Business Case: Delhivery - Feature Engineering

About Delhivery: Delhivery is the largest and fastest-growing fully integrated player in India by revenue in Fiscal 2021. They aim to build the operating system for commerce, through a combination of world-class infrastructure, logistics operations of the highest quality, and cutting-edge engineering and technology capabilities.

Problem Statement: Delhivery wants to process raw data from their data pipelines to support forecasting models. This involves:

- Cleaning and manipulating the raw data.
- Extracting useful features for model-building and business insight.
- Aggregating multiple rows of trip data effectively.

```
In [142… !wget https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/551/origin
```

```
--2024-09-26 15:02:20--  https://d2beiqkhq929f0.cloudfront.net/asset
s/000/001/551/original/delhivery_data.csv
Resolving d2beiqkhq929f0.cloudfront.net (d2beiqkhq929f0.cloudfront.net)... 13.224.
9.181, 13.224.9.129, 13.224.9.103, ...
Connecting to d2beiqkhq929f0.cloudfront.net (d2beiqkhq929f0.cloudfront.net)|13.22
4.9.181|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 55617130 (53M) [text/plain]
Saving to: 'delhivery_data.csv.16'

delhivery_data.csv. 100%[===================>]  53.04M   169MB/s    in 0.3s

2024-09-26 15:02:20 (169 MB/s) - 'delhivery_data.csv.16' saved [55617130/55617130]
```

```python
In [142…
# importing the libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as spy

from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler, MinMaxScaler, LabelEncoder


import warnings
warnings.filterwarnings('ignore')
```

```python
In [142…
# Reading the dataset
df = pd.read_csv('delhivery_data.csv')
```

Analyzing the data

```python
In [143… df.shape
```

```
Out[1430]:  (144867, 24)
```

In [143… `df.head()`

Out[1431]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_c…
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812

5 rows × 24 columns

In [143… `df.tail()`

Out[1432]:

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

5 rows × 24 columns

In [143… `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
```

Data Cleaning:

- Checking for Duplicates and if exists handling them
- Identiying Missing Values
- Handling Missing Values
- Dropping unnecessary columns
- Converting the columns into appropriate data types

In [143…
```python
# Checking for Duplicates
df.duplicated().sum()
```

Out[1434]:   0

In [143…
```python
df.nunique() # 144867 rows in total
```

Out[1435]:

	0
data	2
trip_creation_time	14817
route_schedule_uuid	1504
route_type	2
trip_uuid	14817
source_center	1508
source_name	1498
destination_center	1481
destination_name	1468
od_start_time	26369
od_end_time	26369
start_scan_to_end_scan	1915
is_cutoff	2
cutoff_factor	501
cutoff_timestamp	93180
actual_distance_to_destination	144515
actual_time	3182
osrm_time	1531
osrm_distance	138046
factor	45641
segment_actual_time	747
segment_osrm_time	214
segment_osrm_distance	113799
segment_factor	5675

dtype: int64

In [143…

```
# missing value calculation
df.isnull().sum()
```

Out[1436]:

	0
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
is_cutoff	0
cutoff_factor	0
cutoff_timestamp	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
segment_factor	0

dtype: int64

```python
round((df.isnull().sum() / len(df))*100,2) # missing values in percentage
```

Out[1437]:

	0
data	0.00
trip_creation_time	0.00
route_schedule_uuid	0.00
route_type	0.00
trip_uuid	0.00
source_center	0.00
source_name	0.20
destination_center	0.00
destination_name	0.18
od_start_time	0.00
od_end_time	0.00
start_scan_to_end_scan	0.00
is_cutoff	0.00
cutoff_factor	0.00
cutoff_timestamp	0.00
actual_distance_to_destination	0.00
actual_time	0.00
osrm_time	0.00
osrm_distance	0.00
factor	0.00
segment_actual_time	0.00
segment_osrm_time	0.00
segment_osrm_distance	0.00
segment_factor	0.00

dtype: float64

In [143… `df.dropna(inplace=True) # Since the percentage of null is less than 1% we prefer dr`

In [143… `round((df.isnull().sum() / len(df))*100,2) # after removing null values.`

Out[1439]:

	0
data	0.0
trip_creation_time	0.0
route_schedule_uuid	0.0
route_type	0.0
trip_uuid	0.0
source_center	0.0
source_name	0.0
destination_center	0.0
destination_name	0.0
od_start_time	0.0
od_end_time	0.0
start_scan_to_end_scan	0.0
is_cutoff	0.0
cutoff_factor	0.0
cutoff_timestamp	0.0
actual_distance_to_destination	0.0
actual_time	0.0
osrm_time	0.0
osrm_distance	0.0
factor	0.0
segment_actual_time	0.0
segment_osrm_time	0.0
segment_osrm_distance	0.0
segment_factor	0.0

dtype: float64

In [144…
```python
# Data Reduction
# Dropping Unnecessary Columns
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 144316 entries, 0 to 144866
Data columns (total 24 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   data                         144316 non-null  object
 1   trip_creation_time           144316 non-null  object
 2   route_schedule_uuid          144316 non-null  object
 3   route_type                   144316 non-null  object
 4   trip_uuid                    144316 non-null  object
 5   source_center                144316 non-null  object
 6   source_name                  144316 non-null  object
 7   destination_center           144316 non-null  object
 8   destination_name             144316 non-null  object
 9   od_start_time                144316 non-null  object
 10  od_end_time                  144316 non-null  object
 11  start_scan_to_end_scan       144316 non-null  float64
 12  is_cutoff                    144316 non-null  bool
 13  cutoff_factor                144316 non-null  int64
 14  cutoff_timestamp             144316 non-null  object
 15  actual_distance_to_destination 144316 non-null float64
 16  actual_time                  144316 non-null  float64
 17  osrm_time                    144316 non-null  float64
 18  osrm_distance                144316 non-null  float64
 19  factor                       144316 non-null  float64
 20  segment_actual_time          144316 non-null  float64
 21  segment_osrm_time            144316 non-null  float64
 22  segment_osrm_distance        144316 non-null  float64
 23  segment_factor               144316 non-null  float64
dtypes: bool(1), float64(10), int64(1), object(12)
memory usage: 26.6+ MB
```

In [144…
```python
datetime_columns= ["trip_creation_time","od_start_time","od_end_time"]
for i in datetime_columns:
  df[i]=pd.to_datetime(df[i])
```

In [144…
```python
df["data"]=df["data"].astype("category")
df["route_type"]=df["route_type"].astype("category")
```

In [144…
```python
unknown_columns = ['is_cutoff', 'cutoff_factor', 'cutoff_timestamp', 'factor', 'seg
df.drop(unknown_columns, axis=1, inplace=True)
```

In [144…
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 144316 entries, 0 to 144866
Data columns (total 19 columns):
 #   Column                         Non-Null Count    Dtype
---  ------                         --------------    -----
 0   data                           144316 non-null   category
 1   trip_creation_time             144316 non-null   datetime64[ns]
 2   route_schedule_uuid            144316 non-null   object
 3   route_type                     144316 non-null   category
 4   trip_uuid                      144316 non-null   object
 5   source_center                  144316 non-null   object
 6   source_name                    144316 non-null   object
 7   destination_center             144316 non-null   object
 8   destination_name               144316 non-null   object
 9   od_start_time                  144316 non-null   datetime64[ns]
 10  od_end_time                    144316 non-null   datetime64[ns]
 11  start_scan_to_end_scan         144316 non-null   float64
 12  actual_distance_to_destination 144316 non-null   float64
 13  actual_time                    144316 non-null   float64
 14  osrm_time                      144316 non-null   float64
 15  osrm_distance                  144316 non-null   float64
 16  segment_actual_time            144316 non-null   float64
 17  segment_osrm_time              144316 non-null   float64
 18  segment_osrm_distance          144316 non-null   float64
dtypes: category(2), datetime64[ns](3), float64(8), object(6)
memory usage: 20.1+ MB
```

Feature Engineering:

- Converting the raw data into meaningful data.
- Creation of new fields from the existing fields

In [144…
```python
# Feature Engineering
df.head()
```

Out[1445]:

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_c
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…	Carting	trip-153741093647649320	IND38812

In [144…
```python
# grouping the data by "trip_uuid","source_center","destination_center"
grouping_features= ["trip_uuid","source_center","destination_center"]
group_df1 = df.groupby(by=grouping_features, as_index=False).agg({"data":"first",
                                    "trip_creation_time": "first",
                                    "route_type":"first",
```

```
                                        "source_name":"first",
                                        "destination_name":"first",
                                        "od_start_time":"first",
                                        "od_end_time":"first",
                                        "start_scan_to_end_scan":"first",
                                        "actual_distance_to_destination":"l
                                        "actual_time":"last",
                                        "osrm_time":"last",
                                        "osrm_distance":"last",
                                        "segment_actual_time":"sum",
                                        "segment_osrm_time":"sum",
                                        "segment_osrm_distance":"sum"})
```

In [144…]   `group_df1`

Out[1447]:

|	trip_uuid	source_center	destination_center	data	trip_creation_time	route_
0	trip-153671041653548748	IND209304AAA	IND000000ACB	training	2018-09-12 00:00:16.535741	
1	trip-153671041653548748	IND462022AAA	IND209304AAA	training	2018-09-12 00:00:16.535741	
2	trip-153671042288605164	IND561203AAB	IND562101AAA	training	2018-09-12 00:00:22.886430	Ca
3	trip-153671042288605164	IND572101AAA	IND561203AAB	training	2018-09-12 00:00:22.886430	Ca
4	trip-153671043369099517	IND000000ACB	IND160002AAC	training	2018-09-12 00:00:33.691250	
...	...	...	...	...	...	
26217	trip-153861115439069069	IND628204AAA	IND627657AAA	test	2018-10-03 23:59:14.390954	Ca
26218	trip-153861115439069069	IND628613AAA	IND627005AAA	test	2018-10-03 23:59:14.390954	Ca
26219	trip-153861115439069069	IND628801AAA	IND628204AAA	test	2018-10-03 23:59:14.390954	Ca
26220	trip-153861118270144424	IND583119AAA	IND583101AAA	test	2018-10-03 23:59:42.701692	
26221	trip-153861118270144424	IND583201AAA	IND583119AAA	test	2018-10-03 23:59:42.701692	

26222 rows × 18 columns

In [144…]
```python
# Calculating total time taken for the trip
group_df1["od_total_time"] = group_df1["od_end_time"] - group_df1["od_start_time"]
group_df1["od_total_time"] = group_df1["od_total_time"].apply(lambda x: round(x.tot
```

In [144…]   `group_df1`

Out[1449]:

	trip_uuid	source_center	destination_center	data	trip_creation_time	route_
0	trip-153671041653548748	IND209304AAA	IND000000ACB	training	2018-09-12 00:00:16.535741	
1	trip-153671041653548748	IND462022AAA	IND209304AAA	training	2018-09-12 00:00:16.535741	
2	trip-153671042288605164	IND561203AAB	IND562101AAA	training	2018-09-12 00:00:22.886430	Ca
3	trip-153671042288605164	IND572101AAA	IND561203AAB	training	2018-09-12 00:00:22.886430	Ca
4	trip-153671043369099517	IND000000ACB	IND160002AAC	training	2018-09-12 00:00:33.691250	
...	...	...	...	...	...	
26217	trip-153861115439069069	IND628204AAA	IND627657AAA	test	2018-10-03 23:59:14.390954	Ca
26218	trip-153861115439069069	IND628613AAA	IND627005AAA	test	2018-10-03 23:59:14.390954	Ca
26219	trip-153861115439069069	IND628801AAA	IND628204AAA	test	2018-10-03 23:59:14.390954	Ca
26220	trip-153861118270144424	IND583119AAA	IND583101AAA	test	2018-10-03 23:59:42.701692	
26221	trip-153861118270144424	IND583201AAA	IND583119AAA	test	2018-10-03 23:59:42.701692	

26222 rows × 19 columns

In [145...
```python
group_df1.drop(columns=["od_start_time","od_end_time"],inplace=True) # dropping the
```

In [145...
```python
# grouping the grouped data obtained by "trip_uuid"
group_df2 = group_df1.groupby(by="trip_uuid", as_index=False).agg({"data":"first",
                                        "trip_creation_time": "first",
                                        "route_type":"first",
                                        "source_name":"first",
                                        "destination_name":"last",
                                        "start_scan_to_end_scan":"sum",
                                        "actual_distance_to_destination":"s
                                        "actual_time":"sum",
                                        "osrm_time":"sum",
                                        "osrm_distance":"sum",
                                        "segment_actual_time":"sum",
                                        "segment_osrm_time":"sum",
                                        "segment_osrm_distance":"sum",
                                        "od_total_time":"sum"})
```

In [145...
```python
# Obtaining the meaningul data
df = group_df2
```

In [145...
```python
df.head()
```

Out[1453]:

	trip_uuid	data	trip_creation_time	route_type	source_name	des
0	trip-153671041653548748	training	2018-09-12 00:00:16.535741	FTL	Kanpur_Central_H_6 (Uttar Pradesh)	Kanp
1	trip-153671042288605164	training	2018-09-12 00:00:22.886430	Carting	Doddablpur_ChikaDPP_D (Karnataka)	Doddablp
2	trip-153671043369099517	training	2018-09-12 00:00:33.691250	FTL	Gurgaon_Bilaspur_HB (Haryana)	Gurga
3	trip-153671046011330457	training	2018-09-12 00:01:00.113710	Carting	Mumbai Hub (Maharashtra)	Mur
4	trip-153671052974046625	training	2018-09-12 00:02:09.740725	FTL	Bellary_Dc (Karnataka)	Sandu
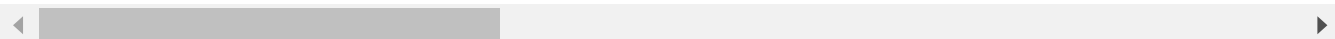
In [145... `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14787 entries, 0 to 14786
Data columns (total 15 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   trip_uuid                      14787 non-null  object
 1   data                           14787 non-null  category
 2   trip_creation_time             14787 non-null  datetime64[ns]
 3   route_type                     14787 non-null  category
 4   source_name                    14787 non-null  object
 5   destination_name               14787 non-null  object
 6   start_scan_to_end_scan         14787 non-null  float64
 7   actual_distance_to_destination 14787 non-null  float64
 8   actual_time                    14787 non-null  float64
 9   osrm_time                      14787 non-null  float64
 10  osrm_distance                  14787 non-null  float64
 11  segment_actual_time            14787 non-null  float64
 12  segment_osrm_time              14787 non-null  float64
 13  segment_osrm_distance          14787 non-null  float64
 14  od_total_time                  14787 non-null  float64
dtypes: category(2), datetime64[ns](1), float64(9), object(3)
memory usage: 1.5+ MB
```

In [145...
```python
# Extracting City, Place, State

location_name = "Kanpur_Central_H_6 (Uttar Pradesh)"

def extract_state(location_name):
  city_part, state_part = location_name.split("(")
  state = state_part.replace(")", "")
  return state

def extract_city(location_name):
  city_part, state_part = location_name.split("(")
  city = city_part.split("_")[0]
  return city

def extract_place(location_name):
  city_part, state_part = location_name.split("(")
  place = city_part.split("_")[1:]
  place = "_".join(place)
  return place
```

```
In [145…    # Extracting City, Place, State from source_name feature
            df["source_state"] = df["source_name"].apply(extract_state)
            df["source_city"] = df["source_name"].apply(extract_city)
            df["source_place"] = df["source_name"].apply(extract_place)
```

```
In [145…    # Extracting City, Place, State from destination_name feature
            df["destination_state"] = df["destination_name"].apply(extract_state)
            df["destination_city"] = df["destination_name"].apply(extract_city)
            df["destination_place"] = df["destination_name"].apply(extract_place)
```

```
In [145…    # Extracting day, month, year, hour from Trip Creation Date
            df['trip_creation_day'] = pd.to_datetime(df['trip_creation_time']).dt.strftime('%a'
            df['trip_creation_month'] = pd.to_datetime(df['trip_creation_time']).dt.strftime('%
            df['trip_creation_year'] = pd.to_datetime(df['trip_creation_time']).dt.year
            df["trip_creation_hour"]=df["trip_creation_time"].dt.hour
```

```
In [145…    df # Obtained meaningful data with additional features which were calculated using
```

Out[1459]:

	trip_uuid	data	trip_creation_time	route_type	source_name	
0	trip-153671041653548748	training	2018-09-12 00:00:16.535741	FTL	Kanpur_Central_H_6 (Uttar Pradesh)	
1	trip-153671042288605164	training	2018-09-12 00:00:22.886430	Carting	Doddablpur_ChikaDPP_D (Karnataka)	Do
2	trip-153671043369099517	training	2018-09-12 00:00:33.691250	FTL	Gurgaon_Bilaspur_HB (Haryana)	
3	trip-153671046011330457	training	2018-09-12 00:01:00.113710	Carting	Mumbai Hub (Maharashtra)	
4	trip-153671052974046625	training	2018-09-12 00:02:09.740825	FTL	Bellary_Dc (Karnataka)	
...	...	...	...	...	...	
14782	trip-153861095625827784	test	2018-10-03 23:55:56.258533	Carting	Chandigarh_Mehmdpur_H (Punjab)	Cha
14783	trip-153861104386292051	test	2018-10-03 23:57:23.863155	Carting	FBD_Balabhgarh_DPC (Haryana)	
14784	trip-153861106442901555	test	2018-10-03 23:57:44.429324	Carting	Kanpur_GovndNgr_DC (Uttar Pradesh)	
14785	trip-153861115439069069	test	2018-10-03 23:59:14.390954	Carting	Tirunelveli_VdkkuSrt_I (Tamil Nadu)	Tir
14786	trip-153861118270144424	test	2018-10-03 23:59:42.701692	FTL	Sandur_WrdN1DPP_D (Karnataka)	

14787 rows × 25 columns

```
In [146…    df.shape
```

Out[1460]:   (14787, 25)

```
In [146…    df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14787 entries, 0 to 14786
Data columns (total 25 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   trip_uuid                      14787 non-null  object
 1   data                           14787 non-null  category
 2   trip_creation_time             14787 non-null  datetime64[ns]
 3   route_type                     14787 non-null  category
 4   source_name                    14787 non-null  object
 5   destination_name               14787 non-null  object
 6   start_scan_to_end_scan         14787 non-null  float64
 7   actual_distance_to_destination 14787 non-null  float64
 8   actual_time                    14787 non-null  float64
 9   osrm_time                      14787 non-null  float64
 10  osrm_distance                  14787 non-null  float64
 11  segment_actual_time            14787 non-null  float64
 12  segment_osrm_time              14787 non-null  float64
 13  segment_osrm_distance          14787 non-null  float64
 14  od_total_time                  14787 non-null  float64
 15  source_state                   14787 non-null  object
 16  source_city                    14787 non-null  object
 17  source_place                   14787 non-null  object
 18  destination_state              14787 non-null  object
 19  destination_city               14787 non-null  object
 20  destination_place              14787 non-null  object
 21  trip_creation_day              14787 non-null  object
 22  trip_creation_month            14787 non-null  object
 23  trip_creation_year             14787 non-null  int32
 24  trip_creation_hour             14787 non-null  int32
dtypes: category(2), datetime64[ns](1), float64(9), int32(2), object(11)
memory usage: 2.5+ MB
```

In [146...
```python
# Dropping unwanted columns
unwanted_columns = ["trip_creation_time","source_name","destination_name"]
df.drop(unwanted_columns, axis=1, inplace=True)
```

In [146...
```python
# converting date fields to object
df["trip_creation_day"] = df["trip_creation_day"].astype("object")
df["trip_creation_month"] = df["trip_creation_month"].astype("object")
df["trip_creation_year"] = df["trip_creation_year"].astype("object")
df["trip_creation_hour"] = df["trip_creation_hour"].astype("object")
```

In [146...
```python
# Final check for nulls in percentage
round((df.isnull().sum() / len(df))*100,2)
```

Out[1464]:

	0
trip_uuid	0.0
data	0.0
route_type	0.0
start_scan_to_end_scan	0.0
actual_distance_to_destination	0.0
actual_time	0.0
osrm_time	0.0
osrm_distance	0.0
segment_actual_time	0.0
segment_osrm_time	0.0
segment_osrm_distance	0.0
od_total_time	0.0
source_state	0.0
source_city	0.0
source_place	0.0
destination_state	0.0
destination_city	0.0
destination_place	0.0
trip_creation_day	0.0
trip_creation_month	0.0
trip_creation_year	0.0
trip_creation_hour	0.0

dtype: float64

In [146…

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14787 entries, 0 to 14786
Data columns (total 22 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   trip_uuid                      14787 non-null  object
 1   data                           14787 non-null  category
 2   route_type                     14787 non-null  category
 3   start_scan_to_end_scan         14787 non-null  float64
 4   actual_distance_to_destination 14787 non-null  float64
 5   actual_time                    14787 non-null  float64
 6   osrm_time                      14787 non-null  float64
 7   osrm_distance                  14787 non-null  float64
 8   segment_actual_time            14787 non-null  float64
 9   segment_osrm_time              14787 non-null  float64
 10  segment_osrm_distance          14787 non-null  float64
 11  od_total_time                  14787 non-null  float64
 12  source_state                   14787 non-null  object
 13  source_city                    14787 non-null  object
 14  source_place                   14787 non-null  object
 15  destination_state              14787 non-null  object
 16  destination_city               14787 non-null  object
 17  destination_place              14787 non-null  object
 18  trip_creation_day              14787 non-null  object
 19  trip_creation_month            14787 non-null  object
 20  trip_creation_year             14787 non-null  object
 21  trip_creation_hour             14787 non-null  object
dtypes: category(2), float64(9), object(11)
memory usage: 2.3+ MB
```

