from err
   3805 except TypeError:
   3806     # If we have a listlike key, _check_indexing_error will raise
   3807     #  InvalidIndexError. Otherwise we fall through and re-raise
   3808     #  the TypeError.
   3809     self._check_indexing_error(key)

KeyError: 'destination_name'
```

In [11]:

```
df.insert(6,"From_City",df['source_name'].str.split('_').str[0])
df.insert(7,"From_Place",df['source_name'].str.split('_').str[1])
df.insert(8,"From_State",df['source_name'].str.extract(r'\((.*?)\)',expand=False).fil

df.drop("source_name",axis=1,inplace=True)

df.head()
df.tail()
```

Out[11]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_c
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND13102
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND13102
144864	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND13102
144865	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND13102
144866	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND13102

5 rows × 23 columns

In [36]:

```
df.insert(1,"Month",df["trip_creation_time"].dt.month)
df.insert(2,"Year",df["trip_creation_time"].dt.year)
df.insert(3,"Day",df["trip_creation_time"].dt.day)

df.drop("trip_creation_time",axis=1,inplace=True)
df.head()
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3802, in Index.get_loc
(self, key, method, tolerance)
   3801 try:
-> 3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:138, in pandas._libs.index.
IndexEngine.get_loc()

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:165, in pandas._libs.index.
IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:5745, in pandas._libs.hashtable.PyObject
HashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:5753, in pandas._libs.hashtable.PyObject
HashTable.get_item()

KeyError: 'trip_creation_time'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[36], line 1
----> 1 df.insert(1,"Month",df["trip_creation_time"].dt.month)
      2 df.insert(2,"Year",df["trip_creation_time"].dt.year)
      3 df.insert(3,"Day",df["trip_creation_time"].dt.day)

File ~\anaconda3\lib\site-packages\pandas\core\frame.py:3807, in DataFrame.__getitem_
_(self, key)
   3805 if self.columns.nlevels > 1:
   3806     return self._getitem_multilevel(key)
-> 3807 indexer = self.columns.get_loc(key)
   3808 if is_integer(indexer):
   3809     indexer = [indexer]

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3804, in Index.get_loc
(self, key, method, tolerance)
   3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:
-> 3804     raise KeyError(key) from err
   3805 except TypeError:
   3806     # If we have a listlike key, _check_indexing_error will raise
   3807     #  InvalidIndexError. Otherwise we fall through and re-raise
   3808     #  the TypeError.
   3809     self._check_indexing_error(key)

KeyError: 'trip_creation_time'
```

In [44]:
```
df.head()
```

Out[44]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA

5 rows × 24 columns

Check from where most orders are coming from (State, Corridor etc)

Busiest corridor, avg distance between them, avg time taken

In [48]:
```
df['time_taken'] = (df['od_end_time'] - df['od_start_time']).dt.total_seconds() / 3600
```

In [56]:
```
corridor_stats = df.groupby(['source_name', 'destination_name']).agg(

order_count=pd.NamedAgg(column='trip_uuid', aggfunc='count'),

avg_distance=pd.NamedAgg(column='actual_distance_to_destination', aggfunc='mean'),

avg_time_taken=pd.NamedAgg(column='time_taken', aggfunc='mean')).reset_index()

corridor_stats.sort_values(by="order_count",ascending=False).reset_index(drop=True)
```

Out[56]:

	source_name	destination_name	order_count	avg_distance	avg_time_taken
0	Gurgaon_Bilaspur_HB (Haryana)	Bangalore_Nelmngla_H (Karnataka)	4976	859.827666	51.634496
1	Bangalore_Nelmngla_H (Karnataka)	Gurgaon_Bilaspur_HB (Haryana)	3316	869.072245	52.099205
2	Gurgaon_Bilaspur_HB (Haryana)	Kolkata_Dankuni_HB (West Bengal)	2862	672.757980	40.417724
3	Gurgaon_Bilaspur_HB (Haryana)	Hyderabad_Shamshbd_H (Telangana)	1639	639.241145	43.188074
4	Gurgaon_Bilaspur_HB (Haryana)	Bhiwandi_Mankoli_HB (Maharashtra)	1617	552.345706	36.266030
...	...	...	...	...	...
2736	Anand_Vaghasi_IP (Gujarat)	Anand_VUNagar_DC (Gujarat)	1	9.383422	1.388117
2737	Hyd_Trimulgherry_Dc (Telangana)	Hyderabad_Alwal_I (Telangana)	1	9.187172	2.668265
2738	Anakapalle_Kothuru_D (Andhra Pradesh)	Visakhapatnam_Gajuwaka_IP (Andhra Pradesh)	1	20.611772	3.430120
2739	Nedumangad_Arsprmbu_D (Kerala)	Trivandrum_Mnanthla_H (Kerala)	1	9.043632	1.236933
2740	Malerkotla_DC (Punjab)	Dhuri_DMComDPP_D (Punjab)	1	17.100289	1.179387

2741 rows × 5 columns

In [52]:
```
# Find the busiest corridor

busiest_corridor = corridor_stats[corridor_stats['order_count'] == corridor_stats['orc

busiest_corridor
```

Out[52]:

	source_name	destination_name	order_count	avg_distance	avg_time_taken
1008	Gurgaon_Bilaspur_HB (Haryana)	Bangalore_Nelmngla_H (Karnataka)	4976	859.827666	51.634496

In [65]:
```
df[df["source_name" == "Gurgaon_Bilaspur_HB (Haryana)"]]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3802, in Index.get_loc
(self, key, method, tolerance)
   3801 try:
-> 3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:138, in pandas._libs.index.
IndexEngine.get_loc()

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:165, in pandas._libs.index.
IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:5745, in pandas._libs.hashtable.PyObject
HashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:5753, in pandas._libs.hashtable.PyObject
HashTable.get_item()

KeyError: False

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[65], line 1
----> 1 df[df["source_name" == "Gurgaon_Bilaspur_HB (Haryana)"]]

File ~\anaconda3\lib\site-packages\pandas\core\frame.py:3807, in DataFrame.__getitem_
_(self, key)
   3805 if self.columns.nlevels > 1:
   3806     return self._getitem_multilevel(key)
-> 3807 indexer = self.columns.get_loc(key)
   3808 if is_integer(indexer):
   3809     indexer = [indexer]

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3804, in Index.get_loc
(self, key, method, tolerance)
   3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:
-> 3804     raise KeyError(key) from err
   3805 except TypeError:
   3806     # If we have a listlike key, _check_indexing_error will raise
   3807     #  InvalidIndexError. Otherwise we fall through and re-raise
   3808     #  the TypeError.
   3809     self._check_indexing_error(key)

KeyError: False
```

Dropping unwanted columns

In [ ]:

In [28]:
```
columns_to_drop = ["actual_distance_to_destination","actual_time","osrm_time","osrm_di
# df=df.drop(columns=columns_to_drop,axis=1)
df.head(10)
```

Out[28]:

	data	Month	Year	Day	route_schedule_uuid	route_type	trip_uuid	source_cente
0	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AA
1	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AA
2	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AA
3	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AA
4	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AA
5	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388620AA
6	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388620AA
7	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388620AA
8	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388620AA
9	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388620AA

10 rows × 25 columns

In [29]:
```
df=df1.drop_duplicates()
df1=df.reset_index(drop=True)
df1
```

Out[29]:

	data	Month	Year	Day	route_schedule_uuid	route_type	trip_uuid	source
0	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388
1	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388
2	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388
3	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388
4	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388
...	...	...	...	...	...	...	...	
144862	training	9	2018	20	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131
144863	training	9	2018	20	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131
144864	training	9	2018	20	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131
144865	training	9	2018	20	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131
144866	training	9	2018	20	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131

144867 rows × 25 columns

In [70]:
```
df1.Month.value_counts()
```

Out[70]:
```
9     127349
10     17518
Name: Month, dtype: int64
```

In [72]:
```
df1.Year.value_counts()
```

Out[72]:
```
2018    144867
Name: Year, dtype: int64
```

In [32]:
```
z=data1.merge(df1,on="trip_uuid",how="left")
z
```

Out[32]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_ti
0	trip-153671041653548748	830.0	394.0	1548.0	100
1	trip-153671041653548748	830.0	394.0	1548.0	100
2	trip-153671041653548748	830.0	394.0	1548.0	100
3	trip-153671041653548748	830.0	394.0	1548.0	100
4	trip-153671041653548748	830.0	394.0	1548.0	100
...	...	...	...	...	
144862	trip-153861115439069069	90.0	50.0	258.0	22
144863	trip-153861118270144424	233.0	42.0	274.0	6
144864	trip-153861118270144424	233.0	42.0	274.0	6
144865	trip-153861118270144424	233.0	42.0	274.0	6
144866	trip-153861118270144424	233.0	42.0	274.0	6

144867 rows × 31 columns

In [85]:
```
df1.shape
```

Out[85]:
```
(26369, 18)
```

In [86]:
```
data1.shape
```

Out[86]:
```
(14817, 7)
```

Calculate the time taken between od_start_time and od_end_time and keep it as a feature

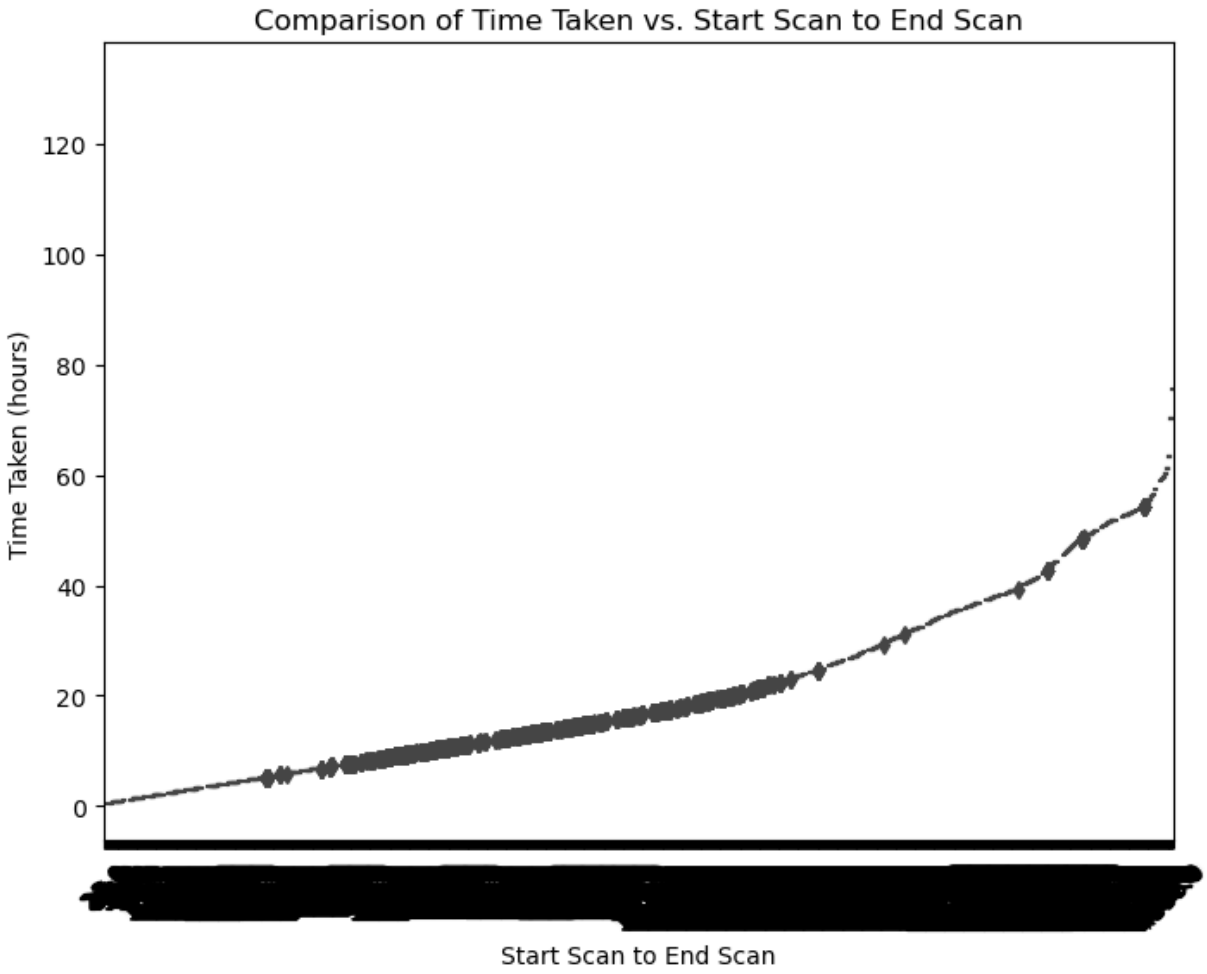
In [110…
```
# Calculate time taken

df1['time_taken'] = (df1['od_end_time'] - df1['od_start_time']).dt.total_seconds() / 3
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3802, in Index.get_loc
(self, key, method, tolerance)
   3801 try:
-> 3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:138, in pandas._libs.index.
IndexEngine.get_loc()