Exploratory Data Analysis:

- Separating Numerical and Categorical columns for analysis.
- Performing In-Depth Analysis.
- Hypothesis Testing to validate the trends.

```python
In [146…  # Separating numerical columns
          numerical_columns = df.select_dtypes(include=['number']).columns

          # Separating categorical columns
          categorical_columns = df.select_dtypes(include=['object', 'category']).columns

          print("Numerical columns:", numerical_columns)
          print("Categorical columns:", categorical_columns)
```

```
Numerical columns: Index(['start_scan_to_end_scan', 'actual_distance_to_destinatio
n',
       'actual_time', 'osrm_time', 'osrm_distance', 'segment_actual_time',
       'segment_osrm_time', 'segment_osrm_distance', 'od_total_time'],
      dtype='object')
Categorical columns: Index(['trip_uuid', 'data', 'route_type', 'source_state', 'so
urce_city',
       'source_place', 'destination_state', 'destination_city',
       'destination_place', 'trip_creation_day', 'trip_creation_month',
       'trip_creation_year', 'trip_creation_hour'],
      dtype='object')
```

```python
In [146…  for i in categorical_columns:
            print(df[i].value_counts())
            print("-"*100)
```

```
trip_uuid
trip-153671041653548748     1
trip-153791166614583191     1
trip-153791188846803726     1
trip-153791199511952768     1
trip-153791203866379915     1
                           ..
trip-153730334220215546     1
trip-153730343361346090     1
trip-153730345417711989     1
trip-153730351817227074     1
trip-153861118270144424     1
Name: count, Length: 14787, dtype: int64
--------------------------------------------------------------------------------
------------------
data
training    10645
test         4142
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
route_type
Carting    8906
FTL        5881
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
source_state
Maharashtra                 2714
Karnataka                   2143
Haryana                     1823
Tamil Nadu                  1039
Telangana                    784
Uttar Pradesh                760
Gujarat                      750
Delhi                        725
West Bengal                  665
Punjab                       536
Rajasthan                    514
Andhra Pradesh               435
Bihar                        351
Madhya Pradesh               318
Kerala                       289
Assam                        268
Jharkhand                    160
Uttarakhand                  114
Orissa                       107
Chandigarh                    93
Goa                           65
Chhattisgarh                  43
Himachal Pradesh              34
Jammu & Kashmir               17
Dadra and Nagar Haveli        15
Pondicherry                   12
Nagaland                       5
Arunachal Pradesh              4
Mizoram                        4
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
source_city
Gurgaon       1128
Bengaluru     1052
Bhiwandi       697
```

```
        Mumbai          654
        Bangalore       648
                        ...
        Dhaka           1
        Ukkadagatri     1
        Sultana         1
        Banka           1
        Sandur          1
        Name: count, Length: 731, dtype: int64
        ------------------------------------------------------------------------
        ------------------
        source_place
        Bilaspur_HB     1052
        Mankoli_HB      697
                        640
        Nelmngla_H      624
        Bomsndra_HB     455
                        ...
        ThaneDPP_D      1
        JatniDPP_D      1
        Chnglptu_DC     1
        Ymunpurm_D      1
        WrdN1DPP_D      1
        Name: count, Length: 756, dtype: int64
        ------------------------------------------------------------------------
        ------------------
        destination_state
        Maharashtra             2561
        Karnataka               2294
        Haryana                 1640
        Tamil Nadu              1084
        Uttar Pradesh           805
        Telangana               784
        Gujarat                 734
        West Bengal             697
        Delhi                   657
        Punjab                  617
        Rajasthan               550
        Andhra Pradesh          442
        Bihar                   367
        Madhya Pradesh          350
        Kerala                  270
        Assam                   232
        Jharkhand               181
        Uttarakhand             122
        Orissa                  119
        Chandigarh              65
        Goa                     52
        Chhattisgarh            43
        Himachal Pradesh        42
        Arunachal Pradesh       25
        Jammu & Kashmir         20
        Dadra and Nagar Haveli  17
        Meghalaya               8
        Mizoram                 6
        Nagaland                1
        Tripura                 1
        Daman & Diu             1
        Name: count, dtype: int64
        ------------------------------------------------------------------------
        ------------------
        destination_city
        Bengaluru       1088
        Mumbai          966
```

```
Gurgaon          877
Delhi            554
Bangalore        551
                 ...
Chapra             1
Shamshabad         1
Kullu              1
Oriyur             1
Lunawada           1
Name: count, Length: 856, dtype: int64
--------------------------------------------------------------------------------
------------------
destination_place
Bilaspur_HB      821
                 753
Nelmngla_H       548
Mankoli_HB       403
DC               384
                 ...
PonaniRD_D         1
JyotiNgr_D         1
Robinson_D         1
KetyDPP_D          1
VrdhriRD_D         1
Name: count, Length: 844, dtype: int64
--------------------------------------------------------------------------------
------------------
trip_creation_day
Wed    2731
Sat    2128
Thu    2103
Fri    2057
Tue    2035
Mon    1980
Sun    1753
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
trip_creation_month
Sep    13011
Oct     1776
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
trip_creation_year
2018    14787
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
trip_creation_hour
22    1123
23    1107
20    1080
0      991
21     872
19     837
1      748
2      702
18     696
3      651
4      635
6      610
17     595
16     526
```

```
5        505
7        472
15       469
14       379
8        345
13       328
9        317
12       270
11       267
10       262
Name: count, dtype: int64
--------------------------------------------------------------------------------
------------------
```

In [146…
```python
for i in categorical_columns:
    print(i)
    print(df[i].unique())
    print("-"*100)
```

```
trip_uuid
['trip-153671041653548748' 'trip-153671042288605164'
 'trip-153671043369099517' ... 'trip-153861106442901555'
 'trip-153861115439069069' 'trip-153861118270144424']
--------------------------------------------------------------------------------
-----------------
data
['training', 'test']
Categories (2, object): ['test', 'training']
--------------------------------------------------------------------------------
-----------------
route_type
['FTL', 'Carting']
Categories (2, object): ['Carting', 'FTL']
--------------------------------------------------------------------------------
-----------------
source_state
['Uttar Pradesh' 'Karnataka' 'Haryana' 'Maharashtra' 'Tamil Nadu'
 'Gujarat' 'Delhi' 'Telangana' 'Rajasthan' 'Assam' 'Madhya Pradesh'
 'West Bengal' 'Andhra Pradesh' 'Punjab' 'Chandigarh' 'Goa' 'Jharkhand'
 'Pondicherry' 'Orissa' 'Uttarakhand' 'Himachal Pradesh' 'Kerala'
 'Arunachal Pradesh' 'Bihar' 'Chhattisgarh' 'Dadra and Nagar Haveli'
 'Jammu & Kashmir' 'Mizoram' 'Nagaland']
--------------------------------------------------------------------------------
-----------------
source_city
['Kanpur' 'Doddablpur' 'Gurgaon' 'Mumbai Hub ' 'Bellary' 'Chennai'
 'HBR Layout PC ' 'Surat' 'Delhi' 'Pune' 'FBD' 'Shirala' 'Hyderabad'
 'Thirumalagiri' 'Gulbarga' 'Jaipur' 'Allahabad' 'Guwahati' 'Narsinghpur'
 'Shrirampur' 'Hoogly' 'Madakasira' 'Sonari' 'Bengaluru' 'Dindigul'
 'Jalandhar' 'Faridabad' 'Chandigarh' 'Deoli' 'Pandharpur' 'CCU'
 'Bhandara' 'Kurnool' 'Bhiwandi' 'Bhatinda' 'RoopNagar' 'Bantwal' 'Lalru'
 'Kadi' 'Shahdol' 'Gangakher' 'Durgapur' 'Vapi' 'Jamjodhpur' 'Jetpur'
 'Mehsana' 'Jabalpur' 'Junagadh' 'Gundlupet' 'Mysore' 'Goa' 'Bhopal'
 'Sonipat' 'Himmatnagar' 'Jamshedpur' 'Pondicherry' 'MAA' 'Anand' 'Udgir'
 'Nadiad' 'Villupuram' 'Purulia' 'Bhopal MP Nagar ' 'Bhubaneshwar'
 'Bamangola' 'Mumbai' 'Tiruppattur' 'Kotdwara' 'Medak' 'Bangalore'
 'Dhrangadhra' 'Hospet ' 'Ghumarwin' 'ChandroknaRD' 'Agra' 'Sitapur'
 'Canacona' 'Bilimora' 'SultnBthry' 'Lucknow' 'Vellore' 'Bhuj' 'Dinhata'
 'BOM' 'Margherita' 'Boisar' 'Vizag' 'Tezpur' 'Koduru' 'Tirupati' 'Pen'
 'Kolkata' 'AMD' 'Ahmedabad' 'Faizabad' 'Gandhinagar' 'Anantapur' 'Betul'
 'Panskura' 'BLR' 'Rasipurm' 'Sankari' 'Jorhat' 'PNQ Pashan DPC '
 'Srikakulam' 'Dehradun' 'Jassur' 'Sawantwadi' 'Shajapur' 'OK' 'Ludhiana'
 'GreaterThane' 'Tirupur' 'Salem ' 'Darjeeling' 'Tiruchi ' 'Noida'
 'PNQ Vadgaon Sheri DPC ' 'Thiruvarur' 'Ranchi' 'Guna' 'Raver'
 'Faridabad ' 'Jairampur' 'Chamoli' 'Pali' 'Kamareddy' 'Gopiganj'
 'Varanasi' 'Dharmapuri' 'Hubli' 'Duddhi' 'Sasaram' 'Davangere' 'Panipat '
 'Chittaurgarh' 'Solapur' 'Pratapgarh' 'Del' 'Vinukonda' 'Ongole'
 'LowerParel' 'Sagara' 'Tikamgarh' 'Ghaziabad' 'Chhapra' 'BiharSarif'
 'Pallakad' 'Kanakapura' 'Mangalore' 'Aurangabad' 'Barh' 'Coimbatore'
 'Bhadrak' 'Narnaul' 'Hisar' 'Bihta' 'Silchar' 'Sillod' 'Nellore' 'Katwa'
 'Thamarassery' 'Safidon' 'Vijayawada' 'Machilipatnam' 'Nazirpur'
 'Vikarabad' 'Rampurhat' 'Visakhapatnam' 'Lalgola' 'Rampur' 'Teok'
 'Kakinada' 'Amalapuram' 'Muzaffrpur' 'Kalka' 'Buldhana' 'Karad'
 'JoguGadwal' 'Madhepura' 'Simrahi' 'Atmakur' 'Hassan' 'Chikodi' 'Rohtak '
 'Patiala' 'Ajmer' 'Channaraya' 'Naugchia' 'Ambala ' 'Korba' 'Pithorgarh'
 'Deoghar' 'Alwar' 'Gorakhpur' 'Bhatpara' 'Dumka' 'Bahadurgarh' 'Kanth'
 'Nichlaul' 'Warangal' 'Aonla' 'Dhar' 'Bagnan' 'Naraingarh' 'Kashipur'
 'Ratanpura' 'Gondia' 'Zahirabad' 'Samana' 'Bhadrachalam' 'Baraut' 'Sikar'
 'Jamnagar' 'Kakdwip' 'Gadarwara' 'Gwalior' 'Mumbai Antop Hill ' 'Akola'
 'Kalluvathukal' 'Surendranagar' 'Buxar' 'Trivandrum' 'Etawah' 'Bhagalpur'
 'Vadodara' 'Panaji Goa ' 'GZB' 'Chhata' 'Luxettipet' 'Mancherial'
 'Kottayam' 'Parakkdavu' 'Pthnmthitt' 'Dhule' 'DehriSone' 'Jaipur '
 'Brahmapuri' 'Ramagundam' 'Gomoh' 'Kollam' 'Wardha' 'Barnala' 'Latur'
```