File ~\anaconda3\lib\site-packages\pandas\_libs\index.pyx:165, in pandas._libs.index.
IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:5745, in pandas._libs.hashtable.PyObject
HashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:5753, in pandas._libs.hashtable.PyObject
HashTable.get_item()

KeyError: 'od_end_time'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[110], line 3
      1 # Calculate time taken
----> 3 df1['time_taken'] = (df1['od_end_time'] - df1['od_start_time']).dt.total_seco
nds() / 3600

File ~\anaconda3\lib\site-packages\pandas\core\frame.py:3807, in DataFrame.__getitem_
_(self, key)
   3805 if self.columns.nlevels > 1:
   3806     return self._getitem_multilevel(key)
-> 3807 indexer = self.columns.get_loc(key)
   3808 if is_integer(indexer):
   3809     indexer = [indexer]

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3804, in Index.get_loc
(self, key, method, tolerance)
   3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:
-> 3804     raise KeyError(key) from err
   3805 except TypeError:
   3806     # If we have a listlike key, _check_indexing_error will raise
   3807     #  InvalidIndexError. Otherwise we fall through and re-raise
   3808     #  the TypeError.
   3809     self._check_indexing_error(key)

KeyError: 'od_end_time'
```

In [88]:
```
df1.drop(['od_start_time', 'od_end_time'], axis=1, inplace=True)
```

In [116…
```
# Box plot to compare time_taken and start_scan_to_end_scan

plt.figure(figsize=(8, 6))

sns.boxplot(x='start_scan_to_end_scan', y='time_taken', data=df1)
```

```
plt.xlabel('Start Scan to End Scan')

plt.ylabel('Time Taken (hours)')

plt.title('Comparison of Time Taken vs. Start Scan to End Scan')

plt.xticks(rotation=45)

plt.show()
```



Timetaken and start_scan_to_end_scan. Do hypothesis testing/ Visual analysis to check.

In [89]:
```
# H0 (null hypothesis): Trip start and end time difference (Timetaken) & timetaken fro
# destination(start_scan_to_end_scan) are totally independent.

# H1 (alternate hypothesis):Trip start and end time difference (Timetaken) & timetaker
# destination(start_scan_to_end_scan) are totally dependent.

from scipy.stats import ttest_ind
ttest_ind(df1["start_scan_to_end_scan"],df1["time_taken"])
```

Out[89]:
```
Ttest_indResult(statistic=108.09189505853819, pvalue=0.0)
```

In [98]:
```
from scipy.stats import f_oneway
f_oneway(df1["start_scan_to_end_scan"],df1["time_taken"])
```

Out[98]:
```
F_onewayResult(statistic=11683.857777346038, pvalue=0.0)
```

From ANOVA & Ttest ,found p-value is very low that both Time-taken to deliver from source to destination & Trip start time-Trip end time seems totally dependent.They are statistically significant.

Hypothesis testing/ visual analysis between actual_time aggregated value and OSRM time aggregated value

In [100…
```
# H0 (null hypothesis): actual_time &  OSRM time are totally independent.
# H1 (alternate hypothesis):actual_time &  OSRM time are totally dependent.


from scipy.stats import ttest_ind
ttest_ind(data1["Total_actualtime"],data1["Total_osrmtime"])
```

Out[100]:
```
Ttest_indResult(statistic=35.184305336896436, pvalue=1.031233484916594e-265)
```

From Ttest ,found that p-value is very low both Total_actualtime(Actual time taken to complete the delivery) & Total_osrmtime(he shortest path between points in a given map) They are statistically different and totally dependent.

Do hypothesis testing/ visual analysis between actual_time aggregated value and segment actual time aggregated value

In [101…
```
# H0 (null hypothesis):   actual_time &  segment actual time are totally independent.
# H1 (alternate hypothesis):actual_time &  segment actual time are totally dependent.


from scipy.stats import ttest_ind
ttest_ind(data1["Total_actualtime"],data1["Total_seg_actualtime"])
```

Out[101]:
```
Ttest_indResult(statistic=-12.498043250299483, pvalue=9.417815251922895e-36)
```

From ANOVA & Ttest ,found that p-value is very low both Total_actualtime(Actual time taken to complete the delivery) &Total_seg_actualtime(Total Time taken by the subset of the package delivery They are statistically significant and totally dependent.

Do hypothesis testing/ visual analysis between osrm distance aggregated value and segment osrm distance aggregated value

In [189…
```
# H0 (null hypothesis):   osrm distance &  ssegment osrm distance are totally indepenc
# H1 (alternate hypothesis):  osrm distance &  segment osrm distance are totally deper

from scipy.stats import ttest_ind
ttest_ind(data1["Total_osrm_distance"],data1["Total_seg_osrm_distance"])
```

Out[189]:
```
Ttest_indResult(statistic=-15.512609113583515, pvalue=4.6528686685002756e-54)
```

From ANOVA & Ttest ,found p-value is very low that both Total_osrm_distance(he shortest path between points in a given map) &Total_seg_actualtime(Distance covered by subset of the package delivery They are statistically significant and totally dependent.

Do hypothesis testing/ visual analysis between osrm time aggregated value and segment osrm time aggregated value

In [103…
```
# H0 (null hypothesis):   osrm time &  segment osrm time are totally independent.
# H1 (alternate hypothesis):  osrm time &  segment osrm time are totally dependent.

from scipy.stats import ttest_ind
ttest_ind(data1["Total_osrmtime"],data1["Total_seg_osrm_time"])
```

Out[103]:
```
Ttest_indResult(statistic=-18.27072888404281, pvalue=3.619072013555336e-74)
```

From ANOVA & Ttest ,found that p-value is very low both Total_osrmtime(shortest path between points in a given map) &Total_seg_osrm_time(Time taken by the subset of the package delivery)They are statistically significant and dependent on eachother.

Outlier detection & Treatment

IQR method

start_scan_to_end_scan float64 actual_distance_to_destination float64 actual_time float64 osrm_time float64 osrm_distance float64 segment_actual_time float64 segment_osrm_time float64 segment_osrm_distance float64

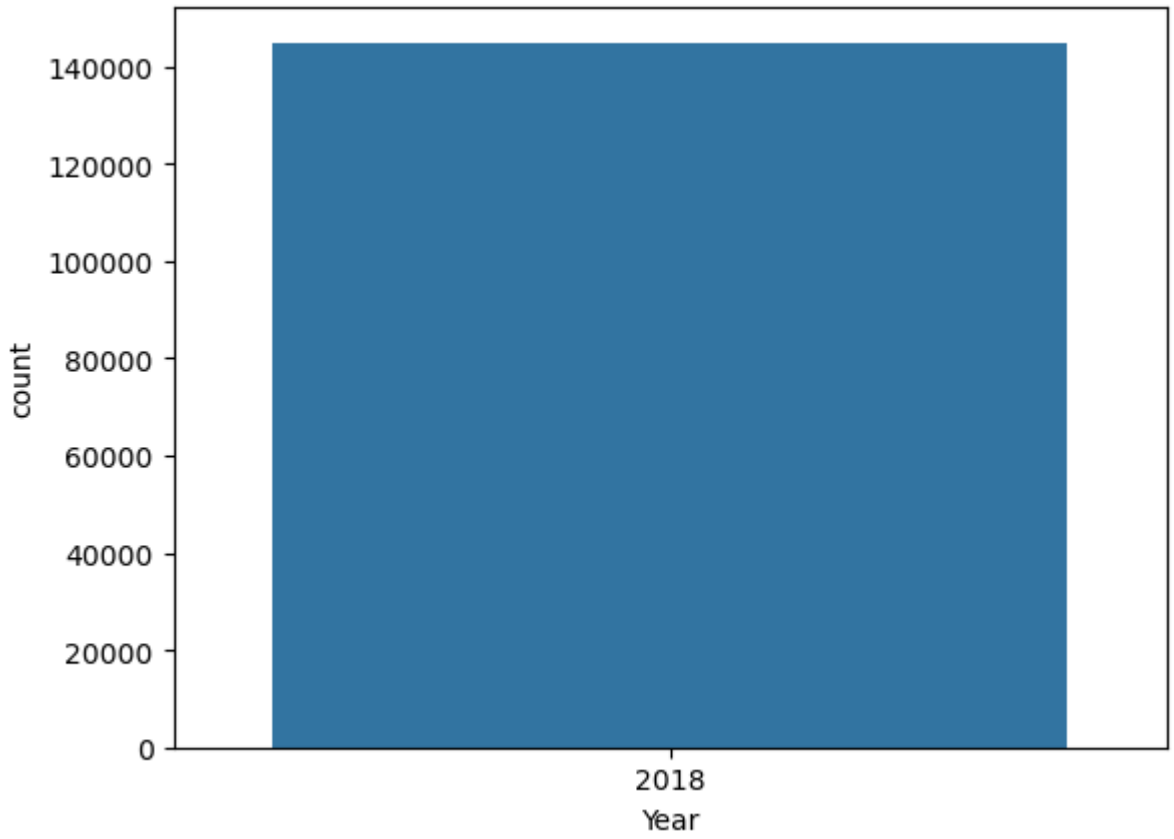
In [161…
```
z.describe()
```

Out[161]:

	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_time	Total_osrm_distanc
count	26369.000000	26369.000000	26369.000000	26369.000000	26369.00000
mean	288.626266	122.629603	433.919565	219.333005	155.93934
std	441.292474	212.666538	542.892810	304.356348	291.07754
min	9.000000	6.000000	9.000000	6.000000	9.07290
25%	79.000000	34.000000	105.000000	49.000000	37.60030
50%	145.000000	62.000000	269.000000	136.000000	72.81340
75%	279.000000	105.000000	520.000000	245.000000	130.63620
max	4532.000000	1686.000000	6230.000000	2564.000000	2326.19910

In [68]:
```
sns.countplot(data=z,x="Year")
```

Out[68]:
```
<Axes: xlabel='Year', ylabel='count'>
```



Outliers Treatment(IQR METHOD)

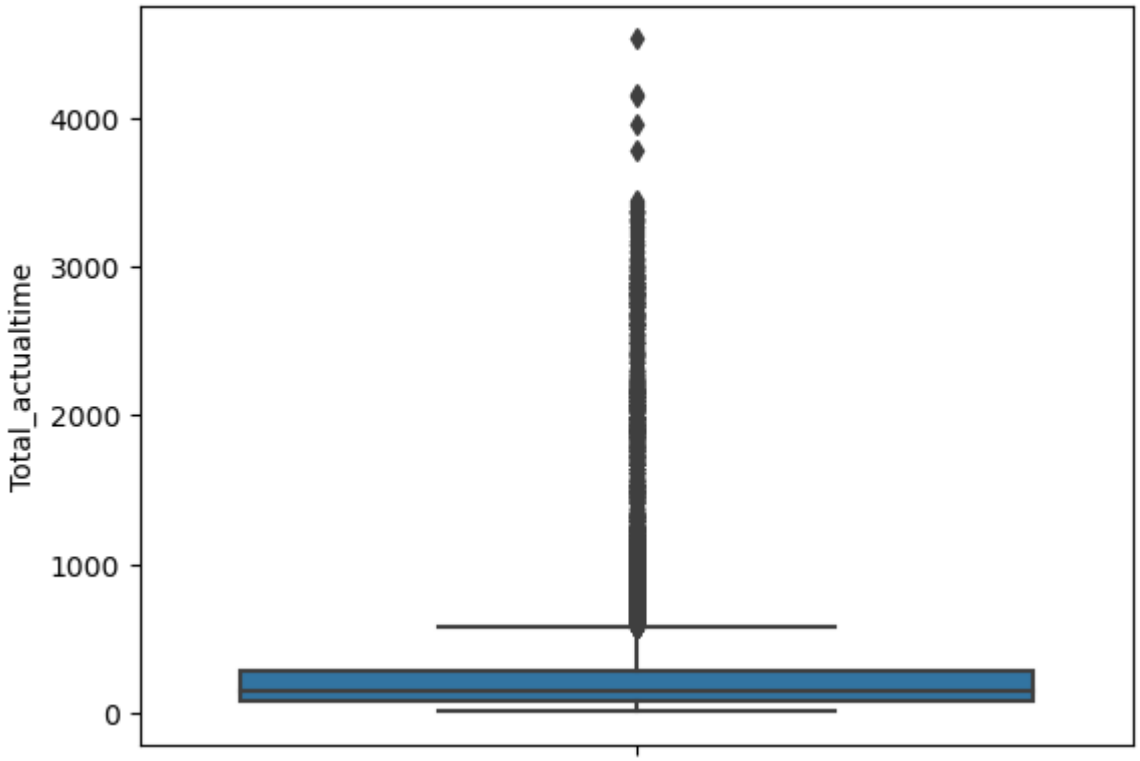
Total_actualtime

In [288…
```
Q1=z["Total_actualtime"].quantile(0.25)
Q3=z["Total_actualtime"].quantile(0.75)
```

```
IQR=Q3-Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR


print(f"lower bound = {lower}")
print(f"Upper bound = {upper}")
```

```
lower bound = -221.0
Upper bound = 579.0
```

In [289…
```
fig = sns.boxplot(y=z["Total_actualtime"])
```



In [291…
```
out1=z[(z["Total_actualtime"]> upper)| (z["Total_actualtime"]<lower)]
out1.shape[0]
```

Out[291]:
```
3054
```

There are around 3054 outlier values in Total_actualtime which could be due to wrong entry and the distribution is not uniform and is skewed.

In [292…
```
#Removing outliers

z[~(z["Total_actualtime"]> upper)| (z["Total_actualtime"]<lower)].reset_index(drop=Tru
```

Out[292]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_tin
0	trip-153671042288605164	96.0	42.0	141.0	65
1	trip-153671042288605164	96.0	42.0	141.0	65
2	trip-153671046011330457	59.0	15.0	59.0	16
3	trip-153671052974046625	147.0	46.0	340.0	115
4	trip-153671052974046625	147.0	46.0	340.0	115
...	...	...	...	...	
23310	trip-153861115439069069	90.0	50.0	258.0	221
23311	trip-153861115439069069	90.0	50.0	258.0	221
23312	trip-153861115439069069	90.0	50.0	258.0	221
23313	trip-153861118270144424	233.0	42.0	274.0	67
23314	trip-153861118270144424	233.0	42.0	274.0	67

23315 rows × 24 columns

Total_osrmtime

In [293...

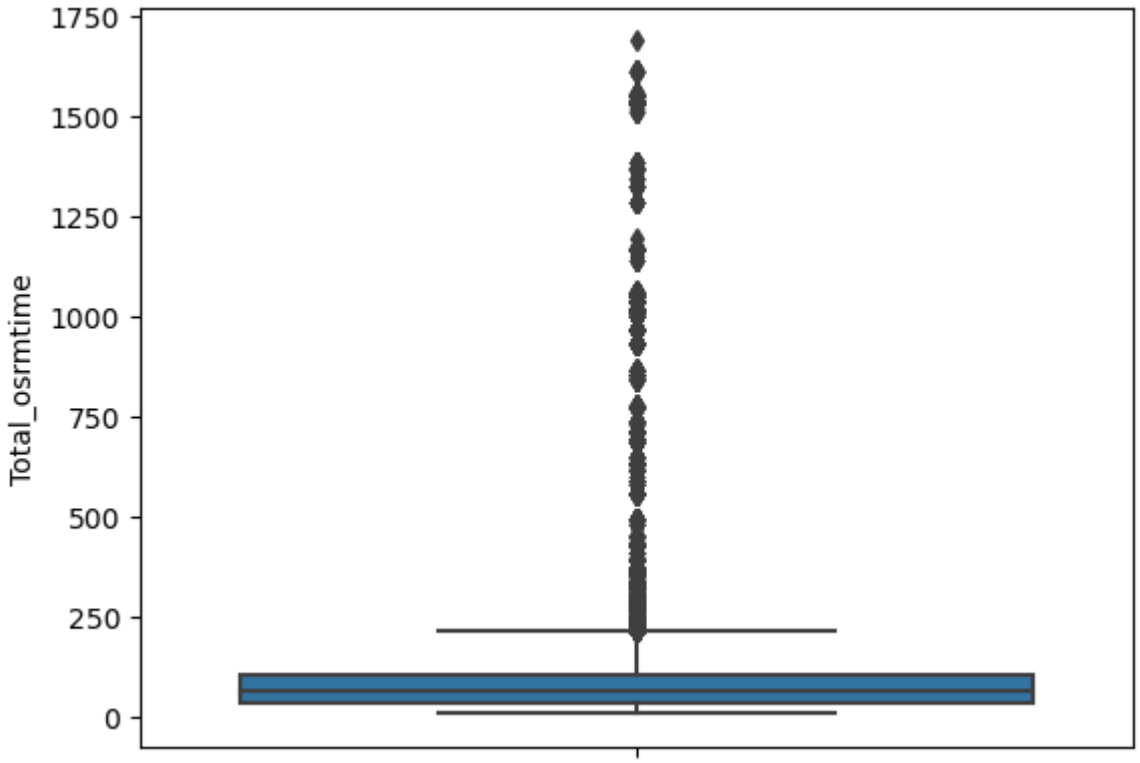
```
Q1=z["Total_osrmtime"].quantile(0.25)
Q3=z["Total_osrmtime"].quantile(0.75)

IQR=Q3-Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR
```

```
print(f"lower bound = {lower}")
print(f"Upper bound = {upper}")
```

```
lower bound = -72.5
Upper bound = 211.5
```

In [197…

```
fig = sns.boxplot(y=z["Total_osrmtime"])
```



In [294…

```
out2=z[(z["Total_osrmtime"]> upper)| (z["Total_osrmtime"]<lower)]
out2.shape[0]
```

Out[294]:

```
2944
```

In [201…

```
#Removing outliers

z[~(z["Total_osrmtime"]> upper)| (z["Total_osrmtime"]<lower)].reset_index(drop=True)
```

Out[201]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_tin
0	trip-153671042288605164	96.0	42.0	141.0	65
1	trip-153671042288605164	96.0	42.0	141.0	65
2	trip-153671046011330457	59.0	15.0	59.0	16
3	trip-153671052974046625	147.0	46.0	340.0	115
4	trip-153671052974046625	147.0	46.0	340.0	115
...	...	...	...	...	
23420	trip-153861115439069069	90.0	50.0	258.0	221
23421	trip-153861115439069069	90.0	50.0	258.0	221
23422	trip-153861115439069069	90.0	50.0	258.0	221
23423	trip-153861118270144424	233.0	42.0	274.0	67
23424	trip-153861118270144424	233.0	42.0	274.0	67

23425 rows × 23 columns

Total_seg_actualtime

In [302…

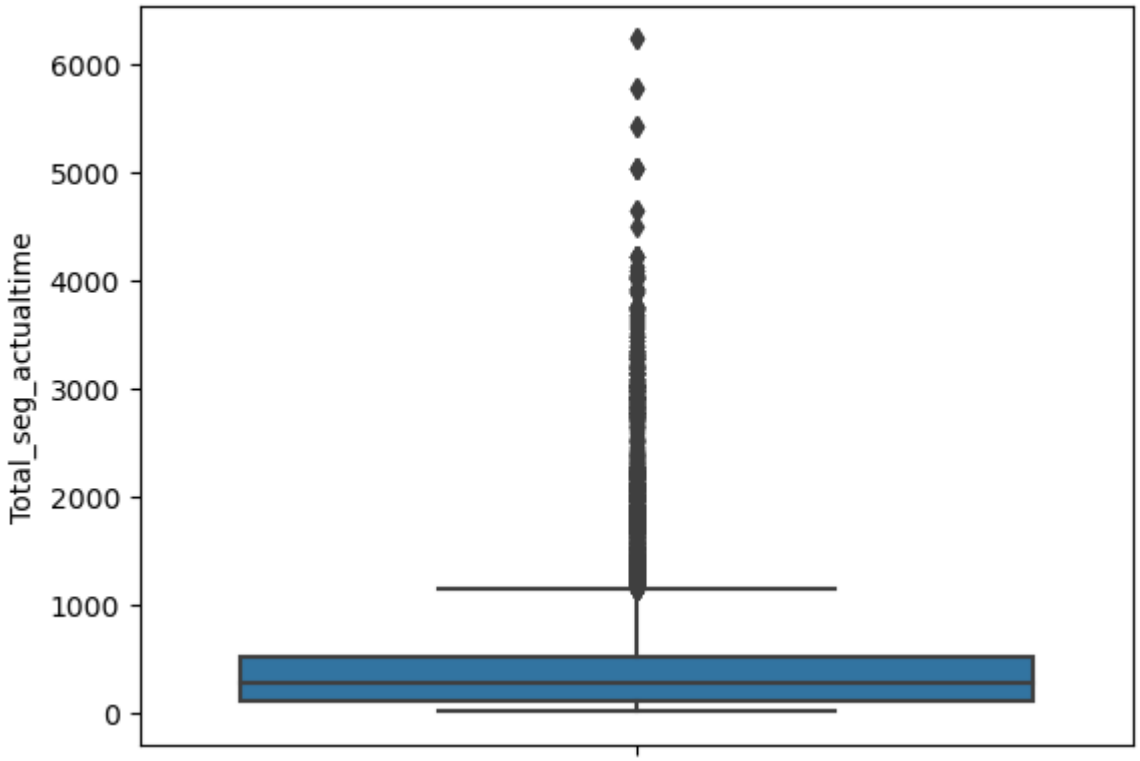
```
Q1=z["Total_seg_actualtime"].quantile(0.25)
Q3=z["Total_seg_actualtime"].quantile(0.75)

IQR=Q3-Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR
```

```
print(f"lower bound = {lower}")
print(f"Upper bound = {upper}")
```

```
lower bound = -517.5
Upper bound = 1142.5
```

In [203…

```
fig = sns.boxplot(y=z["Total_seg_actualtime"])
```



In [303…

```
out3=z[(z["Total_seg_actualtime"]> upper)| (z["Total_seg_actualtime"]<lower)]
out3.shape[0]
```

Out[303]:

```
2010
```

In [206…

```
#Removing outliers

z[~(z["Total_seg_actualtime"]> upper)| (z["Total_seg_actualtime"]<lower)].reset_index(
```

Out[206]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_tin
0	trip-153671042288605164	96.0	42.0	141.0	65
1	trip-153671042288605164	96.0	42.0	141.0	65
2	trip-153671046011330457	59.0	15.0	59.0	16
3	trip-153671052974046625	147.0	46.0	340.0	115
4	trip-153671052974046625	147.0	46.0	340.0	115
...	...	...	...	...	
24354	trip-153861115439069069	90.0	50.0	258.0	221
24355	trip-153861115439069069	90.0	50.0	258.0	221
24356	trip-153861115439069069	90.0	50.0	258.0	221
24357	trip-153861118270144424	233.0	42.0	274.0	67
24358	trip-153861118270144424	233.0	42.0	274.0	67

24359 rows × 23 columns

Total_seg_osrm_time

In [310…

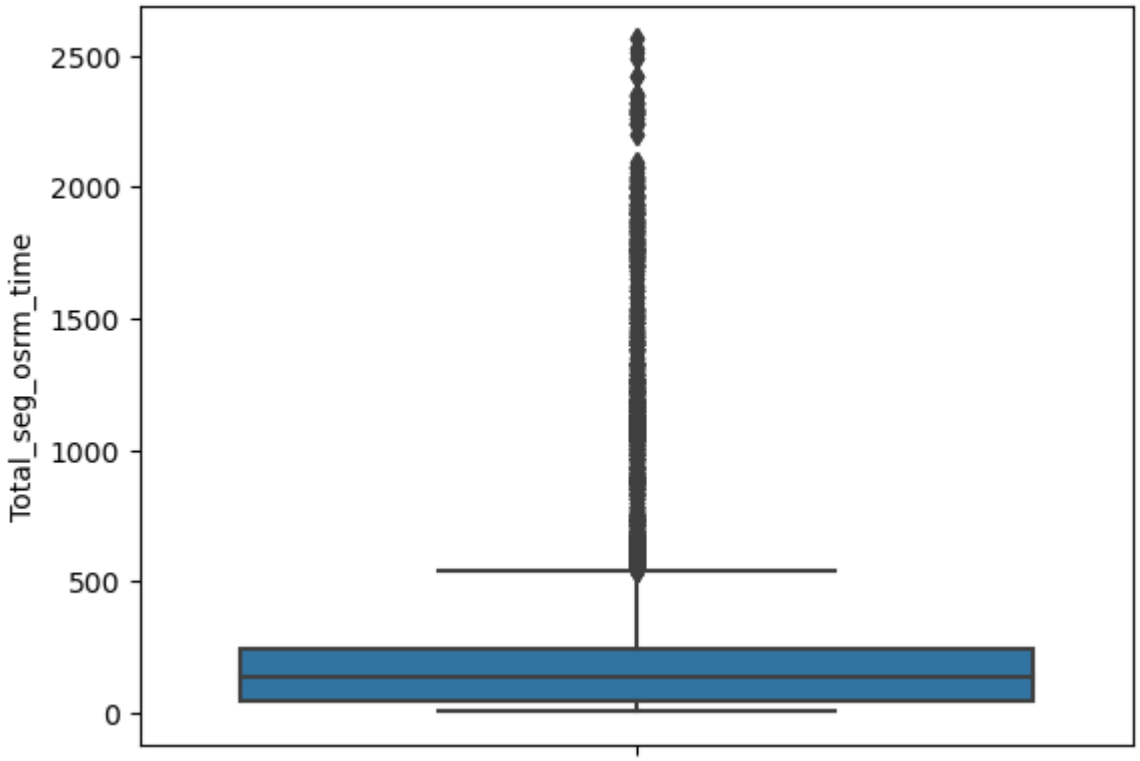
```
Q1=z["Total_seg_osrm_time"].quantile(0.25)
Q3=z["Total_seg_osrm_time"].quantile(0.75)