'Puttaprthi' 'Ghatampur' 'Upleta' 'Khammam' 'Akbarpur' 'Bhanvad' 'Basti'
'Dibrugarh' 'Mussoorie' 'Kalpetta' 'Phalodi' 'Guskhara' 'Mainaguri'
'Gosainganj' 'Bhusawal' 'Vadodara ' 'Hyd' 'Nalbari' 'Talegaon'
'SrinagarUK' 'Shimoga' 'Bailhongal' 'Gonda' 'Manapparai' 'Udaipur'
'Ghazipur' 'Guruvayoor' 'Chetpet' 'Wai' 'Karkala' 'GGN' 'Patancheru'
'Kozhikode' 'Kumbakonam' 'Rameswram' 'Shirur' 'Degana' 'Pattukotai'
'Srisailam' 'Lalpet' 'Madurai' 'Sathyamangalam' 'Cjb' 'Patiala '
'Usilampatti' 'Khurai' 'Nuzvid' 'Koppa' 'Tiptur' 'Rajamundry' 'Nainital'
'Haveri' 'Dumraon' 'Machhiwara' 'NeemKaThana' 'Baheri' 'Dharapuram'
'Mohania' 'Bilaspur' 'Naugarh' 'Patran' 'Mahbubabad' 'Tirunelveli'
'Bhavnagar' 'Dhanbad' 'Mahadevpur' 'Puttur' 'Jammikunta' 'Narsingpur'
'Jagtial' 'Karimnagar' 'Amd' 'Manthani' 'Aligarh' 'Rudrapur'
'Manamelkudi' 'Malegaon' 'Sindhanur' 'Murbad' 'Medchal' 'Kanker' 'Unjha'
'Khambhat' 'Salem' 'Malda' 'Haridwar ' 'Anjar' 'Pathankot' 'Bhubaneswar'
'Chandigarh ' 'Sholinghur' 'Jhansi' 'Panipat' 'Silvassa' 'Balasore'
'Nagaur' 'Bhilwara' 'Ghanpur' 'Achrol' 'Hazaribag' 'Dharwad' 'Chhatarpur'
'Arrah' 'Udupi ' 'Gooty' 'Bareilly' 'Kallachi' 'Devarakonda'
'Mahabubnagar' 'Hailakandi' 'Jeypore' 'Wanaparthy' 'Ramnthpurm'
'Sitamari' 'Makrana' 'Sankaramangalam' 'Ratnagiri' 'Meerut ' 'Chikhli'
'Cumbum' 'Sakleshpur' 'Anthiyour' 'Khanna' 'Bharatpur' 'Bina' 'Lonavala'
'AurngbadBR' 'Ambah' 'Amreli' 'Dadri' 'SikandraRao' 'Kaman' 'Pukhrayan'
'Raichur' 'Raipur' 'Bellmpalli' 'Chinnur' 'Bankura' 'Bareli' 'Panagarh'
'Chhindwara' 'Mananthavady' 'Kharagpur' 'JognderNgr' 'Phagwara'
'Srivijaynagar' 'Thoppur' 'Bongaigaon' 'Rajgurunagar' 'Deoband' 'Chopan'
'Chomu' 'Satara' 'Blr' 'Rewari' 'Mainpuri' 'Nandigama' 'Kolhapur'
'Tirurangadi' 'Vadakara' 'Mariani' 'Baharampur' 'Almora' 'Jayamkondan'
'Sonepur' 'Karnal ' 'Bettiah' 'YamunaNagar' 'Godda' 'Ratlam' 'Sagar'
'Kaptanganj' 'Katni' 'Umaria' 'Sambhal' 'Sitarganj' 'Vaijiapur' 'Akhnoor'
'Ashta' 'Aluva' 'Bokaro' 'ChrkhiDdri' 'Kattappana' 'Vijayawada '
'Dharamshala' 'Dausa' 'Katihar' 'Shirpur' 'Bangarapet' 'Dwarka'
'Bagepalli' 'Khurja' 'Haldwani' 'Asangaon' 'Moodbidri' 'Deesa'
'Kodaikanal' 'Bhabhar' 'Khedbrahma' 'Kodinar' 'RaisingNgr' 'Mejia'
'Vidisha' 'Jammu' 'Malvan' 'Roha' 'Hoskote' 'Tezu' 'Hooghly' 'Mau'
'Sujangarh' 'Gohpur' 'Peterbar' 'Thrissur' 'Rajgir' 'Polur' 'Ankola'
'Kanhangad' 'Chalakudy' 'Midnapore' 'Mungeli' 'Palampur' 'Mungaoli'
'SirhindFatehgarh' 'Jangipur' 'DalsinghSarai' 'Bewar' 'Pakur' 'Jasai'
'Kankavali' 'Hapur' 'Nanded' 'Palani' 'Palanpur' 'Narsapur' 'Dalkhola'
'Purnia' 'Airport ' 'PNQ Rahatani DPC ' 'Kalpakkam' 'Ashokngr'
'MughalSarai' 'Dohrighat' 'Manthuka' 'Bishwanath' 'Tulsipur' 'Aizawl'
'Tirur' 'Cochin' 'Uchila' 'Shevgaon' 'Athani' 'Amravati' 'Nilambur'
'Karimganj' 'Shamli' 'HanumanJNC' 'Bikramgang' 'Fatepur' 'Gangarampr'
'Itahar' 'Lakhnadon' 'Manikchak' 'Sihora' 'Jamtara' 'Giridih' 'Alappuzha'
'Bethamangala' 'Rajkot' 'Gola' 'Ambasamdrm' 'Majalgaon' 'Jabalpur '
'Hanumangarh' 'Kapurthala' 'Barmer' 'Tamluk' 'Palakonda' 'Mahad' 'Chamba'
'Krishnagiri' 'Tirchngode' 'Dholpur' 'Kabuganj' 'Bhadohi' 'Madnapalle'
'Kundapura' 'Irinjlkuda' 'Mokokchung' 'Chapra' 'Lalitpur' 'Murshidabad'
'Bijapur' 'Beed' 'Madhupur' 'Hajipur' 'Khurdha' 'Wankaner' 'Hindupur'
'Bulndshahr' 'Aland' 'BariSadri' 'Husnabad' 'Bhuvanagiri' 'Islampure'
'Manjhaul' 'Bikaner' 'Siwan' 'Rupnarayanpur' 'Plassey' 'Mylduthuri'
'Modinagar' 'Nowda' 'Theni' 'Sagardighi' 'PaontSahib' 'Kaliyaganj'
'Taranagar' 'Jath' 'Chiplun' 'Suratgarh' 'DoomDooma' 'Khonsa' 'Talala'
'Vadnagar' 'Arambag' 'Haldia' 'Sehore' 'Hura' 'Erode ' 'Gadag' 'Shahganj'
'Balrampur' 'Dehradun ' 'Mehkar' 'Kalyandurg' 'Berhampore' 'Dhaka'
'Bassi' 'Ukkadagatri' 'Sultana' 'Banka' 'Mumbai Mahim ' 'Asifabad'
'Sivasagar' 'Jodhpur' 'Khatra' 'LakhimpurN' 'Kishangarh' 'Narktiganj'
'Aliganj' 'Bongaon' 'Nedumangad' 'Chandausi' 'Sujanpur' 'Karukachal'
'Kamarpukur' 'Keshiary' 'Firozabad' 'Melur' 'Thuraiyur' 'Nakashipara'
'Nasirabad' 'Nagamangala' 'Morgram' 'Triveninganj' 'Barhi' 'Bhatiya'
'Chotila' 'Falna' 'Kopargaon' 'AnandprShb' 'Tinusukia' 'Modasa' 'Palasa'
'Dahanu' 'Gudur' 'Khanapur' 'Udala' 'Kathua' 'Moga' 'Ganga Nagar' 'Khed'
'Brajrajnagar' 'Sambalpur' 'Ghanashyampur' 'Seoni' 'Rajpura' 'Kadaba'
'Sangola' 'Jaleswar' 'Bhilad' 'Umreth' 'Pachore' 'Shegaon' 'Sundargarh'
'Sunam' 'Morbi' 'Fatehabad' 'Mundakayam' 'Vrindavan' 'Jalalabad'
'Angamaly' 'Asansol' 'Kadiri' 'Vadakkencherry' 'Balangir' 'Raxaul'

```
 'Sirohi' 'Manmad' 'Halvad' 'Nagpur' 'Shoranur' 'Bareilly ' 'Kaithal'
 'Ranaghat' 'Sakri' 'Bangana' 'Kangayam' 'Palitana' 'Valsad' 'Dabhoi'
 'Muktsar' 'Jhunir' 'Bheemunipatnam' 'Sedam' 'Virudhchlm' 'Gangavathi'
 'Moradabad' 'Karanjia' 'Chimkurthy' 'Phusro' 'Jhajjar' 'Kottarakkara'
 'Shikohabad' 'Munger' 'Chhaygaon' 'Hathras ' 'Kusumnchi' 'Pauri'
 'Rishikesh' 'Khatauli' 'Baddi' 'Mandi' 'Merta' 'Kuthuparamba'
 'Kaghaznagar' 'Auraiya' 'Giddarbaha' 'Paradip' 'Jharsuguda' 'Gobicheti'
 'Arakkonam' 'Pilani ' 'Central Goa ' 'Simlapal' 'Baripada' 'Cuttack'
 'Saharsa' 'Rajgarh' 'Durg' 'Balurghat' 'Dola' 'Pappadahandi' 'Sinnar'
 'Barasat' 'Khanakul' 'Sendhwa' 'Ramgarh' 'BilaspurHP' 'Sidhmukh' 'Angul'
 'SawaiMadhopur' 'Ambegaon' 'Thakurdwara' 'Malemruvathur' 'Bishnupur'
 'Dhoraji' 'Meham' 'Uthangarai' 'Shadnagar' 'Bhiwani' 'Mahasamund'
 'Mandla' 'Janakpuri ' 'Phulera' 'Sandur']
-----------------------------------------------------------------------------
-----------------
source_place
['Central_H_6 ' 'ChikaDPP_D ' 'Bilaspur_HB ' '' 'Dc ' 'Poonamallee '
 'Chrompet_DPC ' 'Central_D_12 ' 'Lajpat_IP ' 'North_D_3 '
 'Balabhgarh_DPC ' 'Central_DPP_3 ' 'Shamshbd_H ' 'Xroad_D ' 'Nehrugnj_I '
 'Central_I_7 ' 'Central_H_1 ' 'Nangli_IP ' 'North ' 'KndliDPP_D '
 'Central_D_9 ' 'DavkharRd_D ' 'Bandel_D ' 'RTCStand_D ' 'Central_DPP_1 '
 'KGAirprt_HB ' 'North_D_2 ' 'Central_D_1 ' 'DC ' 'Mthurard_L '
 'Mullanpr_DC ' 'Central_DPP_2 ' 'RajCmplx_D ' 'Beliaghata_DPC '
 'RjnaiDPP_D ' 'AbbasNgr_I ' 'Mankoli_HB ' 'DPC ' 'Airport_H ' 'Hub '
 'Gateway_HB ' 'Tathawde_H ' 'ChotiHvl_DC ' 'Trmltmpl_D ' 'OnkarDPP_D '
 'Mehmdpur_H ' 'KaranNGR_D ' 'Sohagpur_D ' 'Chrompet_L ' 'Busstand_D '
 'Central_I_1 ' 'IndEstat_I ' 'Court_D ' 'Panchot_IP ' 'Adhartal_IP '
 'DumDum_DPC ' 'Bomsndra_HB ' 'Swamylyt_D ' 'Yadvgiri_IP ' 'Old City '
 'Kundli_H ' 'Central_I_3 ' 'Vasanthm_I ' 'Poonamallee_HB ' 'VUNagar_DC '
 'NlgaonRd_D ' 'Bnnrghta_L ' 'Thirumtr_IP ' 'GariDPP_D ' 'Jogshwri_I '
 'KoilStrt_D ' 'CotnGren_M ' 'Nzbadrd_D ' 'Dwaraka_D ' 'Nelmngla_H '
 'NvygRDPP_D ' 'Gndhichk_D ' 'Central_D_3 ' 'Chowk_D ' 'CharRsta_D '
 'Kollgpra_D ' 'Peenya_IP ' 'GndhiNgr_IP ' 'Sanpada_I ' 'WrdN4DPP_D '
 'Sakinaka_RP ' 'CivilHPL_D ' 'OstwlEmp_D ' 'Gajuwaka ' 'Mhbhirab_D '
 'MGRoad_D ' 'Balajicly_I ' 'BljiMrkt_D ' 'Dankuni_HB ' 'Trnsport_H '
 'Rakhial ' 'Memnagar ' 'East_I_21 ' 'Mithakal_D ' 'TrnspNgr_D '
 'KamaStrt_I ' 'Bilaspur_RP ' 'PatelWrd_D ' 'CottonGreen_DPC ' 'Pawane_L '
 'Central_I_4 ' 'JPNagar_Pc ' 'Knrpati_D ' 'Trchngrd_D ' 'Kengeri_IP '
 'Kharar_DC ' 'KHRoad_I ' 'RicMilRd_D ' 'MiraRd_IP ' 'Potheri '
 'Kuslpram_I ' 'Nayapalli ' 'Nirjanpur_L ' 'Uppal_I ' 'Jalukbari '
 'Chndivli_PC ' 'AadiDPP_D ' 'Hoodi_IP ' 'LaxmiNgr_D ' 'Kadugodi_D '
 'MhpraRD_D ' 'Chikdply_I ' 'Mayapuri_PC ' 'RPC ' 'Mylapore '
 'GillChwk_DC ' 'Anjur_C ' 'Kishangarh_DPC ' 'Rohini_DPC ' 'MilrGanj_HB '
 'Koliplm_I ' 'Ghansoli_DC ' 'Bhogal ' 'Patparganj_DPC ' 'Hillcard_DC '
 'Sec 02_DPC ' 'Kadipur ' 'Chukhndi_D ' 'AmzonDev_V ' 'Mundhawa_L '
 'Alwal_L ' 'Bypasrd_D ' 'HnumnDPP_D ' 'Truptingr_D ' 'Peenya_L '
 'PunjabiB_L ' 'Mdiclcly_D ' 'CGRoad_D ' 'Nayagaon_I ' 'Devenply_I '
 'Wardno13_D ' 'MahmurGj_IP ' 'Central_D_2 ' 'Adargchi_IP ' 'MChwkDPP_D '
 'Central_I_2 ' 'Raiprvlg_L ' 'East_H_1 ' 'Nimachrd_D ' 'Okhla_PC '
 'Bownplly_C ' 'Narynpur_C ' 'PC ' 'HUB ' 'Chikdply_C ' 'Sector02_C '
 'Kothapet_D ' 'SubhVRTL_I ' 'CP ' 'Vardhard_D ' 'Sanpada_CP ' 'Egmore_C '
 'Begumpur_CP ' 'Sodal Road ' 'Beleghta_CP ' 'MndiRoad_D ' 'MohanNgr_C '
 'Prbhunth_D ' 'Soghra_D ' 'ChndrNgr_D ' 'Alandur_C ' 'VaniThtr_D '
 'Kuntikna_H ' 'MhimWest_C ' 'Malahi_D ' 'Vasai_CP ' 'Mdhavram_C '
 'Sector63_L ' 'Karayam_H ' 'IndstlAr_I ' 'ShivNgar_D ' 'Sirikona_H '
 'Jogeshwri_L ' 'ZebaTWR_D ' 'Mhdiptnm_C ' 'North_R_8 ' 'Pratpngr_D '
 'BSarani_D ' 'Kapshera_L ' 'Chungam_D ' 'HatRDDPP_D ' 'Egmore_DPC '
 'Rynapadu_H ' 'Wazirpur_L ' 'Alwal_I ' 'SrifoDPP_D ' 'Gajuwaka_IP '
 'KrsprDPP_D ' 'MiraRoad_M ' 'RoshnBgh_I ' 'Wardno3_D ' 'Vijdurg_D '
 'Bbganj_I ' 'Pinjore_DC ' 'Thsil3PL_D ' 'Mundhe_D ' 'ColctrOf_D '
 'Krishnpr_D ' 'Bazar_D ' 'IndraNgr_D ' 'Pandrnga_I ' 'GainMrkt_L '
 'FoySGRRD_I ' 'patna_D ' 'Vijayght_D ' 'Jhilmil_L ' 'Tilknagr_DC '
 'Kumud_D ' 'Barmasia_D ' 'Nangli_L ' 'Matriprm_IP ' 'TrengaRD_D '
 'Dudhani_D ' 'Dayanand_D ' 'HunterRd_I ' 'KdidmCLY_D ' 'Trimurti_D '
```

```
'Harop_D ' 'Ward2DPP_D ' 'Vaishali_D ' 'MubarDPP_D ' 'RamNgr_D '
'Mohim_D ' 'RIICO_L ' 'PODPP_D ' 'ITDARd_D ' 'SrnprHwy_D ' 'FatehpRd_I '
'KlngrDPP_D ' 'MPward_D ' 'HrihrNgr_I ' 'Gaurkshn_I ' 'Pariplly_D '
'Sarswati_D ' 'NgrNigam_DC ' 'Pettah_D ' 'MhraChng_D ' 'Pbroad_DC '
'Karelibaug_DPC ' 'Mohan_Nagar_DPC ' 'Panvel_D ' 'GvrdnDPP_D '
'Kalyan West _Dc ' 'Ambernath_Dc ' 'ShivaDPP_D ' 'Mumbra_DC ' 'Hitech_D '
'UzanBazr_DC ' 'KeRoad_D ' 'Klskhrpt_D ' 'MIDCAvdn_I ' 'Dilliyan_D '
'Shahdara ' 'DhnliRth_D ' 'Pdmavati_D ' 'KhsmiDPP_D ' 'RamaNgr_D '
'Srnwsngr_D ' 'Gokulam_D ' 'StatinRD_D ' 'AdrshSt_DC ' 'NSTRoad_I '
'NehruNGR_D ' 'Mainroad_D ' 'Madhavaram_L ' 'GndhiNgr_D ' 'Bokule_H '
'BrlwgDPP_D ' 'Ward19_D ' 'PalikDPP_D ' 'Itachnda_D ' 'MilpaDPP_D '
'Pakrela_D ' 'Uppal_L ' 'Samarth_D ' 'Tolichwk_I ' 'Chndivli_D '
'Porur_DPC ' 'Sholinganallur_Dc ' 'LB-Nagar_Dc ' 'Panchkula '
'Bhgtpura_D ' 'Central_DPP_4 ' 'Srikot_D ' 'NamoNagr_D ' 'KotwaliN_D '
'BypassRd_D ' 'Mangri_I ' 'Kaithwal_D ' 'TrtllaRD_L ' 'ManhrBld_D '
'Blmrgnst_D ' 'Idgah_P ' 'MarketRd_D ' 'Ameenpur_I ' 'Feroke_H '
'Mettu_DC ' 'Kdthdstrt_D ' 'BhukrdPP_D ' 'anthniyr_D ' 'Sishumdr_D '
'PriyrNGR_D ' 'Kappalur_H ' 'Mlydpthr_D ' 'Kovaipudur_Dc ' 'MrutiNGR_D '
'Talaiya_D ' 'Ward17_D ' 'BMRd_D ' 'YTRd_D ' 'AtoNgrRd_I ' 'Sookhtal_D '
'GuttalRD_D ' 'Nishangr_D ' 'MnBzrDPP_D ' 'War5DPP_D ' 'HoliCDPP_D '
'Techrcly_D ' 'Bhabua_D ' 'Torwa_DC ' 'New Alipore_DPC ' 'MheshNGR_D '
'Yellanda_D ' 'VdkkuSrt_I ' 'ChtrGIDC_IP ' 'Kalynpur_I ' 'Kataram_D '
'Darbe_DC ' 'ConduDPP_D ' 'JrjolDPP_D ' 'Aravind_D ' 'KamnHbRD_I '
'Chandkheda_Dc ' 'Chaitnya_D ' 'MohnVRTL_D ' 'RjndrNgr_DC ' 'UdhamNgr_H '
'Kalyan ' 'TmpleSrt_D ' 'Raiprkln_C ' 'Rawlgaon_D ' 'Varachha_DC '
'South_D_4 ' 'VasaviNg_D ' 'SnkunDPP_D ' 'MROoffce_D ' 'ZuariNgr_IP '
'Mangol_DC ' 'KrisnKunj_D ' 'Lake Avenue_DPC ' 'MotvdDPP_D '
'Kadtmpty_H ' 'krshnPly_DC ' 'Selakui_D ' 'Tejpal_I ' 'BjbNgr_DC '
'Sixmile ' 'PndrgNgr_DC ' 'Sec-83_DC ' 'ArkonmRD_D ' 'LVMColge_D '
'Samrvrni_D ' 'Ganeshwr_D ' 'KisanCo_D ' 'Palakrty_D ' 'BgwriDPP_D '
'BodomBzr_DC ' 'MaladWest_CP ' 'Katira_D ' 'Mnanthla_D ' 'StatonRd_D '
'Madhavaram_DPC ' 'ZamQuatr_D ' 'DindiRD_D ' 'kalibari_D ' 'Kelasahi_D '
'VallaDPP_D ' 'TnhbBlkC_D ' 'Durma_D ' 'Mwalibad_D ' 'MjgaonRd_D '
'KKndrDPP_D ' 'LFRoad_D ' 'RgvdrDPP_D ' 'TherSRT_D ' 'Srvdycwk_D '
'NngrgnRd_D ' 'Raghogrh_D ' 'Banshkri_DC ' 'Mahindra_D ' 'MrenTirh_D '
'MrktYrd_DC ' 'ICDCant_D ' 'JydevNGR_D ' 'BsstdDPP_D ' 'Chandanagar_Dc '
'Arulimod_D ' 'Chandmari ' 'LSRoad_DC ' 'Barwala ' 'Katora_IP '
'BasthDPP_D ' 'AsnsdhRD_D ' 'KeranDPP_D ' 'SourvDPP_D ' 'NagpurRd_D '
'Dhelu_D ' 'Lovely_D ' 'BhwanDPP_D ' 'HghsclRD_D ' 'Blbgarh_DC '
'Chpaguri_D ' 'Chakan_D ' 'Ramvlg_D ' 'PreetDPP_D ' 'ShsmlDPP_D '
'South_R_11 ' 'Domlur ' 'YuktiDPP_D ' 'Agraroad_I ' 'Chndrlpd_D '
'Shivaji_I ' 'Kooriyad_D ' 'Mandodi_D ' 'Chuanpur_I ' 'SuzkiSrv_D '
'Chithbrm_D ' 'Sabalpur_D ' 'Khjurwli_DC ' 'Bahreya_I ' 'Subshngr_D '
'Bargawan_DC ' 'Perungudi_DPC ' 'Khwssrai_D ' 'North_I_4 ' 'Central_D_7 '
'Patel Nagar ' 'SidculRd_D ' 'YeolaRD_D ' 'ThthiCwk_D ' 'ShantiNg_D '
'Virar_DC ' 'Trimulgherry_Dc ' 'Surajpur_DC ' 'Peedika_H ' 'KSClny_DC '
'PuranDPP_D ' 'Palikval_D ' 'GrudwrRd_D ' 'IndraCln_D ' 'KrantiNgr_D '
'KolarRd_D ' 'StnRoad_DC ' 'TBCross_D ' 'Pnjbiyon_D ' 'Central_D_5 '
'PiliKoti_D ' 'Shahapur_D ' 'MrdiVlge_D ' 'HawaiPlr_DC ' 'Athithnr_DC '
'TirupDPP_D ' 'Patelfli_D ' 'NCplxDPP_D ' 'BisnolDPP_D ' 'ShrprDPP_D '
'BllvMarg_D ' 'DataSagr_D ' 'Farmnala_D ' 'Bagaluru_D ' 'BaliaMod_D '
'JngidDPP_D ' 'GagiDPP_D ' 'Poothole_D ' 'Tolichwk_L ' 'PnditNGR_D '
'StationRD_D ' 'Kakrmath_D ' 'Arangadi_D ' 'SurbhiTh_D ' 'Talkui_D '
'ModelTwn_P ' 'Panderia_D ' 'Cnsrvila_D ' 'Ukkadam_D ' 'AshkngRd_D '
'Rajula_DC ' 'BndhuTRH_D ' 'Hatpada_D ' 'JNPT_D ' 'NrdawDPP_D '
'Swargash_D ' 'Aswningr_I ' 'RjndraRd_D ' 'Jharia_DC ' 'DltprDPP_D '
'Central_H_2 ' 'Vandalur_Dc ' 'Sadras_D ' 'Bypassrd_D ' 'TrnptNgr_L '
'AzmrdDPP_D ' 'ChrliDPP_D ' 'HunthrVg_I ' 'PmthuKlm_D ' 'PanditRd_D '
'Mnanthla_H ' 'Chtrpuza_D ' 'VidyaNgr_D ' 'VikrmMah_D ' 'StteHW28_D '
'Alngjuri_D ' 'KarnalRd_D ' 'VijywdRD_D ' 'Dehrird_D ' 'NH117_D '
'FulbaDPP_D ' 'NatunDPP_D ' 'Enayetpr_D ' 'East_I_20 ' 'KtnRdDPP_D ' 'D '
'Shivalya_D ' 'Pazhvedu_D ' 'Mapusa ' 'FshryOFC_D ' 'Kothriya_DC '
'BkgnRoad_D ' 'Solaiprm_D ' 'SmbjiCwk_D ' 'Nehru3PL_D ' 'VadaiDPP_D '
'Govndsgr_D ' 'Bangotu_D ' 'BnglorRd_D ' 'Mhdhvpur_D ' 'Central_D_15 '
```

```
'GtRoad_D ' 'Jantaclg_D ' 'Rajpura_D ' 'PngnrRd_D ' 'Greens_D '
'NagarDPP_D ' 'Tejpal_M ' 'RjghatRd_D ' 'Sirjudol_D ' 'KirtiNgr_D '
'JalnaRd_D ' 'West_Dc ' 'Sitarmrd_D ' 'ThaneDPP_D ' 'JatniDPP_D '
'Margao_Dc ' 'Rawatpur_D ' 'Chnglptu_DC ' 'JivanDPP_D ' 'Parigi_D '
'AshokNagar_DC ' 'Ymunpurm_D ' 'RazaviRd_D ' 'BhmprDPP_D ' 'Greenmkt_D '
'HBColny_D ' 'ShbdnDPP_D ' 'Purbari_D ' 'Brpc ' 'ITICollg_L '
'HnmntNgr_D ' 'Salanpur_D ' 'BtaiRoad_D ' 'Thiruviz_D ' 'SikriKla_DC '
'AmtlaDPP_D ' 'Rathnam_D ' 'UtBzrDPP_D ' 'Gurudwar_D ' 'FatprDPP_D '
'RajRdDPP_D ' 'Shivprsad_D ' 'LxmiNiws_D ' 'Pthrgoan_D ' 'SsnRdDPP_D '
'BalibDPP_D ' 'Mangalam_D ' 'Cdosclrd_D ' 'KcharaRD_D ' 'PBRDDPP_D '
'ElngoNgr_C ' 'SagarDPP_D ' 'Shankrpa_D ' 'PchpkrRD_D ' 'DmodrNGR_D '
'Tuminkte_D ' 'Ponda_Dc ' 'Wardno6_D ' 'Paldi_D ' 'PaikjNGR_D '
'Babupaty_D ' 'Basni_I ' 'KoralDPP_D ' 'Nerul_D ' 'Uppal_Dc '
'SashPhkn_D ' 'Bomsndra_L ' 'KaimgnjRD_D ' 'SukntDPP_D ' 'Arsprmbu_D '
'Ganesh_D ' 'SainkSCL_D ' 'MnimlaRd_D ' 'ChatiDPP_D ' 'MdnprDPP_D '
'Sudmpuri_D ' 'East ' 'MduraiRD_D ' 'Mthrapuri_D ' 'MaxDPP_D '
'MandyaRD_D ' 'Kntgorya_D ' 'KhirByps_I ' 'Kalyanpr_D ' 'Sarjapur_D '
'SbhRDDPP_D ' 'NkshtrPz_D ' 'Central_D_4 ' 'PnjPiara_D ' 'Chrwpaty_D '
'Sangetha_D ' 'KetyDPP_D ' 'Bngisheb_D ' 'Vepmpttu_DC ' 'NagplDPP_D '
'BOB_D ' 'Vaghasi_IP ' 'Mahad_D ' 'LamtiDPP_D ' 'MunplDPP_D '
'AgrohDPP_D ' 'NditaDPP_D ' 'Kidwai_D ' 'AnugrDPP_D ' 'NaturDPP_D '
'Umargaon_DC ' 'Dakor_DC ' 'Shop3DPP_D ' 'SChwkDPP_D ' 'DiyoDPP_D '
'SirsaDPP_D ' 'MathuraRD_D ' 'Muktsar_D ' 'CivilStn_D ' 'GVManu_D '
'Robinson_D ' 'Patia ' 'Rjndpara_D ' 'KairiyaT_D ' ' Kothanur_L '
'Ricco_D ' 'Malegaon_D ' 'Palam ' 'CrossRD_D ' 'Gondkhry_H '
'South_D_20 ' 'Central_H_4 ' 'KdrShrRd_D ' 'ArickDPP_D ' 'DhuleRoad_D '
'Bhaleti_D ' 'Thvlrsrt_D ' 'STRdDPP_D ' 'BhaRDDPP_D ' 'SaiTempl_D '
'DBRCmplx_D ' 'SelamRd_D ' 'PhrmPlza_D ' 'Mirapati_L ' 'Sarubali_D '
'MSRClgRd_D ' 'RhmgjDPP_D ' 'Central_L_8 ' 'Amankovl_D ' 'Katrmira_D '
'Kaura_D ' 'GwhRDDPP_D ' 'SKRoad_D ' 'Khndyusn_D ' 'Nelmngla_L '
'TilakNgr_D ' 'ByePass_D ' 'PlaceCol_D ' 'IdstrlAr_D ' 'GovndNgr_DC '
'AryaNagr_D ' 'Karelibaug_DC ' 'Udyabata_D ' 'MbRoad_D ' 'Ward25_D '
'Kalol_DC ' 'HousngBd_D ' 'Central_D_10 ' 'Old ' 'KalikDPP_D '
'SliprDPP_DC ' 'Kdvantra_D ' 'Gangjala_D ' 'SadulDPP_D ' 'Pulgaon_DC '
'AkhirDPP_D ' 'ChainDPP_D ' 'KrshnNgr_D ' 'Kolar Mandakni ' 'Ramnagar_D '
'Selaiyur_DC ' 'Chinchwad DC ' 'VishnuVhr_D ' 'HotelPrk_D ' 'Indsarea_D '
'MnbzrDPP_D ' 'Mahuva_DC ' 'Manchar_D ' 'Fathuluh_D ' 'Achipkam_D '
'StnRdDPP_D ' 'JmnvadRd_DC ' 'NapitDPP_D ' 'GModDPP_D ' 'RgstrOFC_D '
'RajpurRD_D ' 'Mharajpr_D ' 'NarenaRD_D ' 'WrdN1DPP_D ']
----------------------------------------------------------------------
------------------
destination_state
['Uttar Pradesh' 'Karnataka' 'Haryana' 'Maharashtra' 'Tamil Nadu'
 'Gujarat' 'Delhi' 'Telangana' 'Rajasthan' 'Madhya Pradesh' 'Assam'
 'West Bengal' 'Andhra Pradesh' 'Punjab' 'Chandigarh'
 'Dadra and Nagar Haveli' 'Orissa' 'Bihar' 'Jharkhand' 'Goa' 'Uttarakhand'
 'Himachal Pradesh' 'Kerala' 'Arunachal Pradesh' 'Mizoram' 'Chhattisgarh'
 'Jammu & Kashmir' 'Nagaland' 'Meghalaya' 'Tripura' 'Daman & Diu']
----------------------------------------------------------------------
------------------
destination_city
['Kanpur' 'Doddablpur' 'Gurgaon' 'Mumbai' 'Sandur' 'Chennai'
 'HBR Layout PC ' 'Surat' 'Delhi' 'PNQ Rahatani DPC ' 'Faridabad '
 'Ratnagiri' 'Bangalore' 'Hyderabad' 'Aland' 'Jaipur' 'Satna' 'Janakpuri '
 'Guwahati' 'Bareli' 'Nashik' 'Hooghly' 'Puttaprthi' 'Sivasagar'
 'Bengaluru' 'Palani' 'Jalandhar' 'Chandigarh' 'Yavatmal' 'Sangola'
 'Kolkata' 'Savner' 'Kurnool' 'FBD' 'Bhatinda' 'Bhiwandi' 'Barnala'
 'Murbad' 'Kadaba' 'Gulbarga' 'Naraingarh' 'Ludhiana' 'Kadi' 'Jabalpur'
 'MAA' 'Gangakher' 'Bankura' 'Silvassa' 'Porbandar' 'Jetpur' 'Khammam'
 'Mehsana' 'Katni' 'Una' 'Malavalli' 'HDKote' 'Radhanpur' 'Visakhapatnam'
 'Pune' 'Bhopal' 'Bhubaneshwar' 'Allahabad' 'Sonipat' 'Himmatnagar'
 'Sasaram' 'Ranchi' 'Thiruvarur' 'GZB' 'Anand' 'Nanded' 'Noida' 'Nadiad'
 'Virudhchlm' 'Durgapur' 'Bhopal MP Nagar ' 'Bhadrak' 'Goa' 'Balurghat'
 'Hisar' 'Tiruppattur' 'Kotdwara' 'Mumbai Hub ' 'Yellareddy' 'Halvad'
```

```
'Hospet ' 'JognderNgr' 'ChandroknaRD' 'Kirauli' 'BLR' 'Dhaurahara'
'Canacona' 'Vansda' 'Mananthavady' 'Lucknow' 'Silchar' 'Bhuj' 'Pundibari'
'PNQ Pashan DPC ' 'LowerParel' 'Changlang' 'Boisar' 'Tezpur' 'Koduru'
'GGN' 'Gudur' 'Pen' 'CCU' 'Amdavad' 'AMD' 'Akbarpur' 'Ahmedabad' 'Purnia'
'Aurangabad' 'Faridabad' 'Anantapur' 'KN' 'Kolhapur' 'Sausar' 'Haldia'
'Dindigul' 'Namakkal' 'Erode ' 'Jaipur ' 'Parvathipuram' 'Srikakulam'
'Nalasopara ' 'Pathankot' 'Malda' 'Malvan' 'Shajapur' 'Ambabadi' 'OK'
'PNQ Vadgaon Sheri DPC ' 'Amritsar' 'Coimbatore' 'Jasai' 'Villupuram'
'Mettur' 'Palwal ' 'Darjeeling' 'Tiruchi' 'Dadri' 'Del' 'Gotan' 'Amroha'
'Datia' 'Dhanbad' 'Guna' 'Burhanpur' 'Mangalore' 'Margherita' 'Chamoli'
'Ajmer' 'Pasighat' 'Mirzapur' 'Ghazipur' 'Hubli' 'Bagalkot' 'Robertsganj'
'Haveri' 'Alwar' 'Udaipur' 'Gandhidham' 'Solapur' 'Belgaum' 'Moga'
'Kendrpara' 'Barshi' 'Addanki' 'Ongole' 'Bokaro' 'Sagara' 'Deoband'
'Chhatarpur' 'Siwan' 'Rajgir' 'Thrissur' 'Mandya' 'Rishikesh'
'Manjeshwar' 'Jamshedpur' 'Bakhtiarpur' 'Dahod' 'Tirupur' 'Karanjia'
'Neemrana' 'Ganga Nagar' 'Arwal' 'Bhiwani' 'Kolasib' 'Midnapore' 'Sillod'
'Nellore' 'Aurangabad ' 'Baharampur' 'Rawatsar' 'Kaithal' 'Kaikaluru'
'Machilipatnam' 'Nazirpur' 'Kalwakurthy' 'Puranpur' 'Jorhat' 'Mandi'
'Rajamundry' 'Chitradurga' 'Draksharamam' 'Muzaffrpur' 'Akola' 'Islampur'
'Madhepura' 'Simrahi' 'Srisailam' 'Bngnpalle' 'Tiptur' 'Dandeli'
'Bijapur' 'Patiala' 'Bijainagar' 'Channaraya' 'Katihar' 'Ratia' 'Makrana'
'Raigarh' 'Almora' 'Godda' 'Bayana' 'Kushinagar' 'Dhaka' 'Kawardha'
'Bahadurgarh' 'Dhampur' 'Gorakhpur' 'Warangal' 'Sambhal' 'Ratlam'
'Rudrapur' 'Sahatwar' 'Balaghat' 'Raxaul' 'Narayankhed' 'Kalyandurg'
'Samana' 'Shamli' 'Gangapur' 'Pilani ' 'Dwarka' 'Kakdwip' 'Ambah'
'Attingal' 'Surendranagar' 'Buxar' 'Anupshahar' 'Kallikkad' 'Auraiya'
'Bhagalpur' 'Panaji Goa ' 'Raikot' 'Hapur' 'Samastipur' 'Kaman'
'DhrmpuriTS' 'Mancherial' 'Haripad' 'Mundakayam' 'Kollam' 'Shahada'
'AurngbadBR' 'Kanti' 'Chamorshi' 'Pandharpur' 'Karimnagar' 'Zirakpur'
'Unnao' 'Aluva' 'Kannad' 'Latur' 'TalwandiSabo' 'Ghatampur' 'Nalgonda'
'Banda' 'Konch' 'Dibrugarh' 'Mussoorie' 'Jhargram' 'Tarkeshwar'
'Kalpetta' 'Phalodi' 'Tekkali' 'Sidhi' 'Bilimora' 'Dinhata' 'Amd'
'Jalgaon' 'Vadodara' 'Hyd' 'Dhubri' 'Vadodara ' 'Dhule' 'Sholinghur'
'Rajgurunagar' 'Hassan' 'Karnaprayag' 'Tangi' 'Sirsi' 'Bailhongal'
'Sikar' 'Gonda' 'Madurai' 'Banswara' 'Ghosi' 'Paota' 'Guruvayoor' 'Attur'
'Polur' 'Loharu' 'Ankola' 'Karkala' 'Hanumangarh' 'Tumkur' 'Kendujhar'
'Alappuzha' 'Kuthuparamba' 'Thirukkatupli' 'Gudalur' 'Devarakonda'
'Ponnamaravathi' 'Karimganj' 'Khed' 'Lalpet' 'Kalka' 'Saharsa' 'Pupri'
'Rohtak ' 'Rajpalayam' 'Bina' 'Ramanathapura' 'Meerut ' 'Amalapuram'
'Bettiah' 'Chintamani' 'Bethamangala' 'Pollachi' 'Jagdishpur' 'Khandela'
'Sikandarpur' 'Motihari' 'Dharapuram' 'Dinara' 'Nawalgarh' 'Champa'
'Bansi' 'Arakkonam' 'Hoskote' 'Nedumangad' 'Rayaparthi' 'Tirunelveli'
'Amreli' 'Tirchchndr' 'Kusumnchi' 'Deoghar' 'Jamtara' 'Bhupalpally'
'Husnabad' 'Narsinghpur' 'Ramagundam' 'Aligarh' 'Ghaziabad' 'Gwalior'
'Sakri' 'Haldwani' 'Chabua' 'Thiruvadanai' 'Manmad' 'Siruguppa'
'Central Goa ' 'Mahasamund' 'Aonla' 'Salem ' 'Bamangola' 'Moradabad'
'CoochBehar' 'Bhalukpong' 'NOI' 'Mumbai Antop Hill ' 'Jammu' 'Medchal'
'Perundurai' 'Marakkanam' 'Bhusawal' 'Vapi' 'Berhampur' 'Balasore'
'Didwana' 'Jagatsghpr' 'Bantwal' 'Achrol' 'Gokak' 'Gopalganj'
'Vadakkencherry' 'Edappal' 'Jhabua' 'Trivandrum' 'Rampur' 'Pali' 'Shirur'
'Mokokchung' 'Jalna' 'Jeypore' 'JoguGadwal' 'Paramakudi' 'Badnaur'
'Patancheru' 'Merta' 'Benipur' 'Jangipur' 'Shegaon' 'Fatehabad' 'Supaul'
'Manjhaul' 'Sakleshpur' 'Sathyamangalam' 'Ooty' 'HazratJandaha'
'Machhiwara' 'Kaptanganj' 'Davangere' 'Lonavala' 'Baraut' 'NeemKaThana'
'DehriSone' 'Bhind' 'Sathupally' 'Malegaon' 'Madhupur' 'Bhavnagar'
'Shindkheda' 'Sangareddy' 'Phulera' 'Chhaygaon' 'Kopargaon' 'Raipur'
'Asifabad' 'Chinnur' 'Bishnupur' 'Basti' 'Nakodar' 'Mansa' 'Kashipur'
'Dola' 'Kodaikanal' 'Patan' 'Thirumalagiri' 'Lakhnadon' 'Bobbili'
'Phulpur' 'SultnBthry' 'BilaspurHP' 'Mahad' 'Srivijaynagar' 'Ashta'
'Pachore' 'Hajo' 'Tulsipur' 'Chopan' 'Shillong ' 'Vinukonda'
'North Delhi ' 'Sujangarh' 'Shimoga' 'Muktsar' 'Molakalmuru' 'Satara'
'Joda' 'Narnaul' 'Nandigama' 'Sidhmukh' 'Printhlmna' 'Kekri' 'Katwa'
'Nabarangpr' 'Pithorgarh' 'Bareilly ' 'Perambalur' 'Dighwara' 'Kandi'
'Lalgola' 'Karnal ' 'Badarpur' 'Patiala ' 'Bariya' 'Bharatpur' 'Jagraon'
```

```
 'Rajpura' 'Nandurbar' 'Budhana' 'Kottayam' 'Rath' 'Shahdol' 'Karauli'
 'Khurdha' 'Hura' 'Bellary' 'Gonikoppal' 'Dhrangadhra' 'Anakapalle'
 'Duliajan' 'Phagwara' 'Kamareddy' 'Kalpakkam' 'Dohrighat' 'Dhekiajuli'
 'Kanigiri' 'Ramgarh' 'Dharuhera' 'Arrah' 'Madhubani' 'Narsingpur' 'Rehli'
 'DehraGopipur' 'Pangodu' 'Pappadahandi' 'Saraiya' 'Dumka' 'PaliBirsighpr'
 'Punalur' 'Sujanpur' 'Bagnan' 'Fatepur' 'Rajkot' 'Ambasamdrm' 'Bagepalli'
 'Metpally' 'Mohania' 'Ratanpura' 'Kasaragod' 'Moodbidri' 'Manvi'
 'Khedbrahma' 'Bhanvad' 'Sawantwadi' 'Jalalpur' 'Veraval' 'Pratapgarh'
 'Silapathar' 'Chandi' 'Cochin' 'Arimbur' 'Sheikhpura' 'Chalakudy'
 'Tandur' 'Kotagiri' 'Nowda' 'Vijayawada' 'Benipatti' 'Padrauna' 'Chaksu'
 'Panskura' 'Chimkurthy' 'Giridih' 'Botad' 'Udgir' 'Junagadh' 'Tezu'
 'Jaisalmer' 'Islampure' 'Pune Balaji Nagar ' 'Agartala' 'Mainpuri'
 'Kathua' 'Chandauli' 'Dharwad' 'Aizawl' 'Uchila' 'Tikamgarh' 'Beed'
 'Koraput' 'Karad' 'Mannargudi' 'Dhone' 'Buldhana' 'Parwanoo' 'Kandukur'
 'Morgram' 'Mungeli' 'Theni' 'PaontSahib' 'Bilaspur' 'Areacode' 'Lalru'
 'Kaliyaganj' 'Paranpur' 'Sihora' 'Shivpuri' 'Nagarcoil' 'Gondal' 'Tirpur'
 'Manuguru' 'Bhota' 'Dhrmsthala' 'Jewar' 'Tonk' 'Rghunthpur' 'Pavagada'
 'Puttur' 'Sinnar' 'Bhandara' 'Bolpur' 'Parbhani' 'Suratgarh' 'Gundlupet'
 'Bodhan' 'Chidambaram' 'Agra ' 'Gangavathi' 'Palakonda' 'Palasa'
 'Kharagpur' 'Tirupati' 'Krishnagiri' 'Erandol' 'LakhimpurN' 'Gopiganj'
 'Baripada' 'Khagaria' 'Jehanabad' 'Palamaner' 'Chanapatna' 'Nohar'
 'Asansol' 'Umerkote' 'Chapra' 'Gangarmpur' 'Aranthangi' 'Shamshabad'
 'Kullu' 'Jalalabad' 'Khanna' 'Kalluvathukal' 'Samsi' 'Hoogly' 'Wankaner'
 'Oriyur' 'Vizianagaram' 'BariSadri' 'Pilibanga' 'Vishakhapatnam '
 'Lodhan' 'Moranhat' 'Barauni' 'Dholpur' 'Ghanpur' 'Mangaldoi' 'Gahmar'
 'Chiraiyakot' 'ChrkhiDdri' 'Baddi' 'Degana' 'Sultana' 'Parakkdavu'
 'Seoni' 'Anupgarh' 'Sindagi' 'Sedam' 'Nakhatrana' 'MirzapurWB'
 'Mathabhang' 'Khatra' 'Jodhpur' 'Champhai' 'Rona' 'Shahganj' 'Jowai'
 'Chittaurgarh' 'Arani' 'Jhajjar' 'Malappuram' 'Kallachi' 'Modinagar'
 'Atmakur' 'Jayamkondan' 'Berhampore' 'Ramnagar' 'Buhana' 'Kahalgaon'
 'Patran' 'Deoria' 'Gadchiroli' 'Neemuch' 'Deoli' 'Sonari' 'Digboi'
 'Namsai' 'Sitamau' 'Churhat' 'Dahanu' 'Chandigarh ' 'Khanapur'
 'Balrampur' 'Varanasi' 'Mandapeta' 'Araria' 'Nuzvid' 'Helencha'
 'Bangarapet' 'Khambhalia' 'Jagtial' 'Jammikunta' 'Soro' 'Contai' 'Anjar'
 'Howrah' 'Lakhipur' 'Chamba' 'Piparcity' 'Mau' 'Ramnthpurm' 'Mehkar'
 'SundarNgr' 'Malerkotla' 'Jadcherla' 'Kasganj' 'Athani' 'Mahbubabad'
 'Manikchak' 'Umaria' 'Karukachal' 'Jalore' 'Vemulawada' 'Jabalpur '
 'Koppa' 'Khanakul' 'Mandsaur' 'Ranipet' 'Jairampur' 'Dhemaji' 'Sirsa'
 'Parbatsar' 'Rajgangpur' 'Bargarh' 'Khanpur' 'Sirohi' 'Chalisgaon'
 'Dabhoi' 'RampuraPhul' 'Bhilad' 'Bhatkal' 'Agra' 'Salem' 'Betnoti'
 'Raichur' 'Ranikhet' 'Cjb' 'Markapur' 'Balangir' 'Panipat ' 'Dharmapuri'
 'Lalitpur' 'Modasa' 'Jasdan' 'Aliganj' 'Khalilabad' 'Nagapttinm' 'Jahu'
 'Akhnoor' 'Jassur' 'Nagpur' 'Aunrihar' 'Dehradun ' 'Rayadurgam' 'Razole'
 'Jhanjharpur' 'Nainital' 'Firozabad' 'Bongaon' 'Sumerpur' 'Valsad'
 'Gangarampr' 'Gujiliam' 'Gomoh' 'Arambag' 'SrinagarUK' 'Phusro' 'Teok'
 'Shadnagar' 'Vadakara' 'Sultanganj' 'Atapadi' 'RoopNagar' 'Giddarbaha'
 'Barmer' 'Chodavaram' 'Kittur' 'Vellore' 'Jind' 'Koyilandy' 'Tirurangadi'
 'KharagpurBR' 'Kolar' 'Raver' 'Varanasi ' 'Paradip' 'Khatauli'
 'Kozhenchery' 'Chandpur' 'Kattappana' 'Rajgarh' 'Ambegaon' 'Udumalpet'
 'Raiganj' 'Mothkur' 'Nirsa' 'Venktagiri' 'Manbazar' 'Udala' 'Cuttack'
 'Sonepur' 'Faridpur' 'Dumraon' 'Kanker' 'Kakinada' 'Luxettipet'
 'Bellmpalli' 'Chanchal' 'Shirpur' 'Oddnchtram' 'Nichlaul' 'Mysore'