IQR=Q3-Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR
```

```
print(f"lower bound = {lower}")
print(f"Upper bound = {upper}")
```

```
lower bound = -245.0
Upper bound = 539.0
```

In [208…

```
fig = sns.boxplot(y=z["Total_seg_osrm_time"])
```



In [311…

```
out4=z[(z["Total_seg_osrm_time"]> upper)| (z["Total_seg_osrm_time"]<lower)]
out4.shape[0]
```

Out[311]:

```
2062
```

In [210…

```
#Removing outliers

z[~(z["Total_seg_osrm_time"]> upper)| (z["Total_seg_osrm_time"]<lower)].reset_index(dr
```

Out[210]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_tin
0	trip-153671042288605164	96.0	42.0	141.0	65
1	trip-153671042288605164	96.0	42.0	141.0	65
2	trip-153671046011330457	59.0	15.0	59.0	16
3	trip-153671052974046625	147.0	46.0	340.0	115
4	trip-153671052974046625	147.0	46.0	340.0	115
...	...	...	...	...	
24302	trip-153861115439069069	90.0	50.0	258.0	221
24303	trip-153861115439069069	90.0	50.0	258.0	221
24304	trip-153861115439069069	90.0	50.0	258.0	221
24305	trip-153861118270144424	233.0	42.0	274.0	67
24306	trip-153861118270144424	233.0	42.0	274.0	67

24307 rows × 23 columns

Total_osrm_distance

In [312…

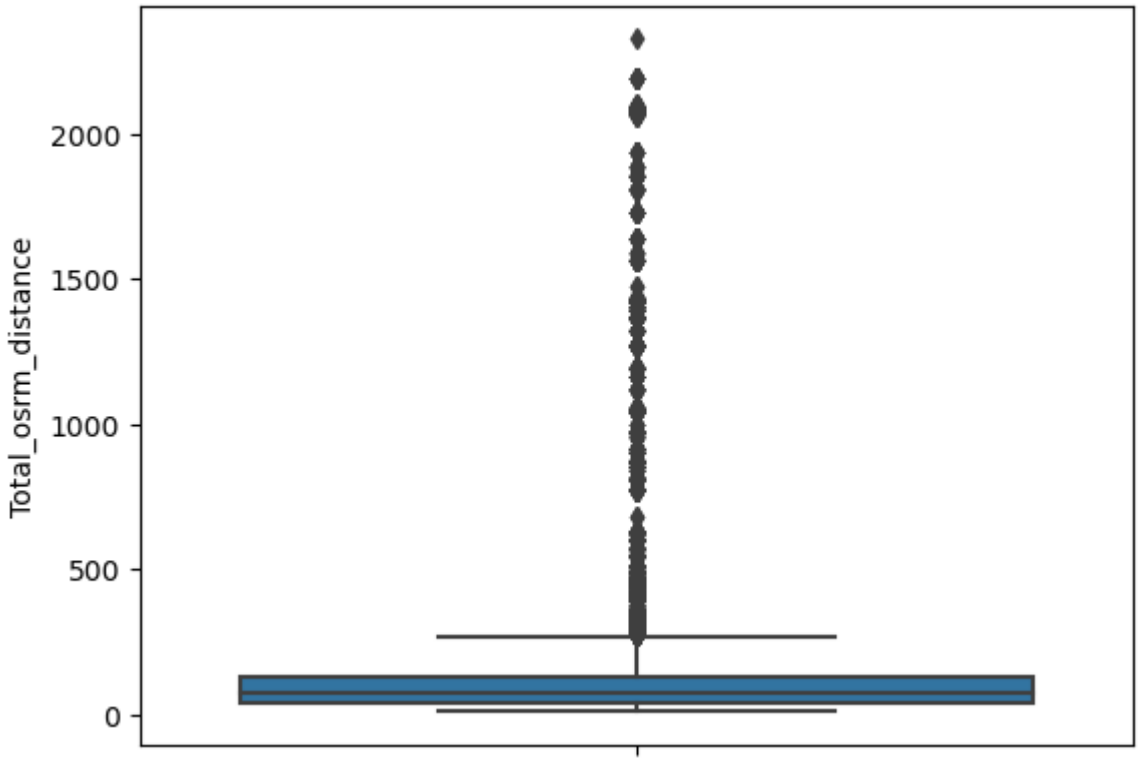
```
Q1=z["Total_osrm_distance"].quantile(0.25)
Q3=z["Total_osrm_distance"].quantile(0.75)

IQR=Q3-Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR
```

```
print(f"lower bound = {lower}")
print(f"Upper bound = {upper}")
```

```
lower bound = -101.95355
Upper bound = 270.19005000000004
```

In [212…

```
fig = sns.boxplot(y=z["Total_osrm_distance"])
```



In [314…

```
out5=z[(z["Total_osrm_distance"]> upper)| (z["Total_osrm_distance"]<lower)]
out5.shape[0]
```

Out[314]:

```
2991
```

In [113…

```
data1.head()
```

Out[113]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_time	T
0	trip-153671041653548748	830.0	394.0	1548.0	1008.0	
1	trip-153671042288605164	96.0	42.0	141.0	65.0	
2	trip-153671043369099517	2736.0	1529.0	3308.0	1941.0	
3	trip-153671046011330457	59.0	15.0	59.0	16.0	
4	trip-153671052974046625	147.0	46.0	340.0	115.0	

In [114…

```
df1.head()
```

Out[114]:

	data	Month	Year	Day	route_schedule_uuid	route_type	trip_uuid	source_cent
0	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AA
1	training	9	2018	20	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388620AA
2	training	9	2018	23	thanos::sroute:ff52ef7a-4d0d-4063-9bfe-cc21172...	FTL	trip-153768492602129387	IND421302AA
3	training	9	2018	14	thanos::sroute:a16bfa03-3462-4bce-9c82-5784c7d...	Carting	trip-153693976643699843	IND400011AA
4	training	9	2018	13	thanos::sroute:76951383-1608-44e4-a284-46d92e8...	FTL	trip-153687145942424248	IND562132AA

In [115…

```
z.head()
```

Out[115]:

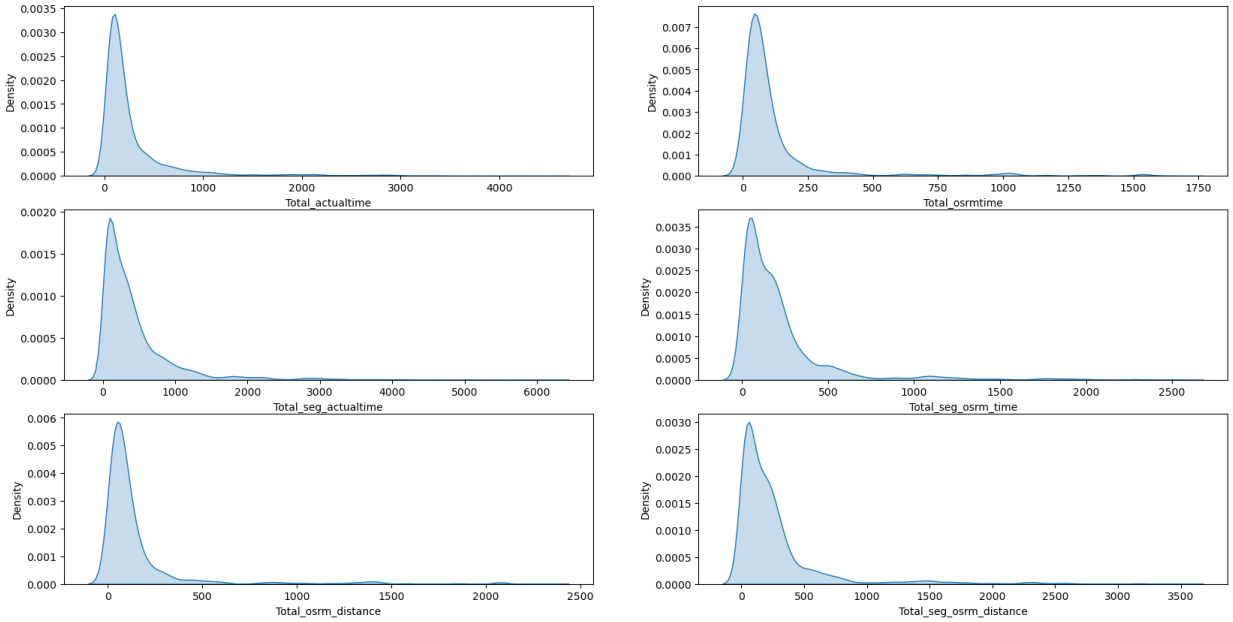
	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_time	T
0	trip-153671041653548748	830.0	394.0	1548.0	1008.0	
1	trip-153671041653548748	830.0	394.0	1548.0	1008.0	
2	trip-153671042288605164	96.0	42.0	141.0	65.0	
3	trip-153671042288605164	96.0	42.0	141.0	65.0	
4	trip-153671043369099517	2736.0	1529.0	3308.0	1941.0	

DISTRIBUTION PLOT OF ALL CONTINUOUS CONTINUOUS VARIABLES

In [134…

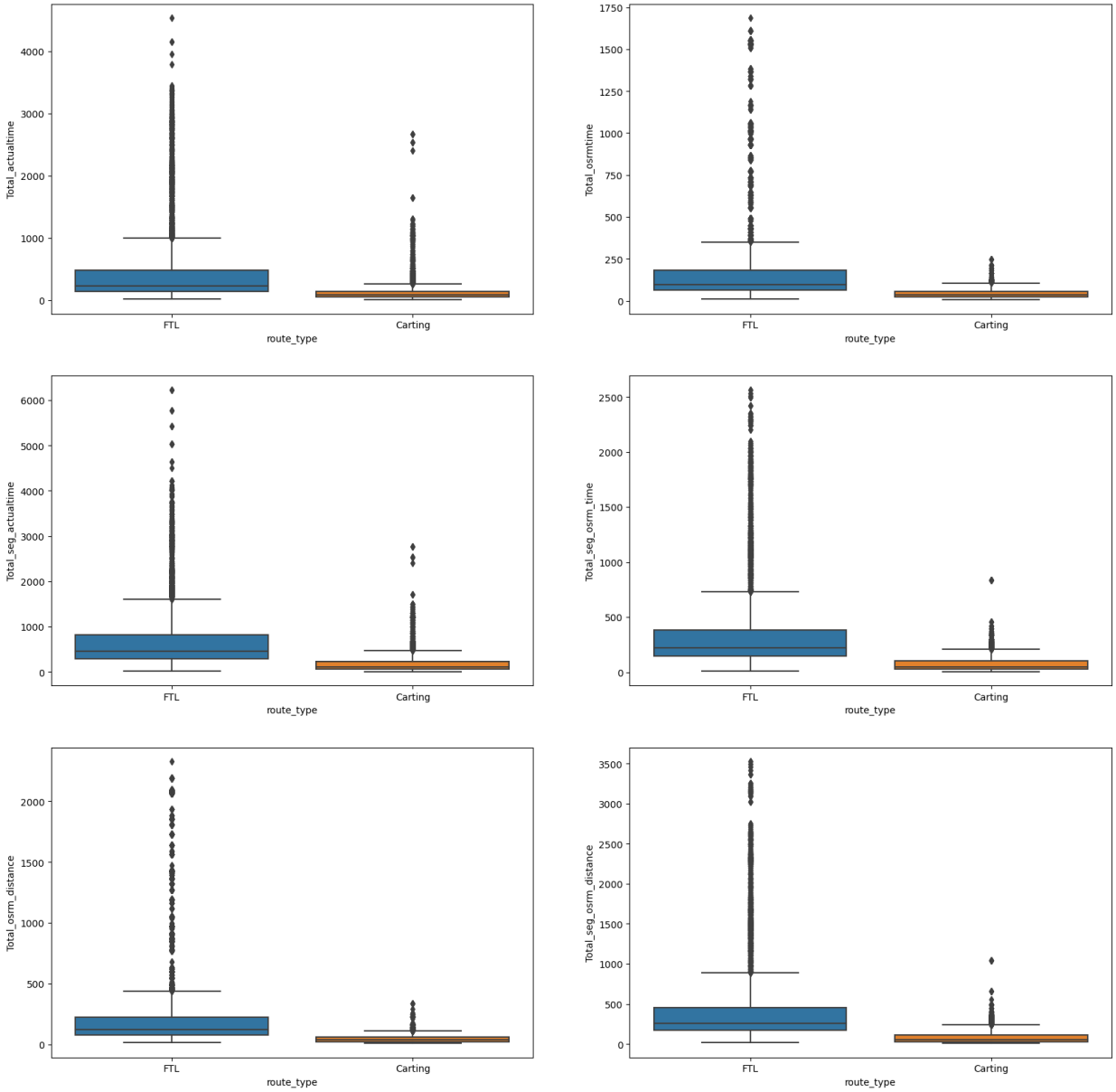
```
fig,axs=plt.subplots(nrows=3,ncols=2,figsize=(20,10))
cols=["Total_actualtime","Total_osrmtime","Total_seg_actualtime","Total_seg_osrm_time"
count=0
for i in range(3):
    for j in range(2):
        sns.kdeplot(data=z[cols[count]], ax=axs[i, j],fill=True)
```