 'Kodad' 'Khambhat' 'Umreth' 'Tilhar' 'Chetpet' 'Rewari' 'Cuddapah'
 'Pazhayannur' 'Sundargarh' 'Baruipur' 'Anandnagar' 'Khetri' 'Manthani'
 'Thakurdwara' 'Malout' 'Chincholi' 'Daman' 'Uthangarai' 'Gosainganj'
 'Chikblapur' 'Farrukhbad' 'Durg' 'Thachnttukra' 'Chikodi' 'Ranaghat'
 'Munger' 'Bijnor' 'Lunawada']
--------------------------------------------------------------------------------
-----------------
destination_place
['Central_H_6 ' 'ChikaDPP_D ' 'Bilaspur_HB ' 'MiraRd_IP ' 'WrdN1DPP_D '
 'Poonamallee ' 'Vandalur_Dc ' '' 'Central_D_3 ' 'Bhogal ' 'MjgaonRd_D '
 'Nelmngla_H ' 'Uppal_I ' 'RazaviRd_D ' 'Central_I_7 ' 'Central_I_2 '
 'Hub ' 'SourvDPP_D ' 'Varachha_DC ' 'TgrniaRD_I ' 'DC ' 'Gokulam_D '
```

```
'Babupaty_D ' 'Bomsndra_HB ' 'Alwal_I ' 'RjndraRd_D ' 'Mehmdpur_H '
'Sanpada_I ' 'JajuDPP_D ' 'Central_DPP_2 ' 'Dankuni_HB ' 'Wagodha_D '
'AbbasNgr_I ' 'Balabhgarh_DPC ' 'DPC ' 'Mankoli_HB ' 'Shamshbd_H '
'SnkunDPP_D ' 'Kharar_DC ' 'AnugrDPP_D ' 'Nehrugnj_I ' 'Ward2DPP_D '
'MilrGanj_HB ' 'KaranNGR_D ' 'Adhartal_IP ' 'Poonamallee_HB '
'Busstand_D ' 'BhowmDPP_D ' 'Samrvrni_D ' 'NSTRoad_I ' 'Panchot_IP '
'Bargawan_DC ' 'KGAirprt_HB ' 'Mamlatdr_DC ' 'SulthnRd_D ' 'Jogeshwri_L '
'BegurRD_D ' 'Santalpr_D ' 'Gajuwaka_IP ' 'Tathawde_H ' 'Trnsport_H '
'Central_H_1 ' 'Kundli_H ' 'Rohini_DPC ' 'Bypasrd_D ' 'Mohan_Nagar_DPC '
'Madhavaram_L ' 'Vaghasi_IP ' 'Aswningr_I ' 'Sec 02_DPC ' 'SelamRd_D '
'Central_I_1 ' 'Porur_DPC ' 'Perungudi_DPC ' 'AkhirDPP_D ' 'IndstlAr_I '
'Raiprvlg_L ' 'Jhilmil_L ' 'KoilStrt_D ' 'Nzbadrd_D ' 'JKRoad_D '
'Mayapuri_PC ' 'Hoodi_IP ' 'CrossRD_D ' 'Dhelu_D ' 'Central_DPP_3 '
'AchneraRD_D ' 'JPNagar_Pc ' 'KHRoad_I ' 'TahsilRD_D ' 'Kishangarh_DPC '
'CharRsta_D ' 'CottonGreen_DPC ' 'CikhliRD_D ' 'PunjabiB_L '
'Central_D_1 ' 'Kengeri_IP ' 'Indira Nagar ' 'Peenya_IP ' 'Sirikona_H '
'Khandeshwar_Dc ' 'Alwal_L ' 'StatonRD_D ' 'CP ' 'OstwlEmp_D '
'Mhbhirab_D ' 'MGRoad_D ' 'Bngisheb_D ' 'Sector63_L ' 'BljiMrkt_D '
'Bnnrghta_L ' 'Beliaghata_DPC ' 'Airport_H ' 'Lake Avenue_DPC ' 'East '
'Memnagar ' 'Mumbra_DC ' 'Satellite ' 'Auliyapr_D ' 'Ulhasngr_DC '
'East_H_1 ' 'Pawane_L ' 'Kalyan West _Dc ' 'Central_H_2 ' 'Mthurard_L '
'New Alipore_DPC ' 'KamaStrt_I ' 'RPC ' 'Peenya_L ' 'Shivaji_I '
'Central_DPP_1 ' 'Central_D_2 ' 'Central_D_12 ' 'Rkcomplx_DC ' 'Mohali '
'Chrompet_L ' 'Central_D_9 ' 'Chrompet_DPC ' 'Kuslpram_I ' 'Sixmile '
'Chandmari ' 'VarunCly_DC ' 'krshnPly_DC ' 'BllvMarg_D ' 'Central_D_10 '
'MhpraRD_D ' 'NgrNigam_DC ' 'Egmore_DPC ' 'Nangli_IP ' 'Karayam_H '
'JNPT_D ' 'Lajpat_IP ' 'Thirumtr_IP ' 'Madhavaram_DPC ' 'RTOroad_D '
'Hillcard_DC ' 'Samyaprm_D ' 'Blbgarh_DC ' 'Manesar ' 'ICDCant_D '
'B_RPC ' 'DKLogDPP_D ' 'SamitiRd_D ' 'Kundli_P ' 'TrtllaRD_L '
'TownDPP_D ' 'Kalynpur_I ' 'Raghogrh_D ' 'StRoad_D ' 'Kuntikna_H '
'CivilHPL_D ' 'CGRoad_D ' 'FoySGRRD_I ' 'NwYlhnka_DC ' 'MissonRd_D '
'JangiRd_D ' 'Kaithwal_D ' 'Adargchi_IP ' 'Jogshwri_I ' 'Sector4_D '
'ArtoDPP_D ' 'GuttalRD_D ' 'Mangri_I ' 'Sector1A_IP ' 'Dc ' 'Shamshbd_P '
'Dankuni_P ' 'Kapleswr_D ' 'Okhla_PC ' 'Nangli_L ' 'Kurduwdi_D '
'Oilmilrd_D ' 'SubhVRTL_I ' 'HUB ' 'Patparganj_DPC ' 'Chndivli_PC '
'KSClny_DC ' 'Vardhard_D ' 'Mankoli_GW ' 'Chrompet_PC ' 'Bilaspur_P '
'Pandesra_Gateway ' 'Ramvlg_D ' 'HnmntNgr_D ' 'PnditNGR_D ' 'Poothole_D '
'Tolichwk_I ' 'Central_I_3 ' 'BypassRD_D ' 'Kapshera_L ' 'RIICO_L '
'Koliplm_I ' 'Sarubali_D ' 'Rcocmplx_D ' 'Rozapar_D ' 'Diakkawn_D '
'Talkui_D ' 'ZebaTWR_D ' 'North_R_8 ' 'Chuanpur_I ' 'ShivmDPP_D '
'Atapaka_D ' 'VidyaNGR_D ' 'Mundhawa_L ' 'MdothdRD_D ' 'RicMilRd_D '
'PlaceCol_D ' 'AtoNgrRd_I ' 'Anaipeta_D ' 'Bbganj_I ' 'Gaurkshn_I '
'Shantanu_D ' 'Wazirpur_L ' 'Krishnpr_D ' 'Bazar_D ' 'Sishumdr_D '
'Enkndla_D ' 'YTRd_D ' 'RKComplx_D ' 'KirtiNgr_D ' 'GainMrkt_L '
'patna_D ' 'Shyndco_D ' 'SchdvDPP_D ' 'Mwalibad_D ' 'MithmdRd_D '
'SuzkiSrv_D ' 'PushPlza_D ' 'LalBagh_D ' 'KasyaDPP_D ' 'East_I_21 '
'PchpkrRD_D ' 'TrnsptNGR_D ' 'NaginaRD_D ' 'Matriprm_IP ' 'HunterRd_I '
'Khwssrai_D ' 'Khjurwli_DC ' 'North_I_4 ' 'UdhamNgr_H ' 'PnchmDPP_D '
'Kosmi_D ' 'KairiyaT_D ' 'Datatrya_D ' 'Shankrpa_D ' 'PODPP_D '
'KarnalRd_D ' 'Ward6DPP_D ' 'StnRoad_DC ' 'KlngrDPP_D ' 'MrenTirh_D '
'Tejpal_I ' 'Sarswati_D ' 'DcntCLY_D ' 'Mutyvila_D ' 'AryaNagr_D '
'Kharghar_D ' 'Pbroad_DC ' 'Swargash_D ' 'Haripur_D ' 'BsstdDPP_D '
'HanumDPP_D ' 'Hitech_D ' 'Kumrpurm_D ' 'Nandrbar_D ' 'Mahindra_D '
'Khar West _Dc ' 'RajCmplx_D ' 'KamnHbRD_I ' 'VikasRam_D ' 'Peedika_H '
'KolheDPP_D ' 'Srnwsngr_D ' 'Wardno3_D ' 'StatinRD_D ' 'Rawatpur_D '
'HydRoad_DC ' 'GayatriN_D ' 'GovndNgr_DC ' 'Bokule_H ' 'BrlwgDPP_D '
'Chikdply_I ' 'UBamdDPP_D ' 'Naraynpr_D ' 'Ward19_D ' 'PalikDPP_D '
'Phaphamu_DC ' 'AmvdiDPP_D ' 'Padra_D ' 'WrdN4DPP_D ' 'Chandkheda_Dc '
'Skynet_INT ' 'Vepmpttu_DC ' 'Karelibaug_DPC ' 'Paschim_DC '
'LB-Nagar_Dc ' 'CotnGren_M ' 'MiraRoad_M ' 'Tetultol_D ' 'MIDCAvdn_I '
'ArkonmRD_D ' 'Chakan_D ' 'Pandrnga_I ' 'Umalodge_D ' 'SriDPP_D '
'Vidygiri_D ' 'NamoNagr_D ' 'FatehpRd_I ' 'KotwaliN_D ' 'Kappalur_H '
'KhandDPP_D ' 'Jamalpur_D ' 'SmClyDPP_D ' 'ManhrBld_D ' 'DumDum_DPC '
'KaaduRd_D ' 'StationRD_D ' 'BstndDPP_D ' 'Kakrmath_D ' 'MarketRd_D '
```

```
'Veersagr_I ' 'Sirjudin_D ' 'Pazhvedu_D ' 'IdstrlAr_D ' 'Poondi_D '
'kalmpuza_D ' 'DindiRD_D ' 'Puduvalvu_D ' 'Alngjuri_D ' 'Mahad_D '
'PriyrNGR_D ' 'Pinjore_DC ' 'Gangjala_D ' 'SngihiRD_D ' 'AshkTalk_D '
'Srvdycwk_D ' 'Vijdurg_D ' 'BypassRd_D ' 'FshryOFC_D ' 'Venkatsa_DC '
'Wardno13_D ' 'PlsrdDPP_D ' 'RajaBzr_D ' 'Techrcly_D ' 'Wardno7_D '
'NavldiDPP_D ' 'Brplicwk_D ' 'GangDPP_D ' 'Banshkri_DC ' 'HousngBd_D '
'Arsprmbu_D ' 'Perkadrd_D ' 'VdkkuSrt_I ' 'Rajula_DC ' 'Shnmgprm_D '
'SKRoad_D ' 'Uppal_L ' 'Barmasia_D ' 'D ' 'JwahrNGR_D ' 'Greenmkt_D '
'KndliDPP_D ' 'Pdmavati_D ' 'KhirByps_I ' 'RjndrNgr_DC ' 'HrihrNgr_I '
'DhuleRoad_D ' 'PiliKoti_D ' 'CollgeRD_D ' 'North ' 'Thiruvlr_DC '
'RamnadRD_D ' 'Malegaon_D ' 'South_D_12 ' 'Central_D_7 ' 'Wrd12DPP_D '
'Ponda_Dc ' 'RajpurRD_D ' 'KdidmCLY_D ' 'Psthrjhr_D ' 'PC ' 'Khenewa_D '
'TrnptNgr_L ' 'MROoffce_D ' 'Trimulgherry_Dc ' 'Panvel_D ' 'Viveka_DC '
'MJRDPP_D ' 'Central_I_4 ' 'Samarth_D ' 'IndEstat_I ' 'Khajuria_I '
'Ganeshwr_D ' 'KatlaDPP_D ' 'Markndpr_D ' 'Trmltmpl_D ' 'Mehmdpur_P '
'BgwriDPP_D ' 'Bsavangr_D ' 'GopalDPP_D ' 'Robinson_D ' 'PonaniRD_D '
'Ward11_D ' 'Mnanthla_H ' 'RoshnBgh_I ' 'Nayagaon_I ' 'BhgyaNgr_D '
'Kelasahi_D ' 'ColctrOf_D ' 'VagaiNgr_D ' 'BhmrdDPP_D ' 'Ameenpur_I '
'Javahar_D ' 'SChwkDPP_D ' 'BhunaDPP_D ' 'SadrHsptl_D ' 'Purbari_D '
'RgvdrDPP_D ' 'Mlydpthr_D ' 'Davisdle_D ' 'HajiprRD_D ' 'MnBzrDPP_D '
'Subshngr_D ' 'NngrgnRd_D ' 'SrnprHwy_D ' 'Margao_Dc ' 'Tejpal_M '
'War5DPP_D ' 'Dilliyan_D ' 'BhrolDPP_D ' 'Rawlgaon_D ' 'Sitarmrd_D '
'Kadugodi_D ' 'Mahuva_DC ' 'Shahdara ' 'KakaCplx_D ' 'Pothredy_D '
'NarenaRD_D ' 'Sriperumbudur_Dc ' 'GwhRDDPP_D ' 'Kalyan ' 'NkshtrPz_D '
'Barwala ' 'Central_D_5 ' 'PaikjNGR_D ' 'AsnsdhRD_D ' 'StnRdDPP_D '
'GndhiNgr_D ' 'ChowkDPP_D ' 'GreenVly_D ' 'Vaishali_D ' 'ChainDPP_D '
'Athithnr_DC ' 'Xroad_D ' 'ColegRd_D ' 'Shekhpur_D ' 'Kollgpra_D '
'Indsarea_D ' 'Govndsgr_D ' 'BhwanDPP_D ' 'ShantiNg_D ' 'Shop3DPP_D '
'Sardala_D ' 'MohnVRTL_D ' 'MohanNgr_C ' 'Manikndm_H ' 'PreetDPP_D '
'Kothapet_D ' 'ChtrGIDC_IP ' 'LdnunDPP_D ' 'Mhdiptnm_C ' 'KnsgraRD_D '
'Bomsndra_PC ' 'Chndrlpd_D ' 'MnbzrDPP_D ' 'EmsPnmbi_D ' 'LxmntDPP_D '
'BSarani_D ' 'PhdofDPP_D ' 'Kumud_D ' 'goplpurm_D ' 'SadarHPL_D '
'DohalDPP_D ' 'KrsprDPP_D ' 'Konapara_D ' 'BgnprDPP_D ' 'DhuleRd_D '
'Mughlpra_D ' 'Sohagpur_D ' 'HnsChowk_D ' 'MdhsnDPP_D ' 'Thomas_D '
'NvygRDPP_D ' 'Kothuru_D ' 'StatonRd_D ' 'Lovely_D ' 'Potheri '
'Devenply_I ' 'Chatrpr_DC ' 'Sadras_D ' 'AzmrdDPP_D ' 'Wardno5_D '
'Tiglgndi_D ' 'HotelPrk_D ' 'Pandriba_L ' 'Katira_D ' 'Bardivan_D '
'BypRDDPP_D ' 'JrjolDPP_D ' 'Ward7DPP_D ' 'East_L_23 ' 'Wardno4_D '
'Dudhani_D ' 'CtyLgDPP_D ' 'PostofJN_D ' 'SainkSCL_D ' 'Harop_D '
'NH117_D ' 'PedakRd_P ' 'Solaiprm_D ' 'TBCross_D ' 'GunjRDPP_D '
'Bhabua_D ' 'MubarDPP_D ' 'Todapur_DC ' 'Nullipad_D ' 'MrdiVlge_D '
'APMCYard_D ' 'Patelfli_D ' 'Mainroad_D ' 'LaxmiNgr_D ' 'HSR_Layout_PC '
'Mhimapur_D ' 'Nimachrd_D ' 'Hejunagr_D ' 'SH78_D ' 'Kdvantra_D '
'Veluthur_D ' 'Bgwtichk_D ' 'SurbhiTh_D ' 'SnthiNGR_D ' 'CroslySRT_D '
'AmtlaDPP_D ' 'Rynapadu_H ' 'WardNo1_D ' 'BawliDPPP_D ' 'ModelTwn_P '
'NraynDPP_D ' 'TonkRoad_D ' 'MSRClgRd_D ' 'Shivalya_D ' 'JatniDPP_D '
'Mangol_DC ' 'Mullanpr_DC ' 'NlgaonRd_D ' 'Farmnala_D ' 'Gopa3PL_D '
'ShbdnDPP_D ' 'Chmpmura_I ' 'Agraroad_I ' 'Sholinganallur_Dc '
'North_D_3 ' 'BOB_D ' 'ShubsNGR_D ' 'HunthrVg_I ' 'MndiRoad_D '
'Sector02_C ' 'JalnaRd_D ' 'GhtimDPP_D ' 'Mundhe_D ' 'Vadasari_D '
'RmNyrDPP_D ' 'Thsil3PL_D ' 'LICOffce_D ' 'Kntgorya_D ' 'Panderia_D '
'Rathnam_D ' 'Gurudwar_D ' 'GrmNgriya_D ' 'Puthalam_D ' 'OnkarDPP_D '
'FatprDPP_D ' 'Nijgan_D ' 'KtnRdDPP_D ' 'AsrplmRd_DC ' 'Mapusa '
'Palladam_DC ' 'AskNagar_D ' 'BpassDPP_D ' 'Beltngdi_D ' 'SJRoad_D '
'barkarRd_D ' 'PnukndRD_D ' 'Darbe_DC ' 'Patel Nagar ' 'KaremDPP_D '
'WebelDPP_D ' 'Rjndrngr_D ' 'Old City ' 'Swamylyt_D ' 'SuryaDPP_D '
'ARBNorth_DC ' 'Aliganj ' 'PhrmPlza_D ' 'VadaiDPP_D ' 'Nerul_D '
'Balajicly_I ' 'BnglorRd_D ' 'BsStdDPP_D ' 'SashPhkn_D ' 'KalikDPP_D '
'GodamDPP_D ' 'Madarpur_D ' 'Lakshmi_D ' 'NagarDPP_D ' 'Whitefld_L '
'KhdimDPP_D ' 'HsptlRod_D ' 'Lngrguda_D ' 'AkhraBzr_D ' 'farukngr_D '
'Pariplly_D ' 'RatuaDPP_D ' 'Bandel_D ' 'JivanDPP_D ' 'Vllyaprm_D '
'NcsRd_DC ' 'BhmprDPP_D ' 'GoalpDPP_D ' 'TiloiDPP_D ' 'Wardno10_D '
'GtRoad_D ' 'Palakrty_D ' 'LNBRoad_D ' 'Wardnor4_D ' 'KamalDPP_D '
'PuranDPP_D ' 'ByePass_D ' 'BhukrdPP_D ' 'KeRoad_D ' 'Kidwai_D '
```

```
'PrmNrDPP_D ' 'KalyanNg_D ' 'DBRCmplx_D ' 'ClgRDDPP_D ' 'VidyaDPP_D '
'Pshimpra_D ' 'Sec-83_DC ' 'KoralDPP_D ' 'Basni_I ' 'AwmpiVng_D '
'GadagRD_D ' 'KcharaRD_D ' 'Ldthlabh_D ' 'KrthiKyn_D ' 'Munduprm_D '
'ZamQuatr_D ' 'SikriKla_DC ' 'IndraNgr_D ' 'Chithbrm_D ' 'BhwniGnj_D '
'CourtDPP_D ' 'NdiaTola_D ' 'MheshNGR_D ' 'CCRoad_D ' 'KarjuDPP_D '
'JyotiNgr_D ' 'HelipadRD_D ' 'KetyDPP_D ' 'PBRDDPP_D ' 'MahmurGj_IP '
'Mainrd_D ' 'Chtrpuza_D ' 'Wardn13_D ' 'Ward17_D ' 'ColnyDPP_D '
'Pettah_D ' 'KolarRd_D ' 'JdswarRD_D ' 'East_I_20 ' 'Aravind_D '
'ConduDPP_D ' 'Mylapore ' 'UttarDPP_D ' 'Kanakpur_D ' 'Salap_DC '
'Kolar Mandakni ' 'Bangotu_D ' 'Kadipur ' 'SohnaRd_D ' 'BaliaMod_D '
'TnhbBlkC_D ' 'SagarDPP_D ' 'Bhogpur_D ' 'Badeplly_D ' 'BnkrGate_D '
'VikrmMah_D ' 'Yellanda_D ' 'AgrohDPP_D ' 'Enayetpr_D ' 'MnimlaRd_D '
'RoopNgr_D ' 'Jharia_DC ' 'Sangetha_D ' 'Ramnagar_D ' 'YashDPP_D '
'East_D_8 ' 'MBTRd_DC ' 'Mdiclcly_D ' 'WardNo3_D ' 'AnprnDPP_D '
'AjmhwDPP_D ' 'JJCpxDPP_D ' 'NehruNGR_D ' 'Ukkadam_D ' 'Nrsampt_D '
'Ricco_D ' 'BhadgDPP_D ' 'Poonamallee_L ' 'Umargaon_DC ' 'KmkshBul_D '
'Idgah_P ' 'Kadtmpty_D ' 'MhliaDPP_D ' 'LSRoad_DC ' 'Central_L_8 '
'Subhash_D ' 'Kondapur_D ' 'Muktsar_D ' 'Ghansoli_DC ' 'Kovaipudur_Dc '
'Lajwanti ' 'VidyaNgr_D ' 'Central_H_4 ' 'Rjndpara_D ' 'RjghatRd_D '
'MotiDPP_D ' 'KaimgnjRD_D ' 'MrgnjDPP_D ' 'Sttyapar_D ' 'Sulgwan_D '
'ThthiCwk_D ' 'AadiDPP_D ' 'Jaripatk_DC ' 'Sarjapur_D ' 'RailGate_D '
'Shanthi_D ' 'Nagar_D ' 'Sookhtal_D ' ' Kothanur_L ' 'Gobindgarh_DC '
'Sudmpuri_D ' 'SukntDPP_D ' 'JiswlDPP_D ' 'West_Dc ' 'BazarDPP_D '
'FulbaDPP_D ' 'Parai_D ' 'KhsmiDPP_D ' 'BalibDPP_D ' 'Srikot_D '
'RhmgjDPP_D ' 'Mandodi_D ' 'Vidyangr_D ' 'Bhandup West_Dc '
'ChotiHvl_DC ' 'Nehru3PL_D ' 'BaljiDPP_D ' 'ColageRD_D ' 'keshod_DC '
'GndhiNgr_IP ' 'Pshrikvu_D ' 'Kooriyad_D ' 'Ambedkar_D ' 'Truptingr_D '
'Udyabata_D ' 'TilakNgr_D ' 'MunplDPP_D ' 'Cherukole_D ' 'NorprRD_D '
'Palikval_D ' 'SadulDPP_D ' 'Manchar_D ' 'Artclgrd_D ' 'SubrtDPP_D '
'Old ' 'ShantiDPP_D ' 'TirupthiRd_D ' 'Royapuram ' 'NagplDPP_D '
'SliprDPP_DC ' 'BrezeDPP_D ' 'ShjnprRD_D ' 'Nishangr_D ' 'KrisnKunj_D '
'ShivaDPP_D ' 'BasthDPP_D ' 'BargaDPP_D ' 'KrantiNgr_D ' 'Palani_D '
'Yadvgiri_IP ' 'MotvdDPP_D ' 'Dakor_DC ' 'East_D_7 ' 'SingCLNY_D '
'Blmrgnst_D ' 'YuktiDPP_D ' 'AlathurRD_D ' 'DiyoDPP_D ' 'Bnsibtla_D '
'LohiaDPP_D ' 'Chaitnya_D ' 'Fathuluh_D ' 'RgstrOFC_D ' 'Sanpada_CP '
'Pakrela_D ' 'ShntiSgr_D ' 'Pnchlght_D ' 'Bhilai_DC ' 'Nattukal_D '
'ArickDPP_D ' 'Kaura_D ' 'NaginaRd_D ' 'VrdhriRD_D ']
---------------------------------------------------------------------------
-----------------
trip_creation_day
['Wed' 'Thu' 'Fri' 'Sat' 'Sun' 'Mon' 'Tue']
---------------------------------------------------------------------------
-----------------
trip_creation_month
['Sep' 'Oct']
---------------------------------------------------------------------------
-----------------
trip_creation_year
[2018]
---------------------------------------------------------------------------
-----------------
trip_creation_hour
[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23]
---------------------------------------------------------------------------
-----------------
```

```python
In [146…   df.describe().T
```