```
        count+=1
plt.show()
```



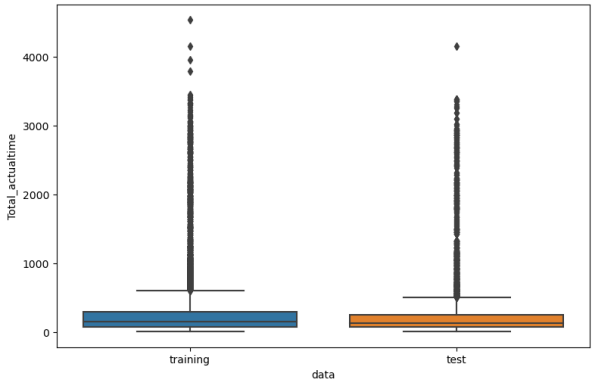
In [139…

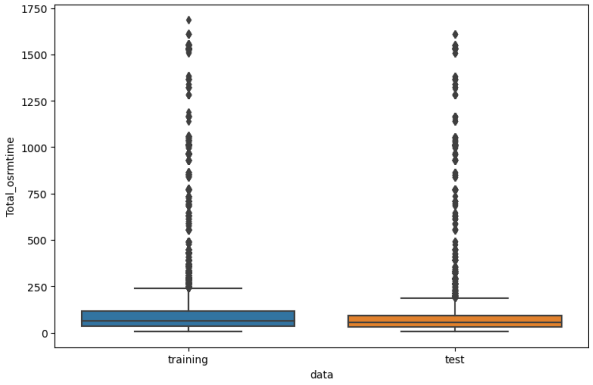
```
fig,axs=plt.subplots(nrows=3,ncols=2,figsize=(20,20))

int_cols=["Total_actualtime","Total_osrmtime","Total_seg_actualtime","Total_seg_osrm_t
          "Total_seg_osrm_distance"]
count=0

for i in range(3):
    for j in range(2):
        sns.boxplot(x=z["route_type"],y=z[int_cols[count]],ax=axs[i,j])
        count+=1
plt.show()
```

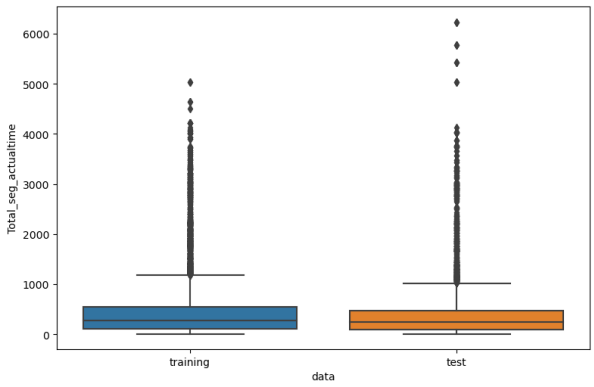


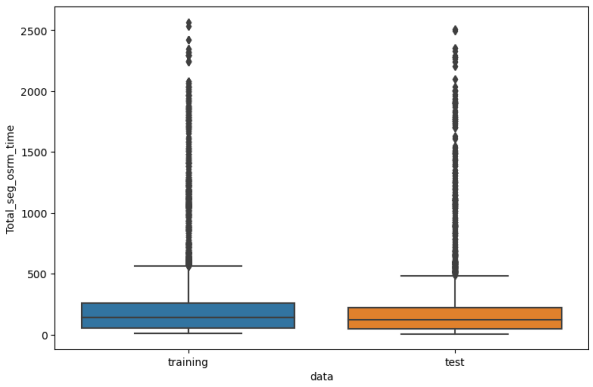
In [280…

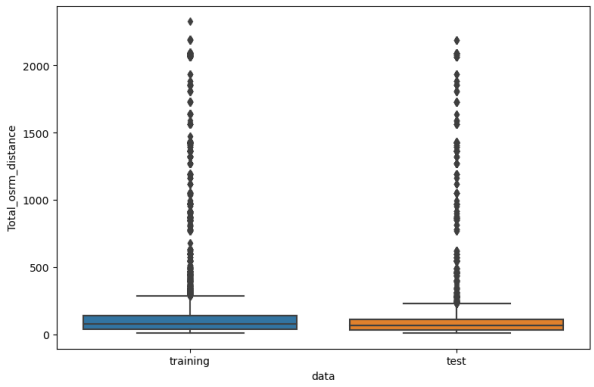
```
fig,axs=plt.subplots(nrows=3,ncols=2,figsize=(20,20))

int_cols=["Total_actualtime","Total_osrmtime","Total_seg_actualtime","Total_seg_osrm_t
          "Total_seg_osrm_distance"]
count=0