Out[1469]:

	count	mean	std	min	25%	50%
start_scan_to_end_scan	14787.0	529.429025	658.254936	23.000000	149.000000	279.000000
actual_distance_to_destination	14787.0	164.090196	305.502982	9.002461	22.777099	48.287894
actual_time	14787.0	356.306012	561.517936	9.000000	67.000000	148.000000
osrm_time	14787.0	160.990938	271.459495	6.000000	29.000000	60.000000
osrm_distance	14787.0	203.887411	370.565564	9.072900	30.756900	65.302800
segment_actual_time	14787.0	353.059174	556.365911	9.000000	66.000000	147.000000
segment_osrm_time	14787.0	180.511598	314.679279	6.000000	30.000000	65.000000
segment_osrm_distance	14787.0	222.705466	416.846279	9.072900	32.578850	69.784200
od_total_time	14787.0	530.313468	658.415416	23.460000	149.695000	279.710000

◄ ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬ ▶

In [147…
```python
df.describe(include='object').T
```

Out[1470]:

	count	unique	top	freq
trip_uuid	14787	14787	trip-153671041653548748	1
source_state	14787	29	Maharashtra	2714
source_city	14787	731	Gurgaon	1128
source_place	14787	756	Bilaspur_HB	1052
destination_state	14787	31	Maharashtra	2561
destination_city	14787	856	Bengaluru	1088
destination_place	14787	844	Bilaspur_HB	821
trip_creation_day	14787	7	Wed	2731
trip_creation_month	14787	2	Sep	13011
trip_creation_year	14787	1	2018	14787
trip_creation_hour	14787	24	22	1123

Outlier Identification

In [147…
```python
# Identiying the distribution of numerical variables before removing outliers
plt.figure(figsize = (12, 10))
for i in range(len(numerical_columns)):
    plt.subplot(3, 3, i + 1)
    sns.histplot(df[numerical_columns[i]], kde = True)
    plt.plot()

plt.suptitle("Distribution of time and distance with Outliers", size = 18, fontweig
plt.tight_layout()
plt.show()
```

## Distribution of time and distance with Outliers
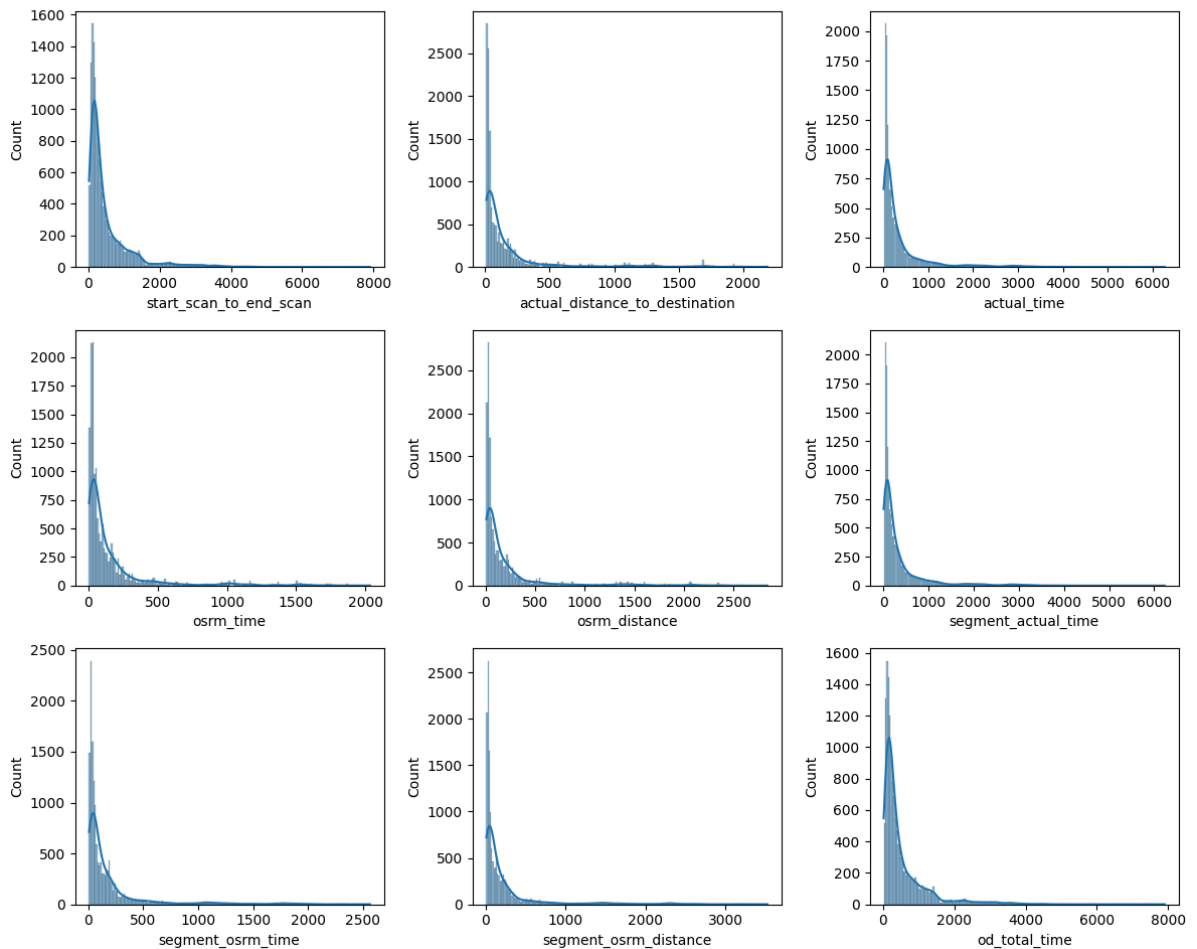
Observations:

- It can be inferred from the above plots that data in all the numerical columns are right skewed.

In [147…
```python
# Visual Analysis of Outliers
plt.figure(figsize = (12, 10))
for i in range(len(numerical_columns)):
    plt.subplot(3, 3, i + 1)
    sns.boxplot(df[numerical_columns[i]])
    plt.plot()

plt.suptitle("Outlier Detection", size = 18, fontweight = "medium")
plt.tight_layout()
plt.show()
```

Outlier Detection
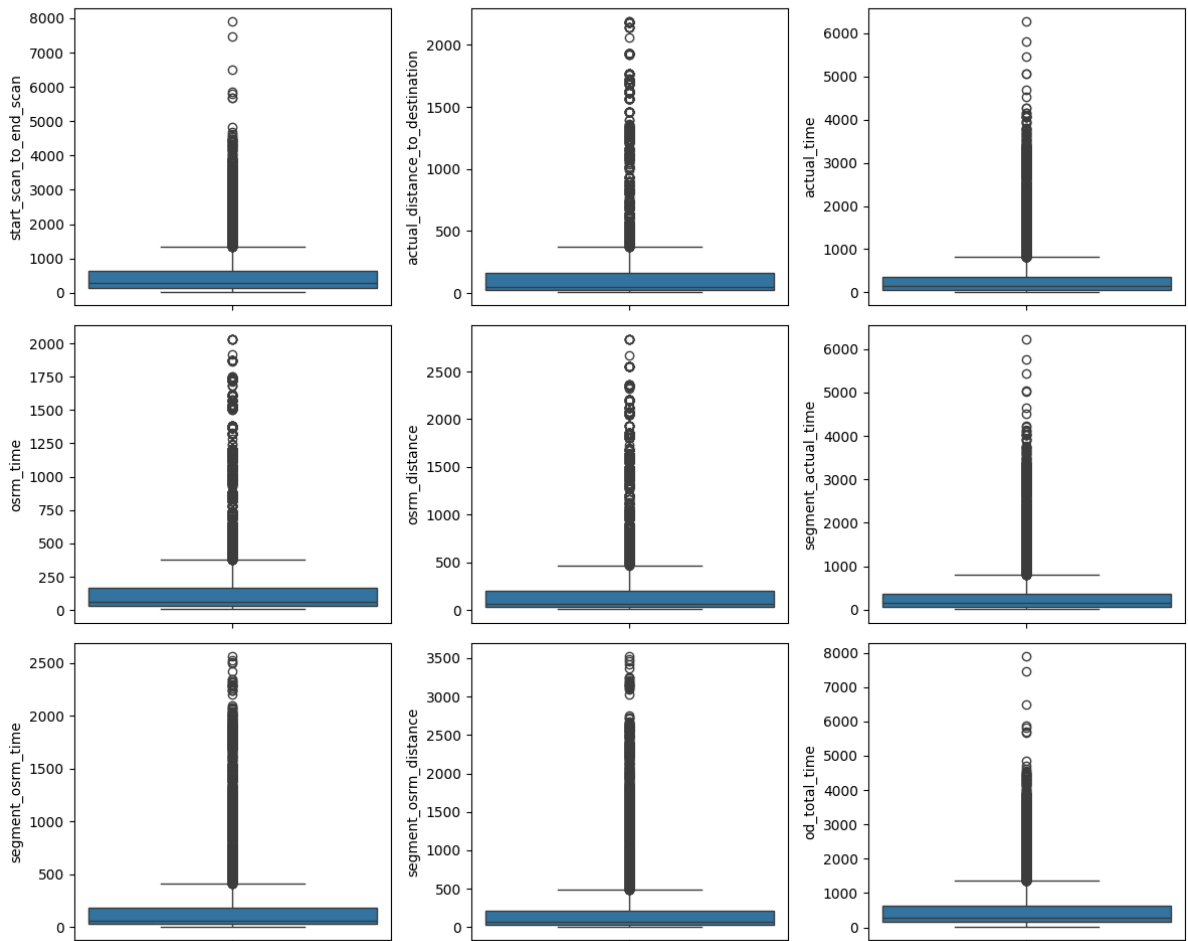
Observations:

- It can be clearly seen in the above plots that there are outliers in all the numerical columns that need to be treated.

Outlier Treatment

In [147…
```python
# Outlier Removal using IQR method
def remove_outliers(df, cols):
    for col in cols:
        Q1 = df[col].quantile(0.25)
        Q3 = df[col].quantile(0.75)
        IQR = Q3 - Q1
        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR
        df = df[(df[col] >= lower_bound) & (df[col] <= upper_bound)]
    return df
```

In [147…
```python
df = remove_outliers(df, numerical_columns)
df.reset_index(inplace=True)
df.drop(columns=['index'],inplace=True)
df.shape
```

Out[1474]: (8734, 22)

Graphical Analysis

In [147…
```python
# Distribution of numerical variables after removing outliers
plt.figure(figsize = (12, 10))
for i in range(len(numerical_columns)):
```

```
    plt.subplot(3, 3, i + 1)
    sns.histplot(df[numerical_columns[i]], kde = True)
    plt.plot()

plt.suptitle("Distribution of time and distance Excluding Outliers", size = 18, for
plt.tight_layout()
plt.show()
```

Distribution of time and distance Excluding Outliers
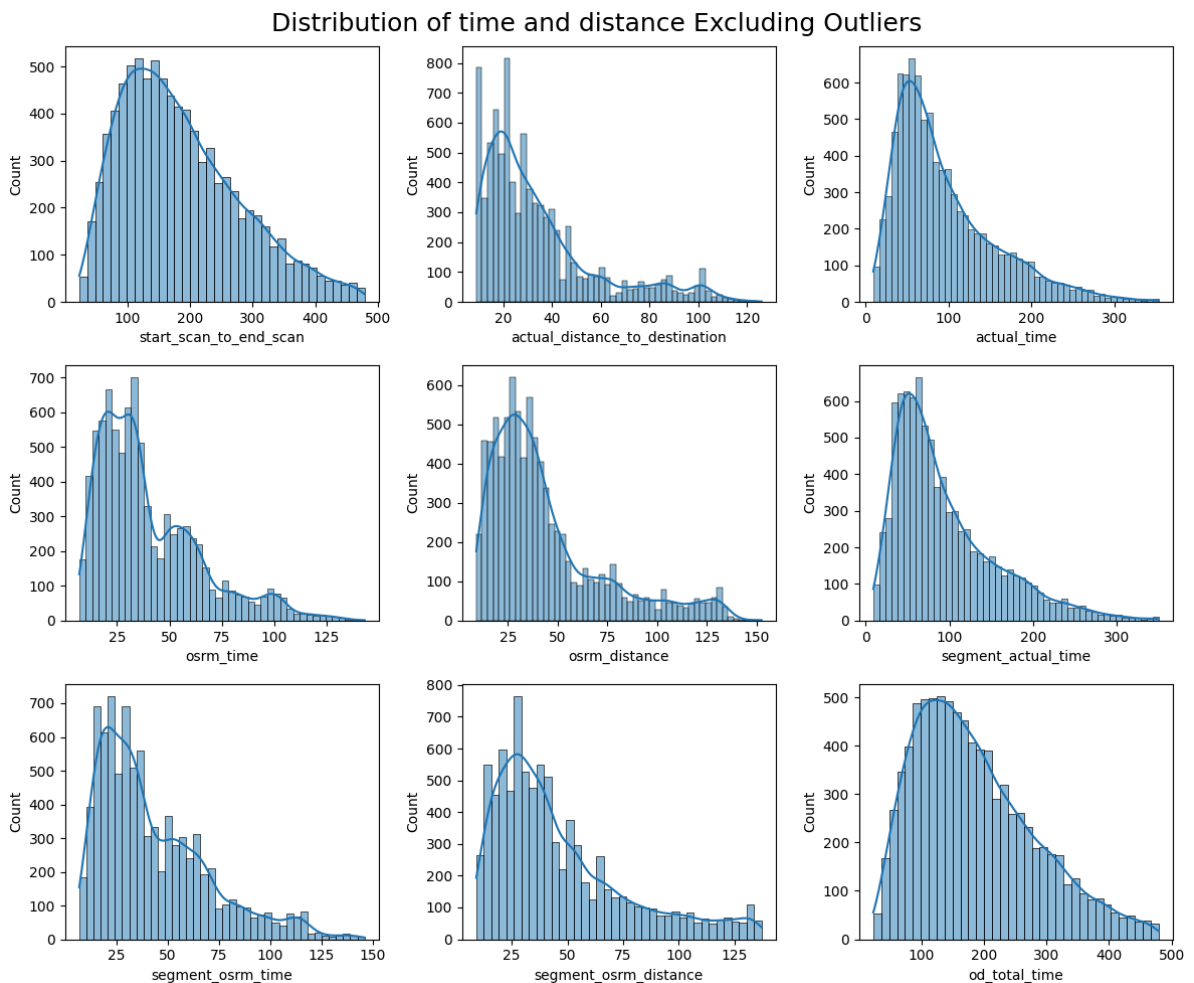
Observations

1. After handling the outliers, all the time and distance data is observed to follow normal distribution.
2. The mean time taken for delivery is significantly different between total time and actual time.
3. The mean time taken for delivery is significantly different between actual time and osrm time.
4. The mean time for osrm and segment_osrm times are observed to be similar.
5. The mean distance between actual distance and osrm distance is observed to be similar.

In [147…

```
sns.heatmap(df.corr(numeric_only=True, method='spearman'), annot = True)

plt.suptitle("Correlation Matrix (Heatmap)", size = 20, fontweight = "medium")
plt.tight_layout()
plt.show()
```

Correlation Matrix (Heatmap)
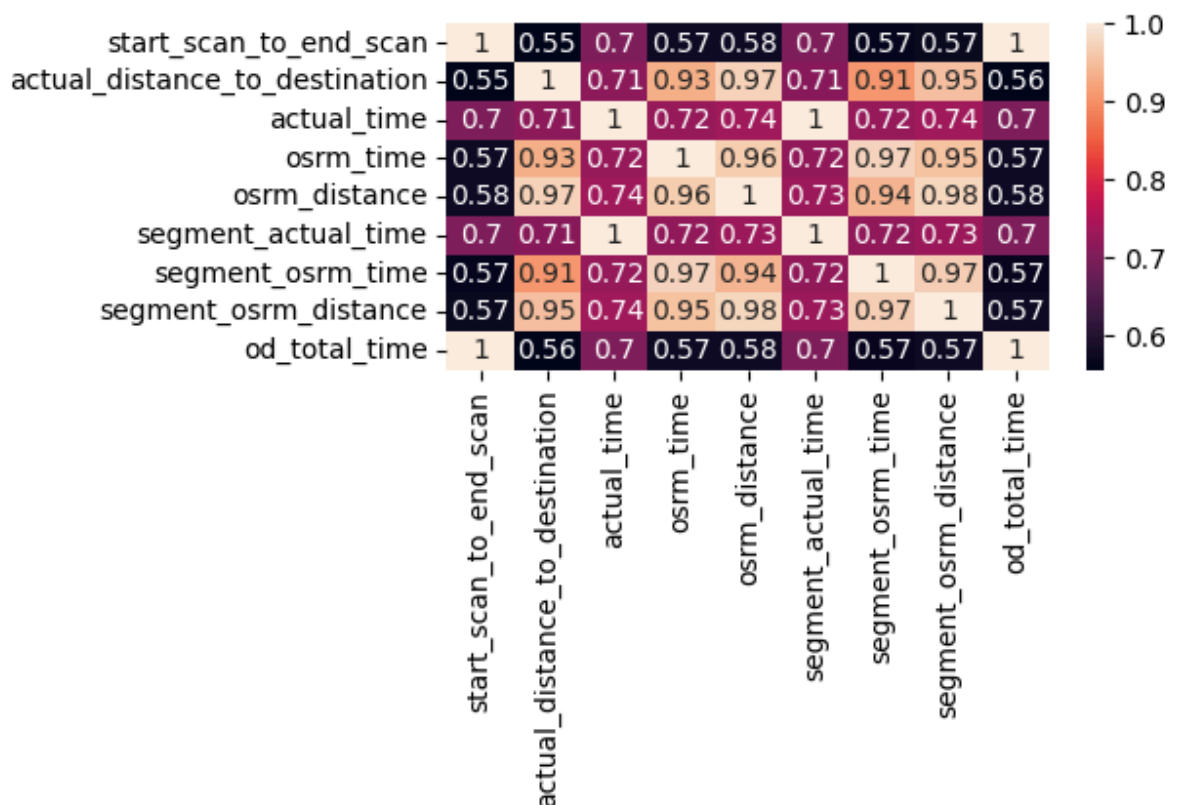
Observations

1. Correlation between actual distance and actual time is lesser than the correlation between osrm distance and osrm time.

2. The actual distance, osrm distance and segment osrm distance are positively highly correlated which explains the accuracy of osrm application.

3. The total time recorded as the start to end scan for completion of delivery has lesser positive correlation with Actual time which signifies there is some delay in updating the delivery/pick-up as complete immediately after completion (*total_time > actual_time*).

4. There is lesser positive correlation between actual time and osrm time which might signify other factors involved in causing the delay in delivery/pick-ups (*actual_time > osrm_time*).

In [147...]
```python
sns.countplot(data=df, x='source_state', hue='route_type')
plt.title('Top 10 Source Cities by Deliveries')
plt.ylabel('Count')
plt.xticks(fontsize=8, rotation = 45)
plt.tight_layout()
plt.show()
```

Top 10 Source Cities by Deliveries
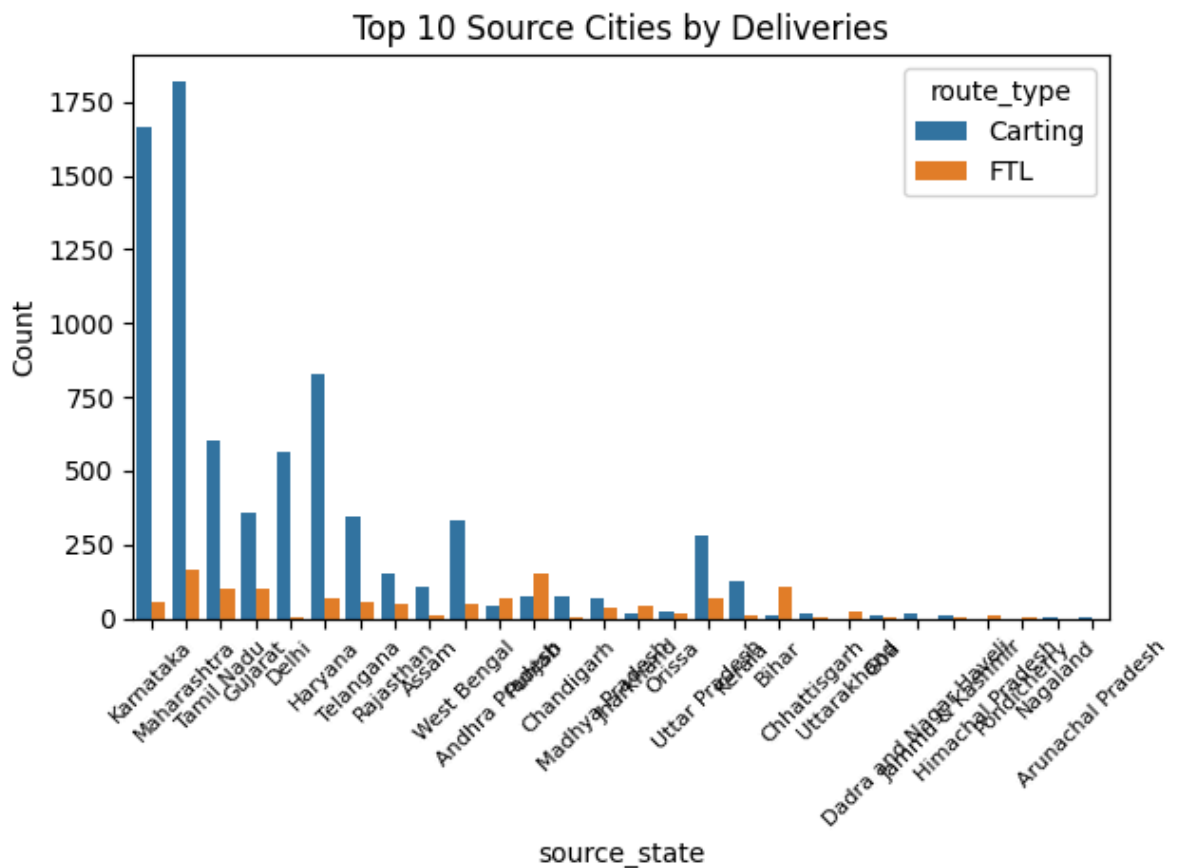

```
In [147…  state_route = df.groupby(['source_state','route_type'])['trip_uuid'].count().reset_
          state_route.head()
```

Out[1478]:

	source_state	route_type	trip_uuid
0	Andhra Pradesh	Carting	40
1	Andhra Pradesh	FTL	65
2	Arunachal Pradesh	Carting	1
3	Arunachal Pradesh	FTL	0
4	Assam	Carting	109

```
In [147…  plt.figure(figsize = (12, 6))
          for i in range(len(numerical_columns)):
              plt.subplot(3, 3, i + 1)
              sns.barplot(data=df, x = 'route_type', y = df[numerical_columns[i]])
              plt.plot()

          plt.suptitle("Time & Disctance analysis by Route type", size = 18, fontweight = "me
          plt.tight_layout()
          plt.show()
```

Time & Disctance analysis by Route type
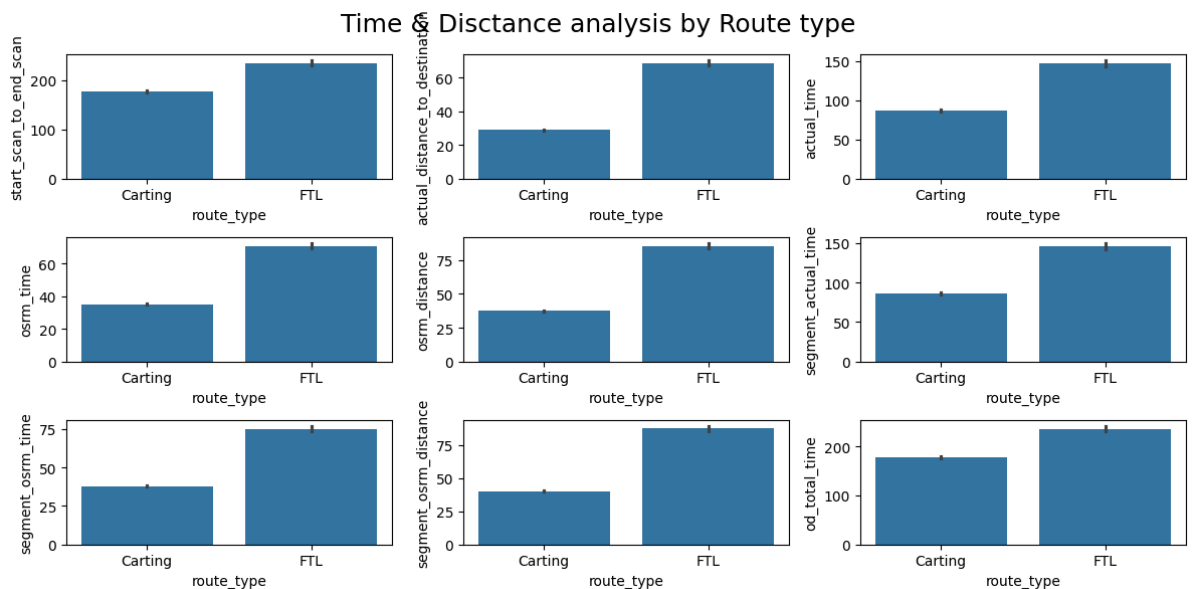
Observations

1. The route type - Carting takes more time ~125 minutes to deliver/pick-up compared to FTL which takes ~110 minutes.
2. The actual time taken to deliver via Carting (~ 65 mins) is greater than the time estimated by osrm system (~ 30 mins).
3. The total time logged for carting type (~ 125 mins) is much greater than the actual time (~ 65 mins).
4. The total time and actual time taken by FTL type (~ 110 mins) is also greater than estimated osrm time (~ 40 mins).
5. The actual distance logged is almost similar to the estimated osrm distance for Carting (~ 25 / 32 mins) and FTL types (~ 38 / 43 mins).

In [148…
```python
# City and State Delivery counts by Route Type
state_route = df.groupby(['source_state','route_type'])['trip_uuid'].count().reset_
state_route = state_route.sort_values(by='trip_uuid', ascending=False)

city_route = df.groupby(['source_city','route_type'])['trip_uuid'].count().reset_ir
city_route = city_route.sort_values(by='trip_uuid', ascending=False)
```

In [148…
```python
aero_blue = "#243e8d"
aero_grey = "#808080"
aero_red = "#db2926"
colors = [aero_red, aero_blue]

plt.figure(figsize = (12,8))
plt.subplot(2,3,1)
df['source_city'].value_counts().head(10).plot(kind='bar')
plt.title('Top 10 Cities by Deliveries')
plt.ylabel('Count')
plt.xticks(fontsize=8, rotation = 45)

plt.subplot(2,3,2)
df['source_state'].value_counts().head(10).plot(kind='bar')
plt.title('Top 10 States by Deliveries')
plt.ylabel('Count')
plt.xticks(fontsize=8, rotation = 45)

plt.subplot(2,3,3)
df['route_type'].value_counts().plot(kind='bar')
```

```python
plt.title('Delivery count by Route type')
plt.ylabel('Count')
plt.xticks(fontsize=8, rotation = 45)

plt.subplot(2,3,4)
sns.barplot(data=city_route.head(10), x='source_city', y='trip_uuid', hue='route_ty
plt.ylabel('Count')
plt.xticks(fontsize=8, rotation = 45)

plt.subplot(2,3,5)
sns.barplot(data=state_route.head(10), x='source_state', y='trip_uuid', hue='route_
plt.xticks(fontsize=8, rotation = 45)

plt.suptitle("Delivery Trend by City & State", size = 18, fontweight = "medium")
plt.show()
```

Delivery Trend by City & State
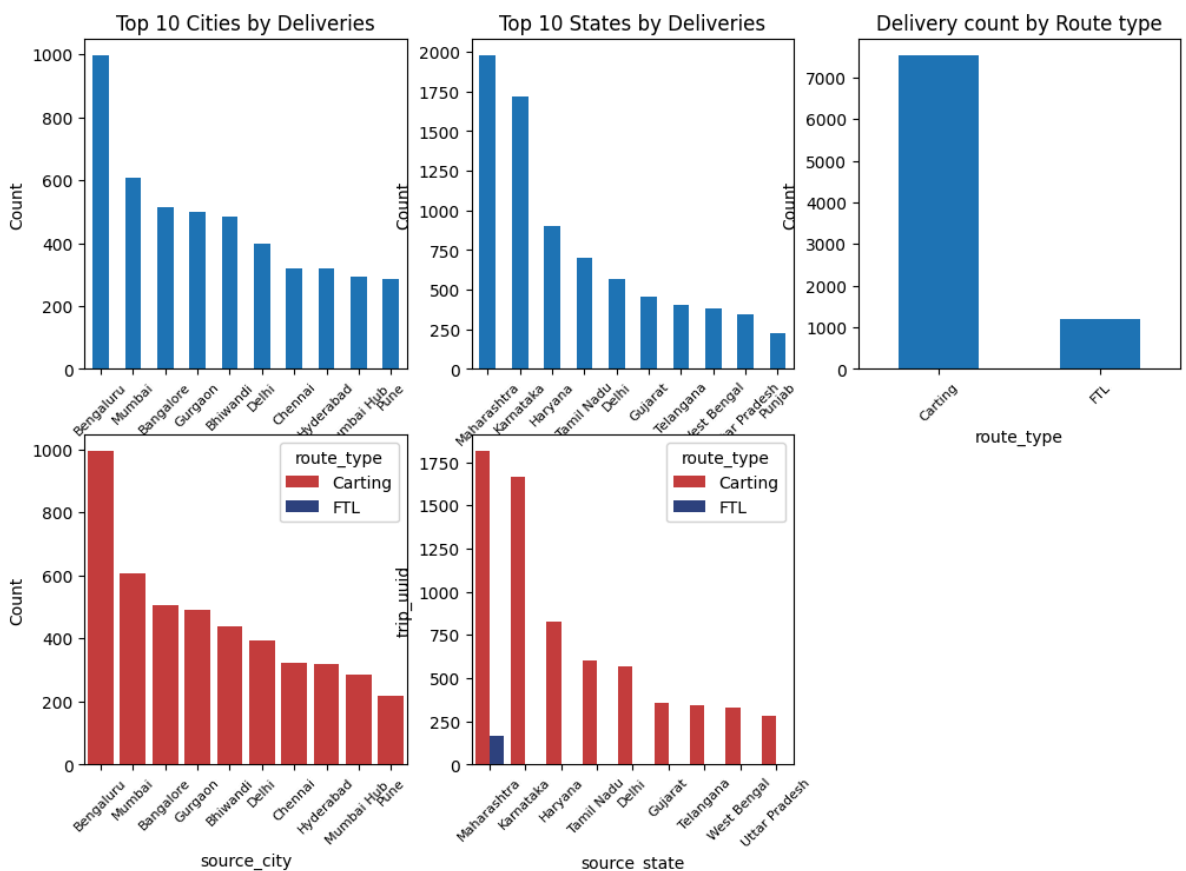
Observations

1. Top cities by count of trips include Bengaluru, Mumbai, Delhi, Bhiwandi, Chennai, Hyderabad, etc.
2. Top states by count of trips include Karnataka, Maharashtra, Tamilnadu, etc.
3. Carting type contributes to most number of deliveries.
4. Maharashtra, Karnataka and Tamilnadu has highest carting type deliveries and Karnataka, Andhra Pradesh, Bihar and Tamilnadu contribute to highest number of FTL type deliveries.

```python
In [148…   df_day=df.groupby(by="trip_creation_day")["trip_uuid"].count().to_frame().reset_ind
           df_day.head()
```

Out[1482]:

	trip_creation_day	trip_uuid
0	Fri	1223
1	Mon	1166
2	Sat	1322
3	Sun	982
4	Thu	1244

In [148…

```python
aero_blue = "#243e8d"
aero_grey = "#808080"
aero_red = "#db2926"
colors = [aero_red, aero_blue, aero_grey]

plt.suptitle("Day wise Trends", size = 18, fontweight = "medium")

plt.figure(figsize = (9,4))
plt.subplot(1,1,1)
sns.countplot(data=df, x='trip_creation_day', hue='route_type', palette=colors)
plt.xticks(fontsize=8)

plt.figure(figsize = (12,3))
plt.subplot(1,2,1)
sns.barplot(data=df, x='trip_creation_day', y='actual_time')
plt.xticks(fontsize=8)

plt.subplot(1,2,2)
sns.barplot(data=df, x='trip_creation_day', y='actual_distance_to_destination')
plt.xticks(fontsize=8)

plt.show()
```
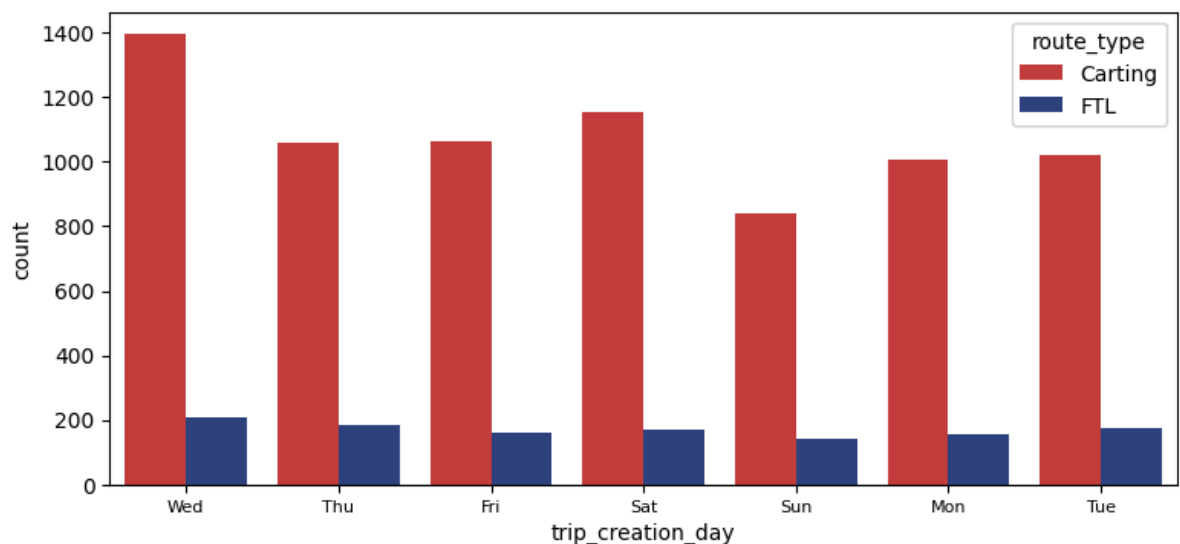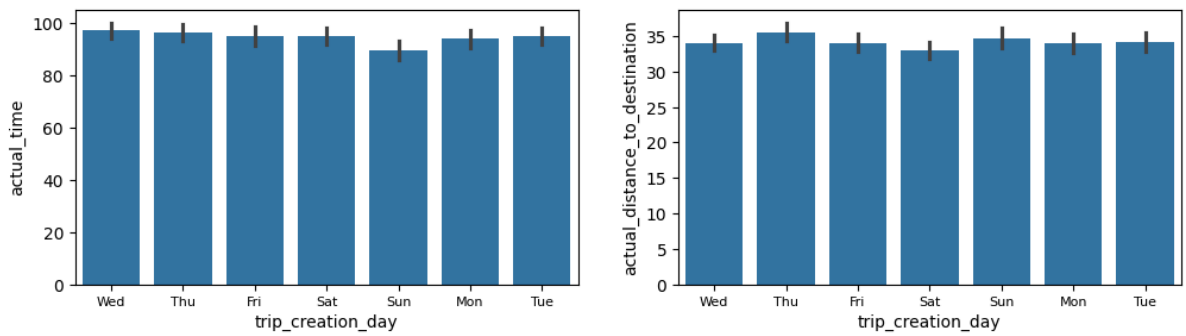
```
<Figure size 640x480 with 0 Axes>
```

Observations

1. Most number of deliveries are dispatched during mid of the week which is between Tuesday and Friday for both Carting and FTL types.

```
In [148…  df_hour=df.groupby(by="trip_creation_hour")["trip_uuid"].count().to_frame().reset_i
          df_hour.head()
```

Out[1484]:

	trip_creation_hour	trip_uuid
0	0	651
1	1	386
2	2	427
3	3	463
4	4	443

```
In [148…  plt.figure(figsize=(10,5))
          sns.lineplot(data=df_hour,x=df_hour["trip_creation_hour"], y=df_hour["trip_uuid"] )
          plt.xticks(np.arange(0,24))
          plt.grid("both")
          plt.tight_layout()
          plt.show()
```
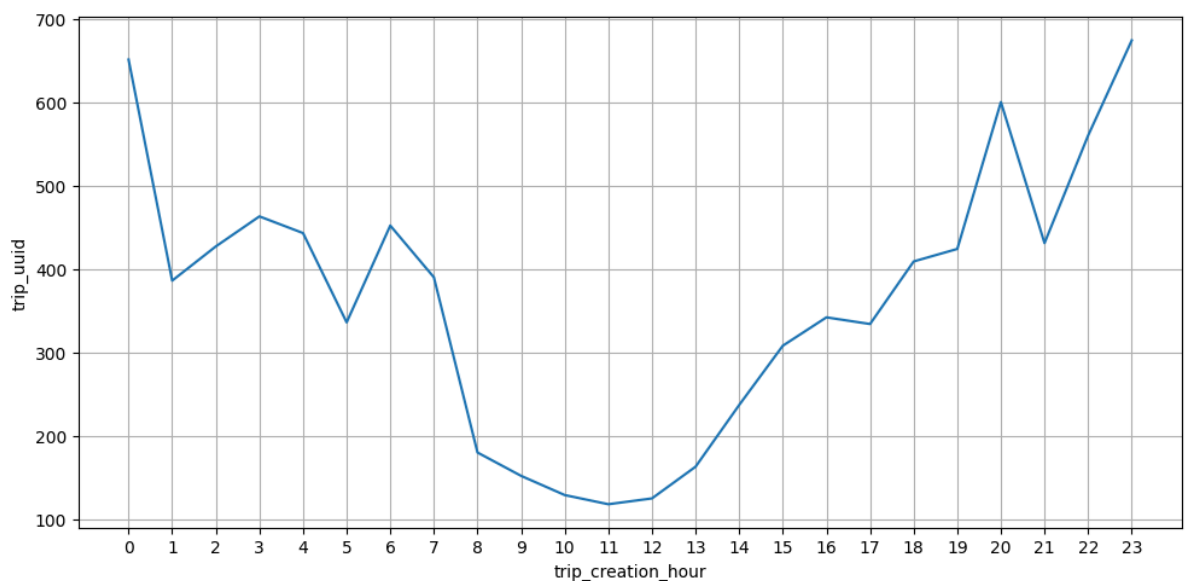
Observation:

- It can be inferred from the above plot, that the number of trips started increasing after the noon, becomes maximum at 10 PM and then starts decreasing.

Hypothesis testing

1. Compare the difference between Point a. and start_scan_to_end_scan. Do hypothesis testing/ Visual analysis to check.

- Null Hypothesis (H0):od_total_time (Total Trip Time) and start_scan_to_end_scan (Expected total trip time) are same.

- Alternate Hypothesis (HA): od_total_time (Total Trip Time) and start_scan_to_end_scan (Expected total trip time) are different.

2.Checking for basic assumpitons for the hypothesis:

- Distribution check using QQ Plot

- Homogeneity of Variances using Lavene's test

1. Define Test statistics; Distribution of T under H0.--If the assumptions of T Test are met then we can proceed performing T Test for independent samples else we will perform the nonparametric test equivalent to T Test for independent sample i.e., Mann-Whitney U rank test for two independent samples.

1. Compute the p-value and set alpha value be to 0.05
2. If p value< alpha: Reject Ho, p value> alpha: Accept Ho

- **QQ plot: To check normality**

```
In [148…  plt.figure(figsize = (15, 6))
          plt.subplot(1, 2, 1)
          plt.suptitle('QQ plots for od_total_time and start_scan_to_end_scan')
          spy.probplot(df['od_total_time'], plot = plt, dist = 'norm')
          plt.title('QQ plot for od_total_time')
          plt.subplot(1, 2, 2)
          spy.probplot(df['start_scan_to_end_scan'], plot = plt, dist = 'norm')
          plt.title('QQ plot for start_scan_to_end_scan')
          plt.plot()
```

Out[1486]:   []

QQ plots for od_total_time and start_scan_to_end_scan
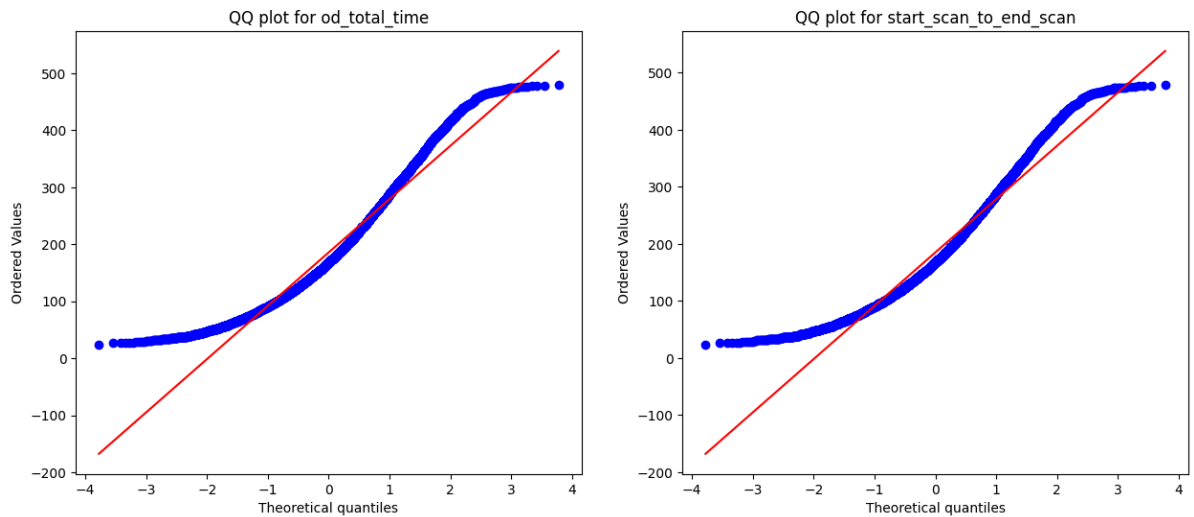
Observations: We observe that total time and start scan to end scan are not normally distributed. Hence, we perform Shapiro-Wilk test of normality

- Ho: Sample follows normal distribution
- Ha: Sample does not follow normal distribution

In [148...
```python
test_stat, p_value = spy.shapiro(df['od_total_time'].sample(5000))
print('p-value', p_value)
if p_value < 0.05:
    print('The sample does not follow normal distribution')
else:
    print('The sample follows normal distribution')
```

```
p-value 2.3461751048382927e-37
The sample does not follow normal distribution
```

In [148...
```python
test_stat, p_value = spy.shapiro(df['start_scan_to_end_scan'].sample(5000))
print('p-value', p_value)
if p_value < 0.05:
    print('The sample does not follow normal distribution')
else:
    print('The sample follows normal distribution')
```

```
p-value 2.0365852406720264e-37
The sample does not follow normal distribution
```

Homogeneity of Variances using Levene's test

- Ho: Sample has homogenous varriance
- Ha: Sample does not have homogenous varriance

In [148...
```python
test_stat, p_value = spy.levene(df['od_total_time'], df['start_scan_to_end_scan'])
print('p-value', p_value)
if p_value < 0.05:
    print('The samples do not have  Homogenous Variance')
else:
    print('The samples have Homogenous Variance ')
```

```
p-value 0.9319576263983587
The samples have Homogenous Variance
```

Observations:

- Since the samples are not normally distributed, T-Test cannot be applied here, we can perform its non parametric equivalent test i.e., Mann-Whitney U rank test for two independent samples.