for i in range(3):
    for j in range(2):
        sns.boxplot(x=z["data"],y=z[int_cols[count]],ax=axs[i,j])
        count+=1
plt.show()
```

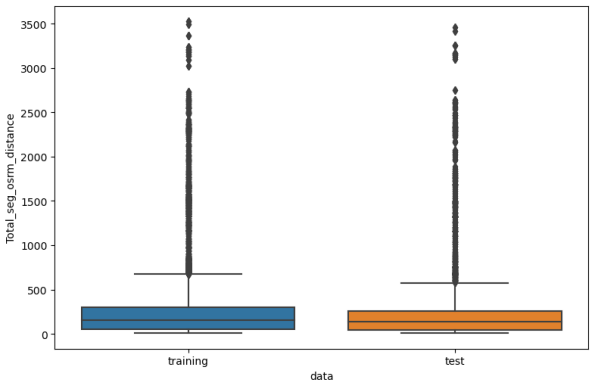








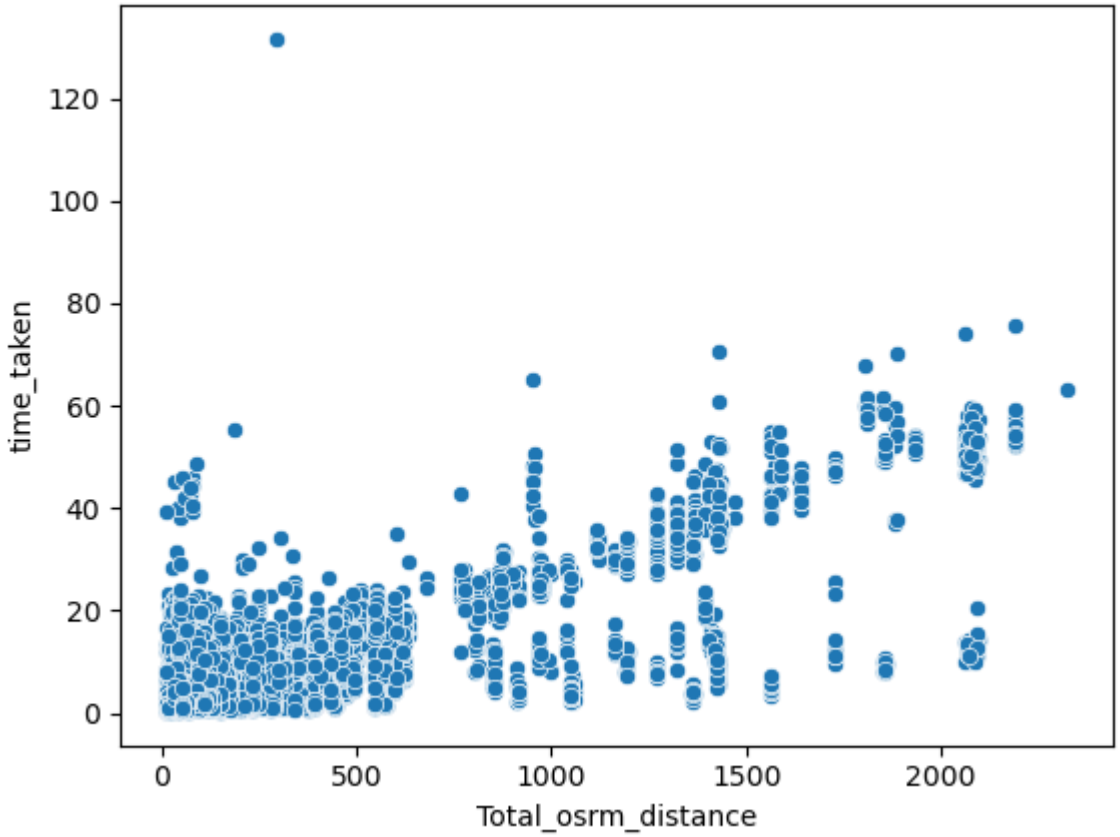




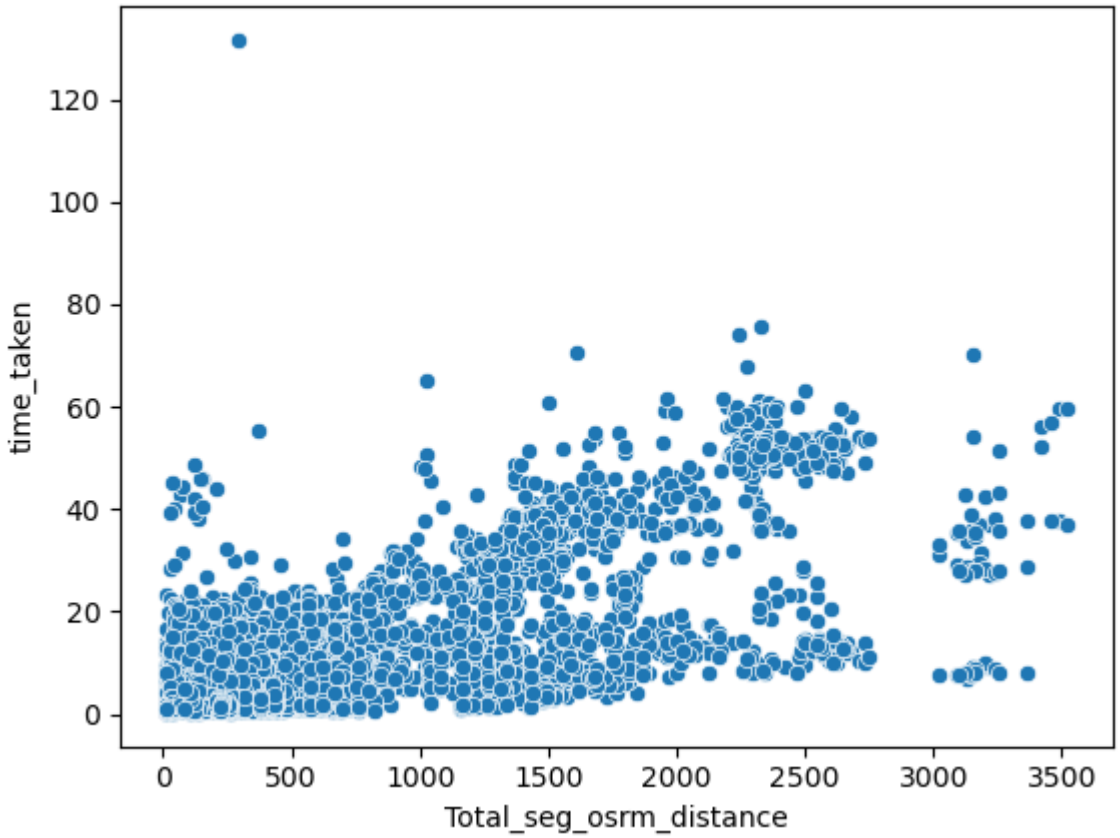
DISTANCE & TIME RELATION

In [147…
```
sns.scatterplot(x=z["Total_osrm_distance"],y=z["time_taken"])
```

Out[147]:
```
<Axes: xlabel='Total_osrm_distance', ylabel='time_taken'>
```

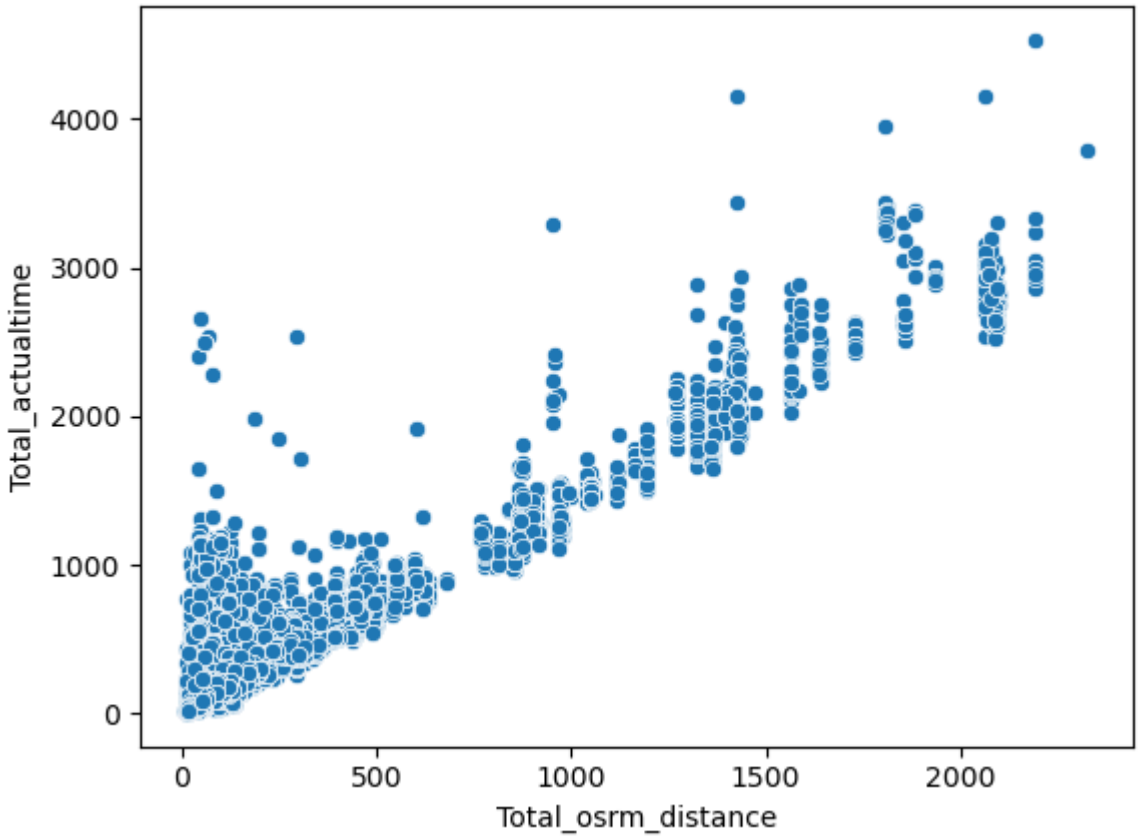


In [146…
```
sns.scatterplot(x=z["Total_seg_osrm_distance"], y=z["time_taken"], data=z);
```



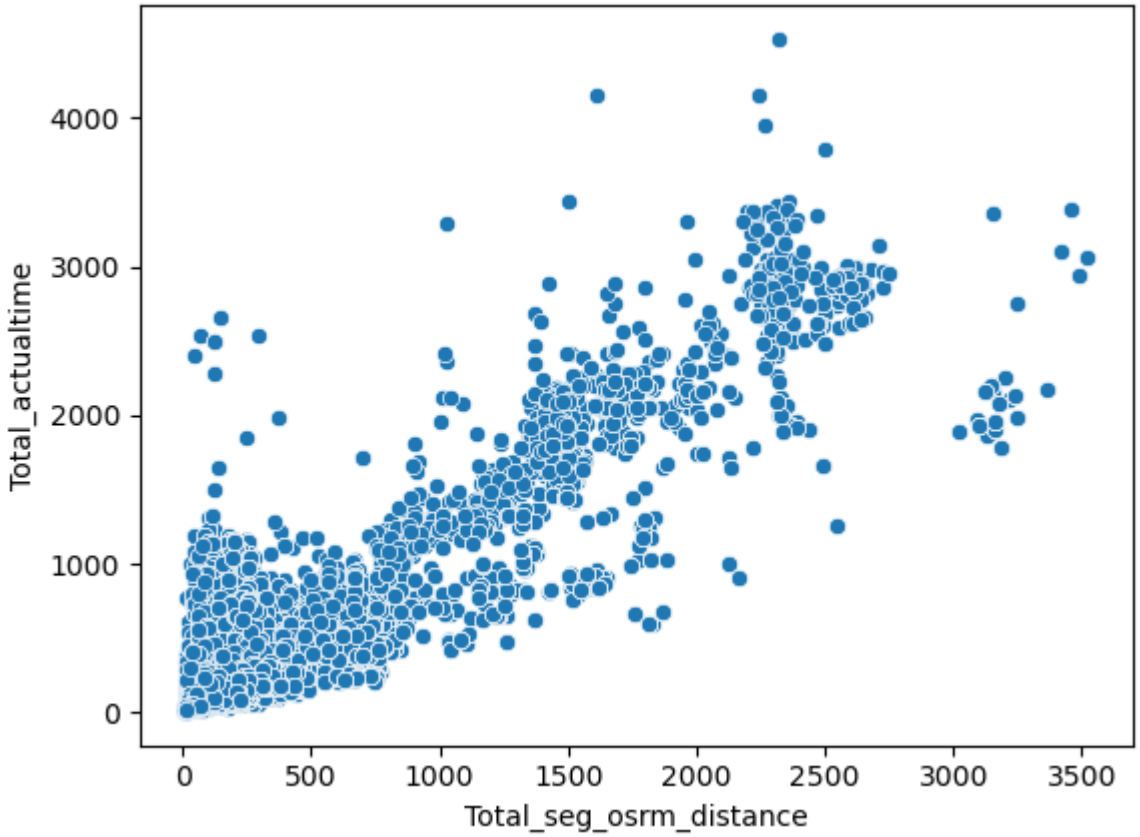
In [149…
```
sns.scatterplot(x=z["Total_osrm_distance"],y=z["Total_actualtime"])
```

Out[149]: <Axes: xlabel='Total_osrm_distance', ylabel='Total_actualtime'>



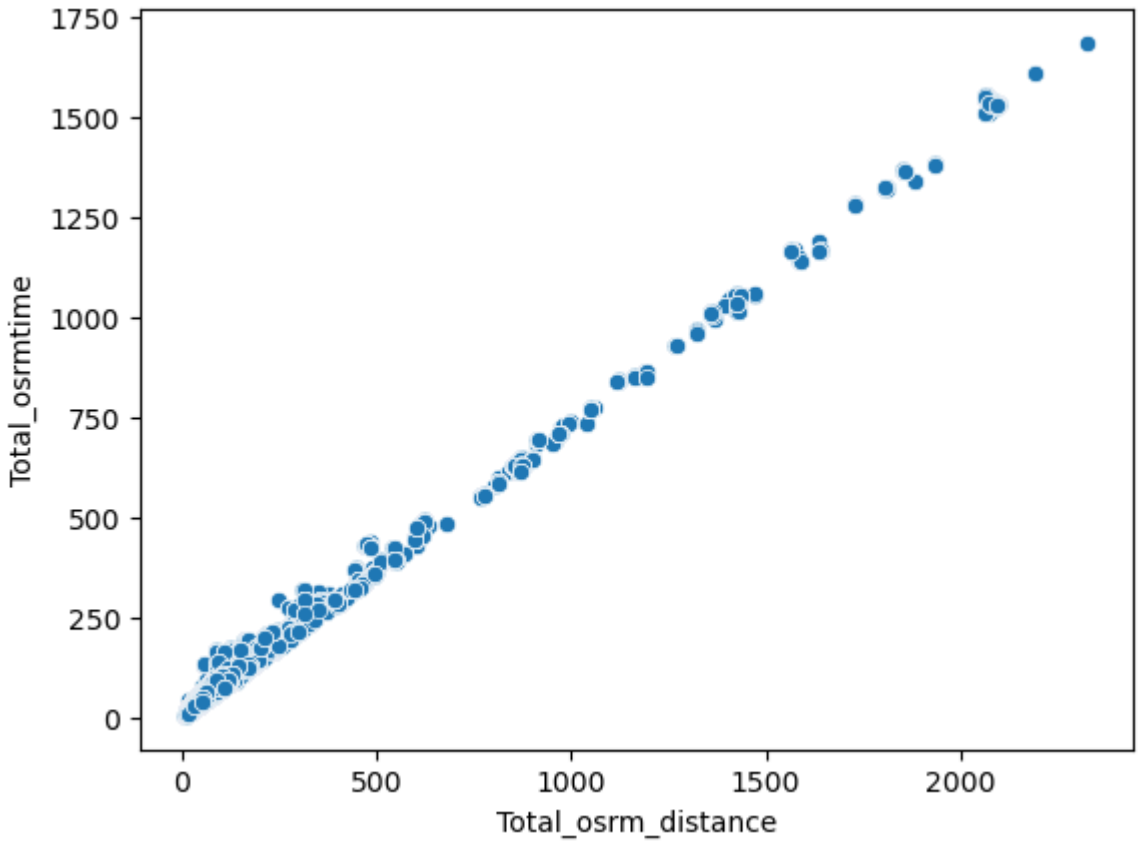
In [150…
```
sns.scatterplot(x=z["Total_seg_osrm_distance"],y=z["Total_actualtime"])
```

Out[150]: <Axes: xlabel='Total_seg_osrm_distance', ylabel='Total_actualtime'>



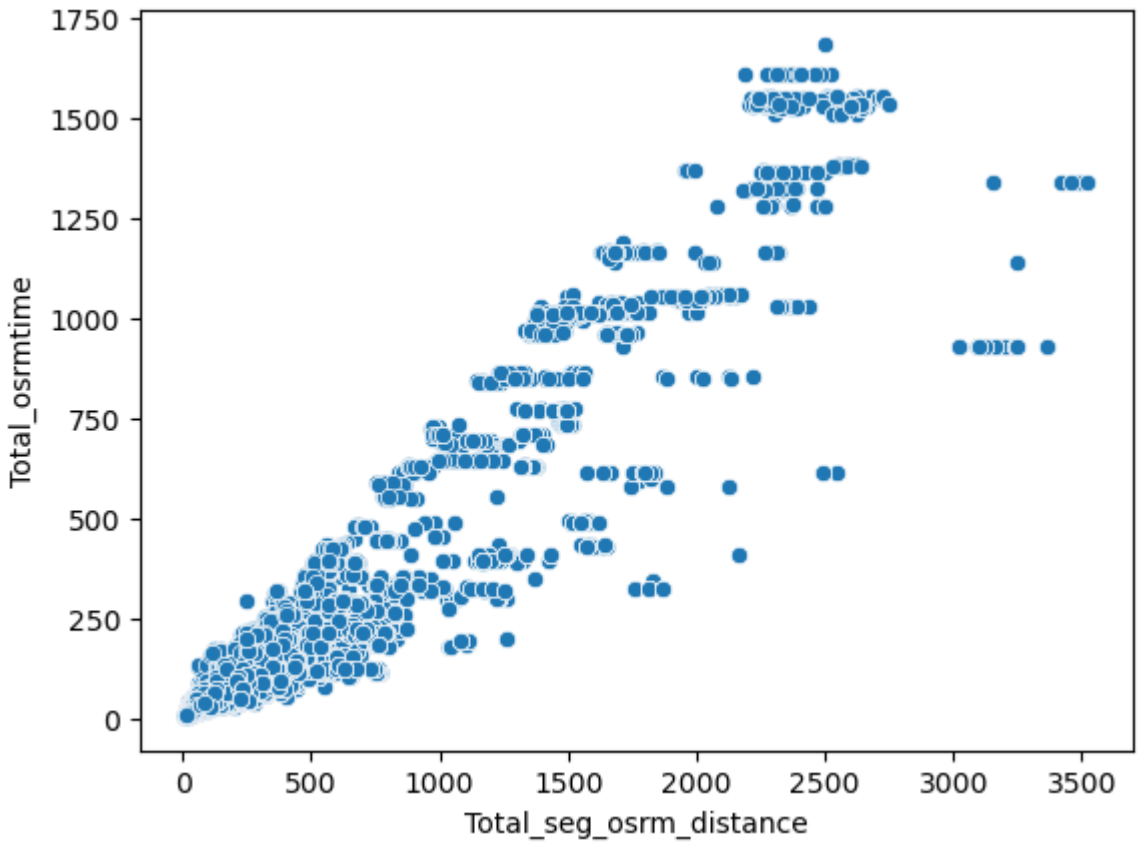
In [151…
```
sns.scatterplot(x=z["Total_osrm_distance"],y=z["Total_osrmtime"])
```

Out[151]:
```
<Axes: xlabel='Total_osrm_distance', ylabel='Total_osrmtime'>
```



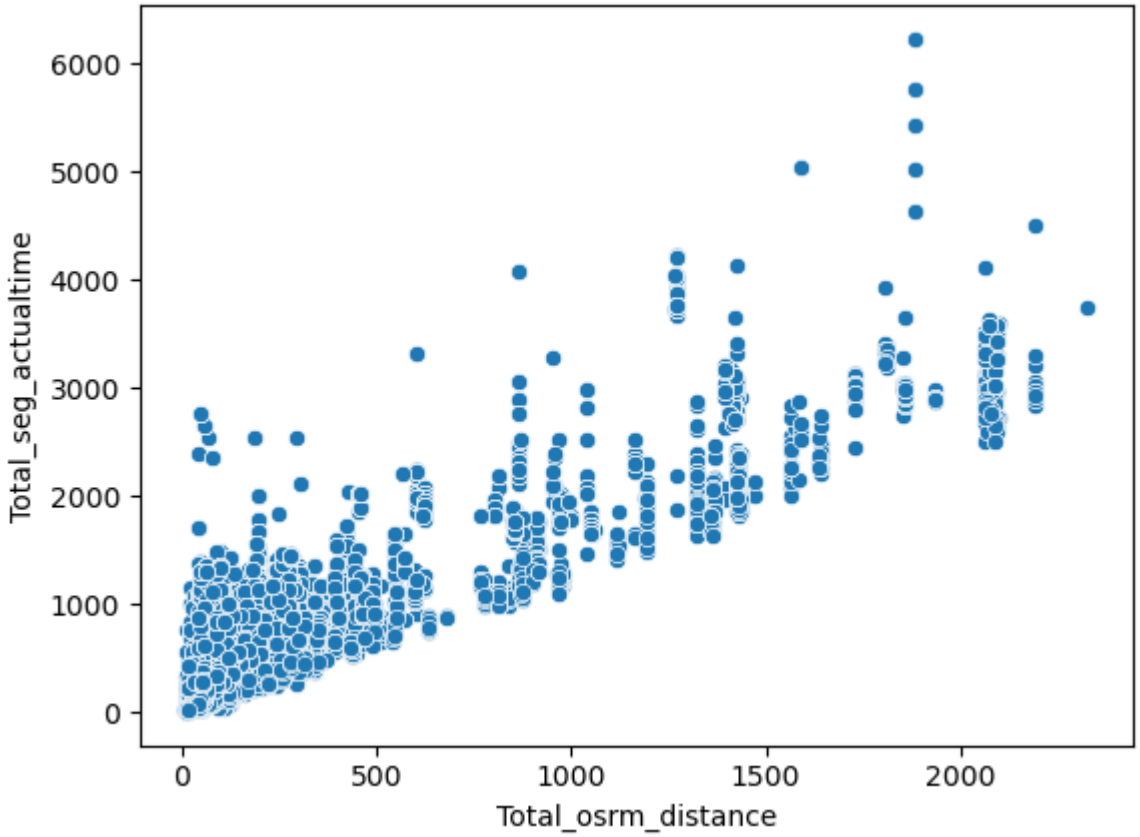
In [152…
```
sns.scatterplot(x=z["Total_seg_osrm_distance"],y=z["Total_osrmtime"])
```

Out[152]:
```
<Axes: xlabel='Total_seg_osrm_distance', ylabel='Total_osrmtime'>
```



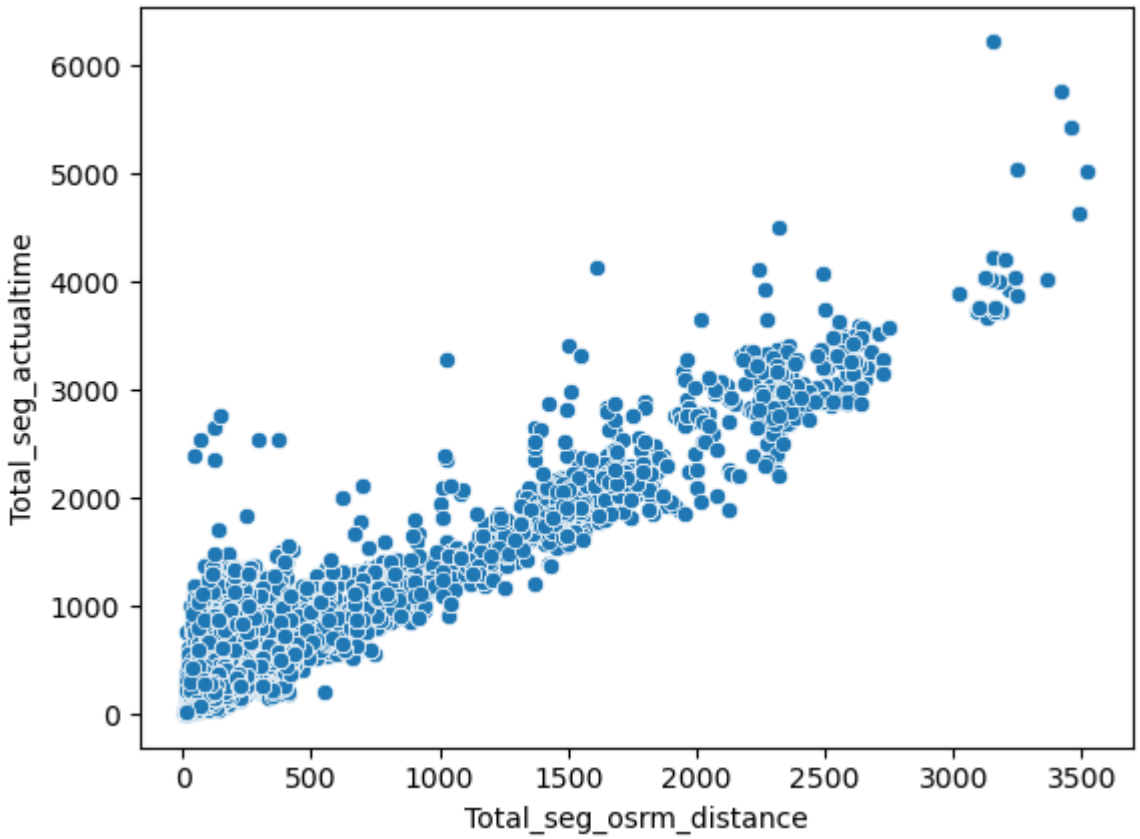
In [153…
```
sns.scatterplot(x=z["Total_osrm_distance"],y=z["Total_seg_actualtime"])
```

Out[153]:
```
<Axes: xlabel='Total_osrm_distance', ylabel='Total_seg_actualtime'>
```



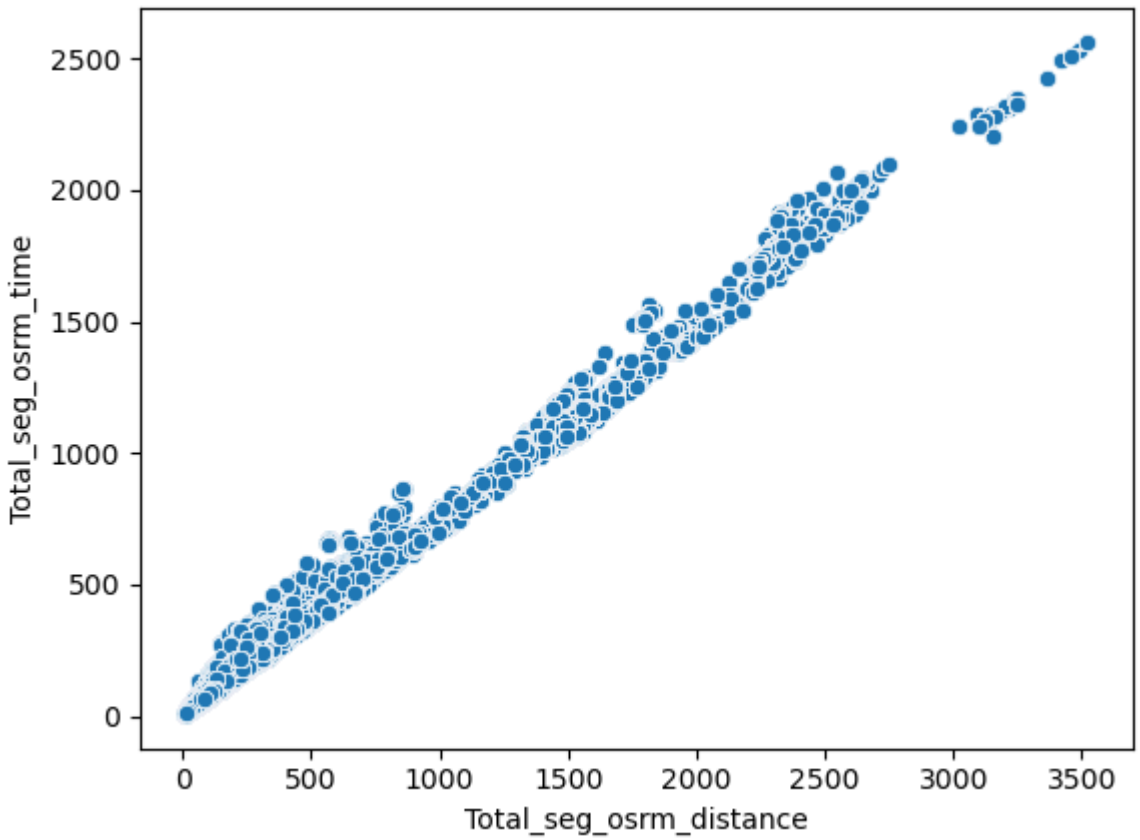
In [154…
```
sns.scatterplot(x=z["Total_seg_osrm_distance"],y=z["Total_seg_actualtime"])
```

Out[154]:
```
<Axes: xlabel='Total_seg_osrm_distance', ylabel='Total_seg_actualtime'>
```



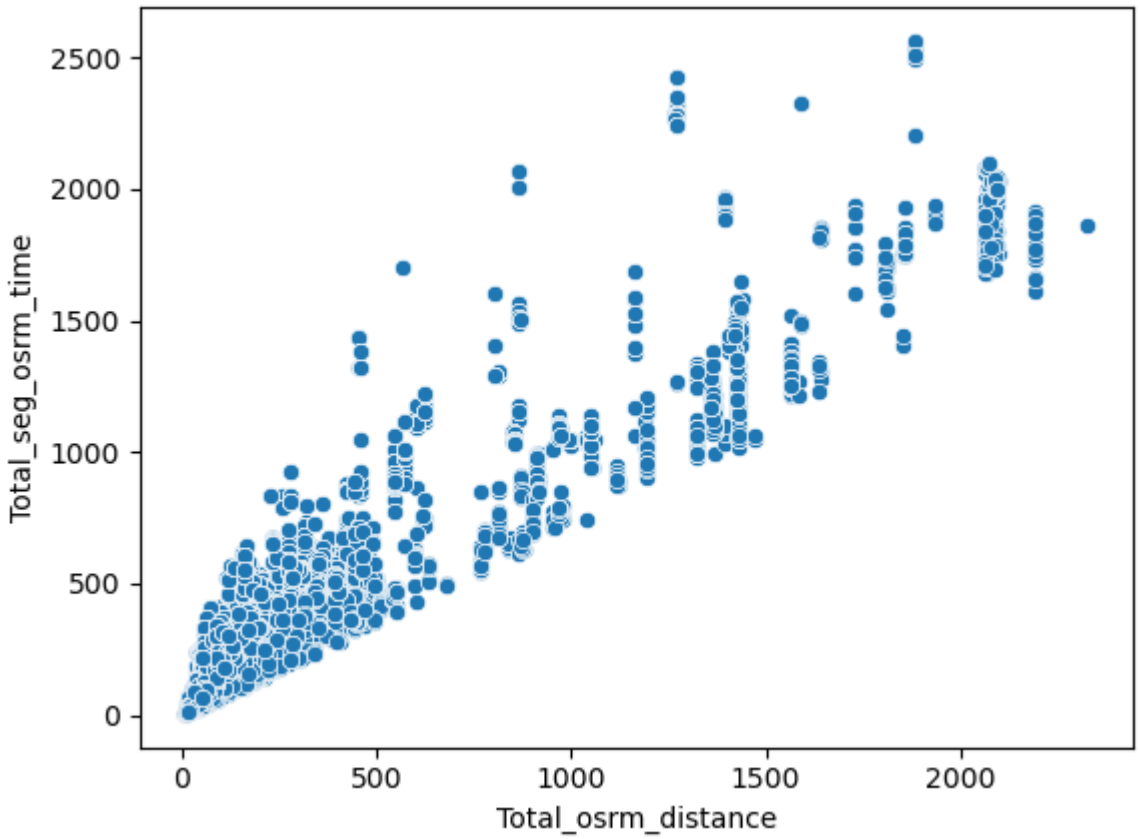
In [73]:
```
sns.scatterplot(x=z["Total_seg_osrm_distance"],y=z["Total_seg_osrm_time"])
```

Out[73]:
```
<Axes: xlabel='Total_seg_osrm_distance', ylabel='Total_seg_osrm_time'>
```



In [156…
```
sns.scatterplot(x=z["Total_osrm_distance"],y=z["Total_seg_osrm_time"])
```

Out[156]:
```
<Axes: xlabel='Total_osrm_distance', ylabel='Total_seg_osrm_time'>
```



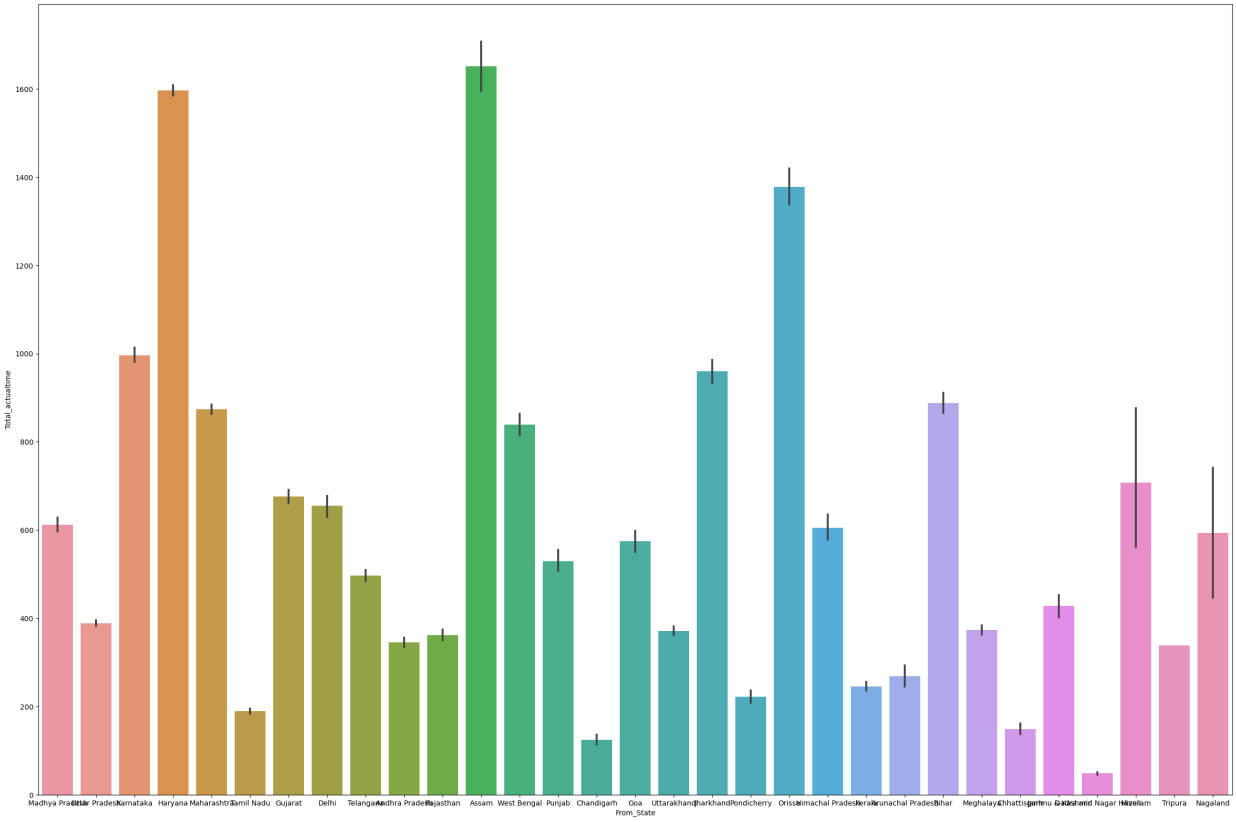
In [281…
```
z.head()
```

Out[281]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_time	T
0	trip-153671041653548748	830.0	394.0	1548.0	1008.0	
1	trip-153671041653548748	830.0	394.0	1548.0	1008.0	
2	trip-153671042288605164	96.0	42.0	141.0	65.0	
3	trip-153671042288605164	96.0	42.0	141.0	65.0	
4	trip-153671043369099517	2736.0	1529.0	3308.0	1941.0	

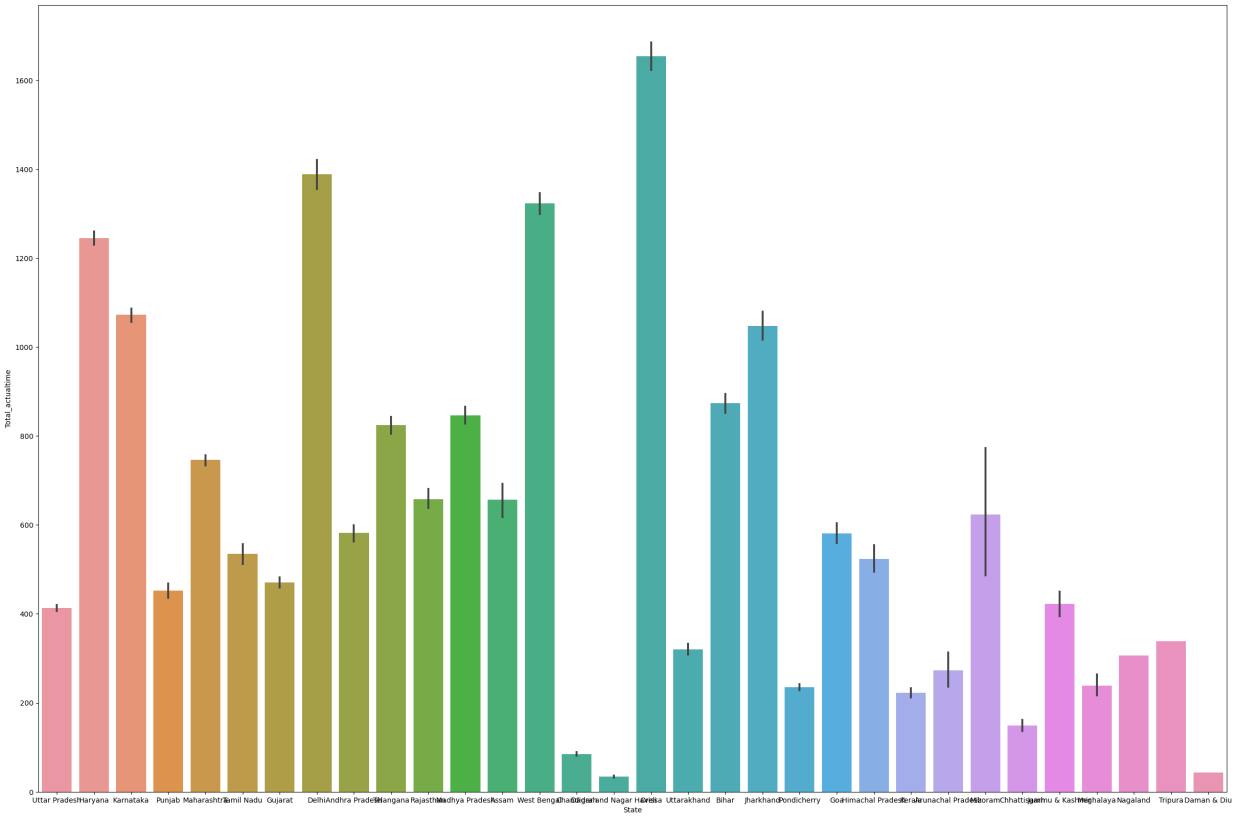
In [74]:
```
plt.figure(figsize=(30,20))
sns.barplot(data=z,x="From_State",y="Total_actualtime")
```

Out[74]:
```
<Axes: xlabel='From_State', ylabel='Total_actualtime'>
```



In [75]:
```
plt.figure(figsize=(30,20))
sns.barplot(data=z,x="State",y="Total_actualtime")
```

Out[75]:
```
<Axes: xlabel='State', ylabel='Total_actualtime'>
```



In [318...
```
plt.figure(figsize=(20,10))
```