In [149…
```python
test_stat, p_value = spy.mannwhitneyu(df['od_total_time'], df['start_scan_to_end_sc
print('P-value :',p_value)
```

P-value : 0.6568186023889078

Observations:

- Since p-value > alpha therfore it can be concluded that the mean of od_total_time and start_scan_to_end_scan are similar.

1. **Do hypothesis testing/ visual analysis between actual_time aggregated value and segment actual time aggregated value (aggregated values are the values you'll get after merging the rows on the basis of trip_uuid)**

- **QQ plot for normality distribution:**

In [149…
```python
plt.figure(figsize = (15, 6))
plt.subplot(1, 2, 1)
plt.suptitle('QQ plots for actual_time and osrm_time')
spy.probplot(df['actual_time'], plot = plt, dist = 'norm')
plt.title('QQ plot for actual_time')
plt.subplot(1, 2, 2)
spy.probplot(df['segment_actual_time'], plot = plt, dist = 'norm')
plt.title('QQ plot for segment_actual_time')
plt.plot()
```

Out[1491]: []
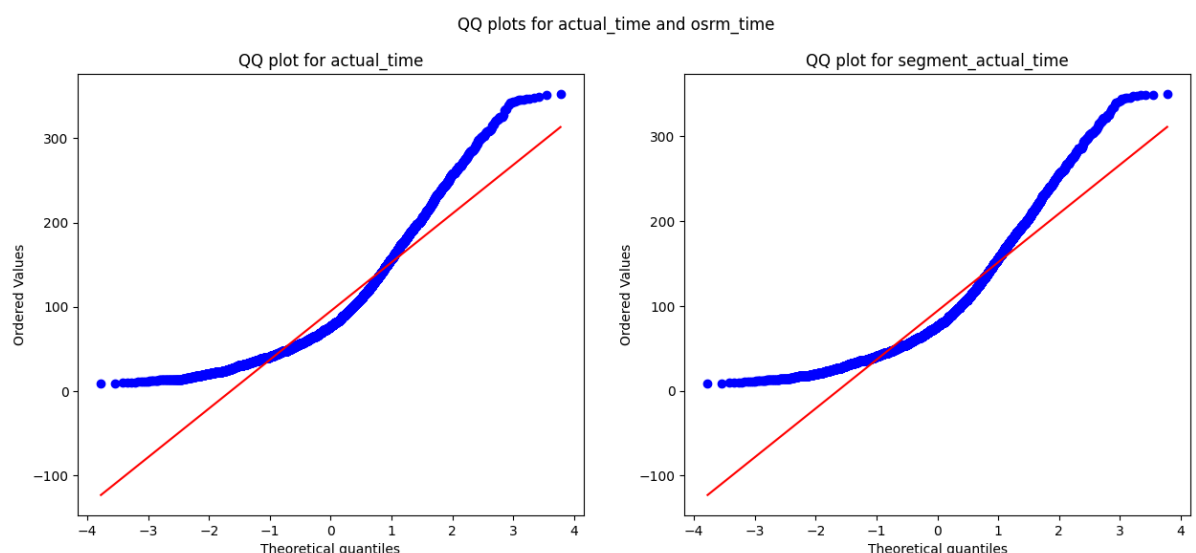
- **Shapiro-Wilk test of Normality**

In [149…
```python
test_stat, p_value = spy.shapiro(df['actual_time'].sample(5000))
print('p-value', p_value)
if p_value < 0.05:
    print('The sample does not follow normal distribution')
```

```
    else:
        print('The sample follows normal distribution')
```

```
p-value 7.992130456425146e-50
The sample does not follow normal distribution
```

In [149...
```python
test_stat, p_value = spy.shapiro(df['segment_actual_time'].sample(5000))
print('p-value', p_value)
if p_value < 0.05:
    print('The sample does not follow normal distribution')
else:
    print('The sample follows normal distribution')
```

```
p-value 5.687945697802934e-50
The sample does not follow normal distribution
```

Leven's test for varraince

In [149...
```python
test_stat, p_value = spy.levene(df['actual_time'], df["segment_actual_time"])
print('p-value', p_value)
if p_value < 0.05:
    print('The samples do not have  Homogenous Variance')
else:
    print('The samples have Homogenous Variance ')
```

```
p-value 0.5899257563473177
The samples have Homogenous Variance
```

In [149...
```python
test_stat, p_value = spy.mannwhitneyu(df['actual_time'], df['segment_actual_time'])
print('p-value', p_value)
if p_value < 0.05:
    print('The samples are not similar')
else:
    print('The samples are similar ')
```

```
p-value 0.23652064800465944
The samples are similar
```

Observations:

- We conclude that,Since p-value > alpha therfore it can be concluded that the mean of actual time and segment_actual_time are similar.

1. **Do hypothesis testing/ visual analysis between osrm distance aggregated value and segment osrm distance aggregated value (aggregated values are the values you'll get after merging the rows on the basis of trip_uuid)**

QQ plot for Normality

In [149...
```python
plt.figure(figsize = (15, 6))
plt.subplot(1, 2, 1)
plt.suptitle('QQ plots for osrm_distance and segment_osrm_distance')
spy.probplot(df['osrm_distance'], plot = plt, dist = 'norm')
plt.title('QQ plot for osrm_distance')
plt.subplot(1, 2, 2)
spy.probplot(df['segment_osrm_distance'], plot = plt, dist = 'norm')
plt.title('QQ plot for segment_osrm_distance')
plt.plot()
```

Out[1496]:   []

QQ plots for osrm_distance and segment_osrm_distance
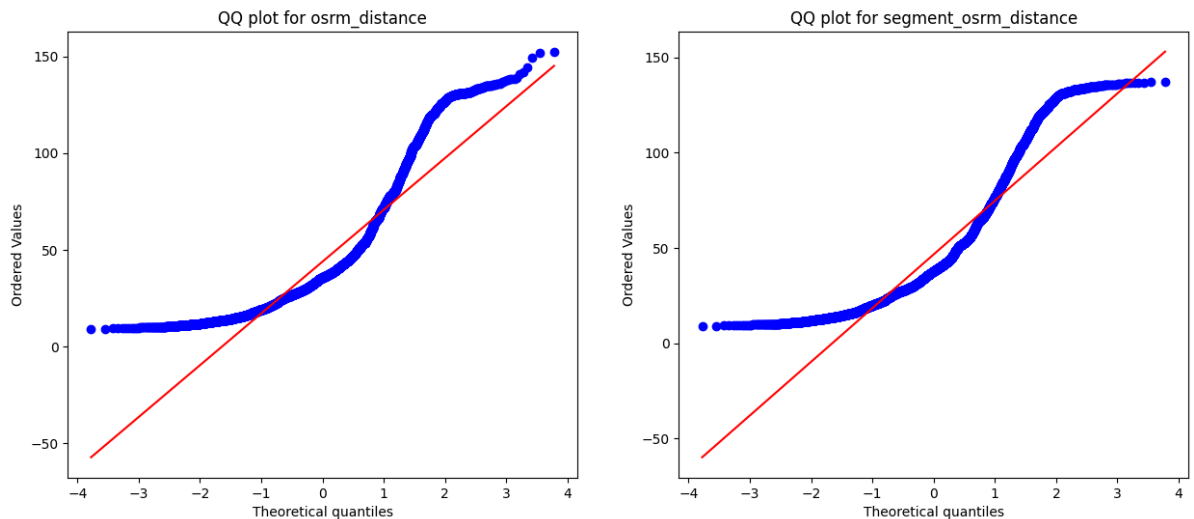
Shapiro-Wilk test

```
In [149…   test_stat, p_value = spy.shapiro(df['osrm_distance'].sample(5000))
           print('p-value', p_value)
           if p_value < 0.05:
               print('The sample does not follow normal distribution')
           else:
               print('The sample follows normal distribution')
```

```
p-value 4.478052377553121e-56
The sample does not follow normal distribution
```

```
In [149…   test_stat, p_value = spy.shapiro(df['segment_osrm_distance'].sample(5000))
           print('p-value', p_value)
           if p_value < 0.05:
               print('The sample does not follow normal distribution')
           else:
               print('The sample follows normal distribution')
```

```
p-value 2.7620639613233005e-52
The sample does not follow normal distribution
```

Levene's test for varriance

```
In [149…   test_stat, p_value = spy.levene(df['osrm_distance'], df['segment_osrm_distance'])
           print('p-value', p_value)

           if p_value < 0.05:
               print('The samples do not have Homogenous Variance')
           else:
               print('The samples have Homogenous Variance ')
```

```
p-value 4.0301907511660975e-07
The samples do not have Homogenous Variance
```

Observations:

- Since the samples do not follow any of the assumptions, T-Test cannot be applied here. We can perform its non parametric equivalent test i.e., Mann-Whitney U rank test for two independent samples.

```
In [150…   test_stat, p_value = spy.mannwhitneyu(df['osrm_distance'], df['segment_osrm_distanc
           print('p-value', p_value)
           if p_value < 0.05:
```

```
        print('The samples are not similar')
else:
        print('The samples are similar ')
```

```
p-value 4.738210070601972e-11
The samples are not similar
```

Observations:

- Since p-value < alpha therfore it can be concluded that osrm_distance and segment_osrm_distance are not similar.

1. **Do hypothesis testing/ visual analysis between osrm time aggregated value and segment osrm time aggregated value (aggregated values are the values you'll get after merging the rows on the basis of trip_uuid)**

- **QQ plot for normality**

In [150…

```
plt.figure(figsize = (15, 6))
plt.subplot(1, 2, 1)
plt.suptitle('QQ plots for osrm_time and segment_osrm_time')
spy.probplot(df['osrm_time'], plot = plt, dist = 'norm')
plt.title('QQ plot for osrm_time')
plt.subplot(1, 2, 2)
spy.probplot(df['segment_osrm_time'], plot = plt, dist = 'norm')
plt.title('QQ plot for segment_osrm_time')
plt.plot()
```

Out[1501]:   []
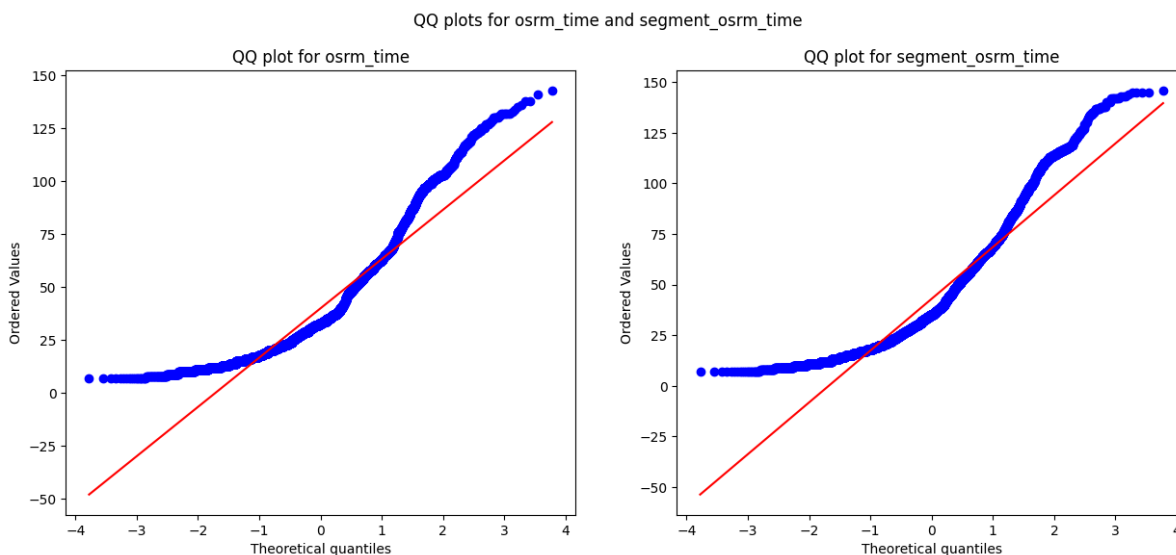
- **Shapiro-Wilk test for normality**

In [150…

```
test_stat, p_value = spy.shapiro(df['osrm_time'].sample(5000))
print('p-value', p_value)
if p_value < 0.05:
        print('The sample does not follow normal distribution')
else:
        print('The sample follows normal distribution')
```

```
p-value 2.6814737350645426e-50
The sample does not follow normal distribution
```

```python
test_stat, p_value = spy.shapiro(df['segment_osrm_time'].sample(5000))
print('p-value', p_value)
if p_value < 0.05:
    print('The sample does not follow normal distribution')
else:
    print('The sample follows normal distribution')
```

```
p-value 2.2487760644677598e-48
The sample does not follow normal distribution
```

- **Levene's test**

```python
test_stat, p_value = spy.levene(df['osrm_time'], df['segment_osrm_time'])
print('p-value', p_value)

if p_value < 0.05:
    print('The samples do not have Homogenous Variance')
else:
    print('The samples have Homogenous Variance ')
```

```
p-value 1.7614929417581122e-15
The samples do not have Homogenous Variance
```

Observations:

- Since the samples do not follow any of the assumptions, T-Test cannot be applied here. We can perform its non parametric equivalent test i.e., Mann-Whitney U rank test for two independent samples.

```python
test_stat, p_value = spy.mannwhitneyu(df['osrm_time'], df['segment_osrm_time'])
print('p-value', p_value)
if p_value < 0.05:
    print('The samples are not similar')
else:
    print('The samples are similar ')
```

```
p-value 7.564533251582031e-10
The samples are not similar
```

Observations:

- Since p-value < alpha therfore it can be concluded that osrm_time and segment_osrm_time are not similar.

One-Hot Encoding of Categorical Variables

```python
df['route_type'].value_counts()
```

Out[1506]:

route_type	count
Carting	7542
FTL	1192

dtype: int64

- **Performing one-hot label encoding on route type column**

In [150...
```python
label_encoder= LabelEncoder()
df["route_type"]=label_encoder.fit_transform(df["route_type"])
```

In [150...
```python
df['route_type'].value_counts() #after one-hot encoding
```

Out[1508]:

	count
route_type	
0	7542
1	1192

dtype: int64

- **Performing one-hot label encoding on data column**

In [150...
```python
df['data'].value_counts()
```

Out[1509]:

	count
data	
training	6195
test	2539

dtype: int64

In [151...
```python
label_encoder = LabelEncoder()
df['data'] = label_encoder.fit_transform(df['data'])
```

In [151...
```python
df['data'].value_counts()
```

Out[1511]:

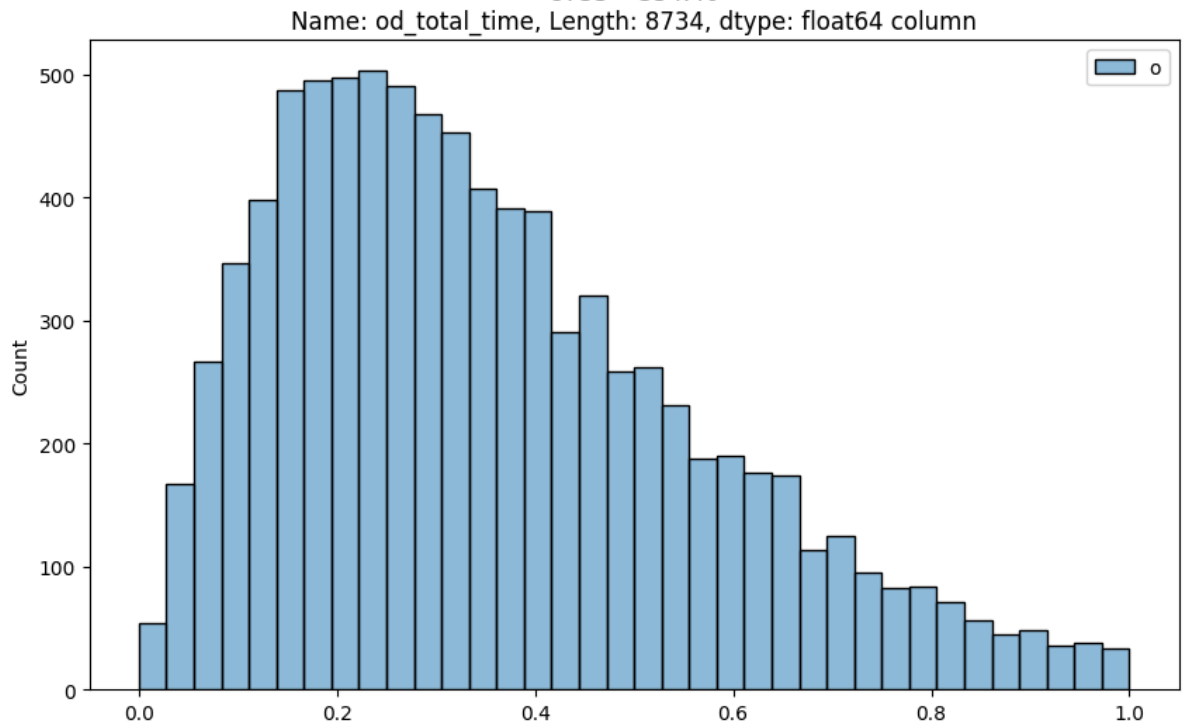|	count
data	
1	6195
0	2539

dtype: int64

Normalize/ Standardize the numerical features using MinMaxScaler or StandardScaler.

In [151...
```python
plt.figure(figsize = (10, 6))
scaler = MinMaxScaler()
scaled = scaler.fit_transform(df['od_total_time'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Normalized {df['od_total_time']} column")
plt.legend('od_total_time')
plt.plot()
```

Out[1512]:   []

```
                      Normalized 0      181.61
                                1      100.49
                                2      190.49
                                3       98.01
                                4      146.84
                                      ...
                             8729      88.22
                             8730     258.03
                             8731      60.59
                             8732     422.12
                             8733     354.40
              Name: od_total_time, Length: 8734, dtype: float64 column
```
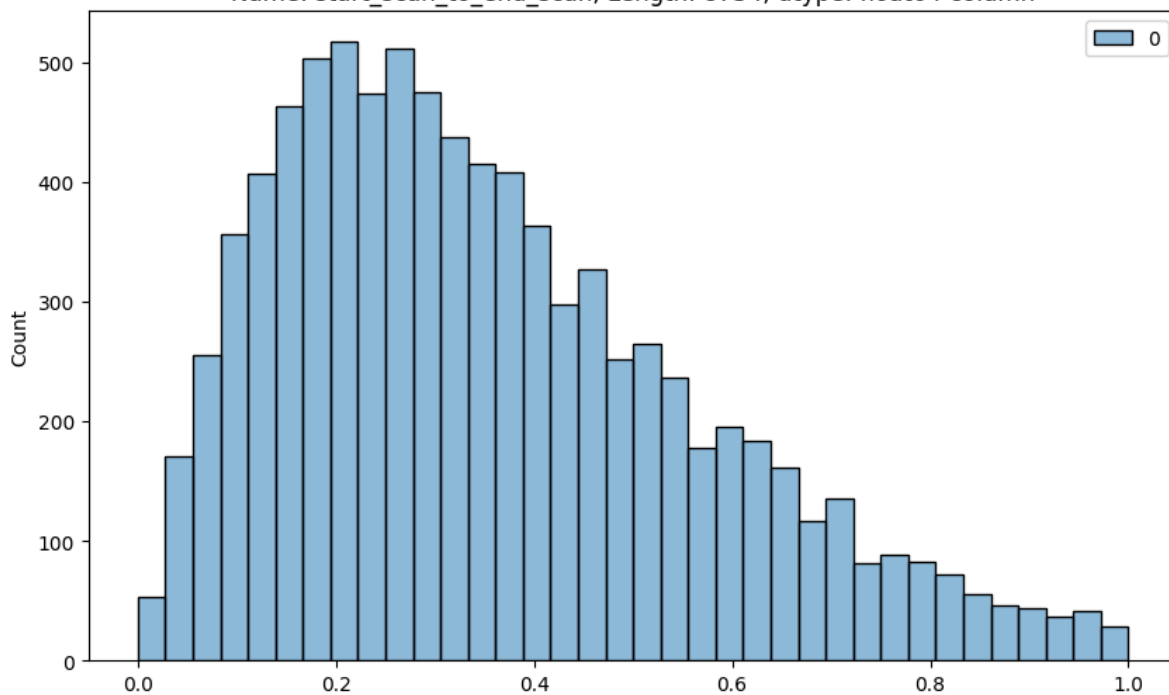


```python
plt.figure(figsize = (10, 6))
scaler = MinMaxScaler()
scaled = scaler.fit_transform(df['start_scan_to_end_scan'].to_numpy().reshape(-1, 1
sns.histplot(scaled)
plt.title(f"Normalized {df['start_scan_to_end_scan']} column")
plt.plot()
```

In [151…

Out[1513]:   []

```
Normalized 0       180.0
           1       100.0
           2       189.0
           3        98.0
           4       146.0
                    ...
        8729        88.0
        8730       257.0
        8731        60.0
        8732       421.0
        8733       353.0
Name: start_scan_to_end_scan, Length: 8734, dtype: float64 column
```


In [151…
```python
plt.figure(figsize = (10, 6))
scaler = MinMaxScaler()
scaled = scaler.fit_transform(df['actual_distance_to_destination'].to_numpy().resha
sns.histplot(scaled)
plt.title(f"Normalized {df['actual_distance_to_destination']} column")
plt.plot()
```

Out[1514]: []

```
Normalized 0      73.186911
           1      17.175274
           2      24.597048
           3       9.100510
           4      22.424210
                    ...
        8729      17.760248
        8730      57.762332
        8731      15.513784
        8732      38.684839
        8733      66.081533
Name: actual_distance_to_destination, Length: 8734, dtype: float64 column
```