```
sns.heatmap(data=data1.corr(method="pearson"),annot=True)
```

```
C:\Users\91944\AppData\Local\Temp\ipykernel_18296\27479017.py:3: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.
  sns.heatmap(data=data1.corr(method="pearson"),annot=True)
```

Out[318]: <Axes: >

Handling categorical values

In [320…

```
from sklearn.preprocessing import OneHotEncoder

# Converting type of columns to category
z['data'] = z['data'].astype('category')
z['route_type'] = z['route_type'].astype('category')


# Assigning numerical values and storing it in another columns
z['data_new'] = z['data'].cat.codes
z['route_type_new'] = z['route_type'].cat.codes


# Create an instance of One-hot-encoder
enc = OneHotEncoder()

# Passing encoded columns

enc_data = pd.DataFrame(enc.fit_transform(
    z[['data_new', 'route_type_new']]).toarray())

# Merge with main
New_df = z.join(enc_data)

New_df
```

Out[320]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_tin
0	trip-153671041653548748	830.0	394.0	1548.0	1008
1	trip-153671041653548748	830.0	394.0	1548.0	1008
2	trip-153671042288605164	96.0	42.0	141.0	65
3	trip-153671042288605164	96.0	42.0	141.0	65
4	trip-153671043369099517	2736.0	1529.0	3308.0	1941
...	...	...	...	...	
26364	trip-153861115439069069	90.0	50.0	258.0	221
26365	trip-153861115439069069	90.0	50.0	258.0	221
26366	trip-153861115439069069	90.0	50.0	258.0	221
26367	trip-153861118270144424	233.0	42.0	274.0	67
26368	trip-153861118270144424	233.0	42.0	274.0	67

26369 rows × 30 columns

Column Normalization /Column Standardization

In [334…

```
# data normalization with sklearn
from sklearn.preprocessing import MinMaxScaler


col_to_norm=["Total_actualtime","Total_osrmtime","Total_seg_actualtime","Total_seg_osr
# fit scaler on training data
norm = MinMaxScaler()
```

```
for i in col_to_norm:
#     z[i].reshape(-1,1)
    z[i]=norm.fit_transform(z[i].values.reshape(-1,1))
z.head()
```

Out[334]:

	trip_uuid	Total_actualtime	Total_osrmtime	Total_seg_actualtime	Total_seg_osrm_time	T
0	trip-153671041653548748	0.181517	0.230952	0.247388	0.391712	
1	trip-153671041653548748	0.181517	0.230952	0.247388	0.391712	
2	trip-153671042288605164	0.019235	0.021429	0.021218	0.023065	
3	trip-153671042288605164	0.019235	0.021429	0.021218	0.023065	
4	trip-153671043369099517	0.602918	0.906548	0.530301	0.756450	

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: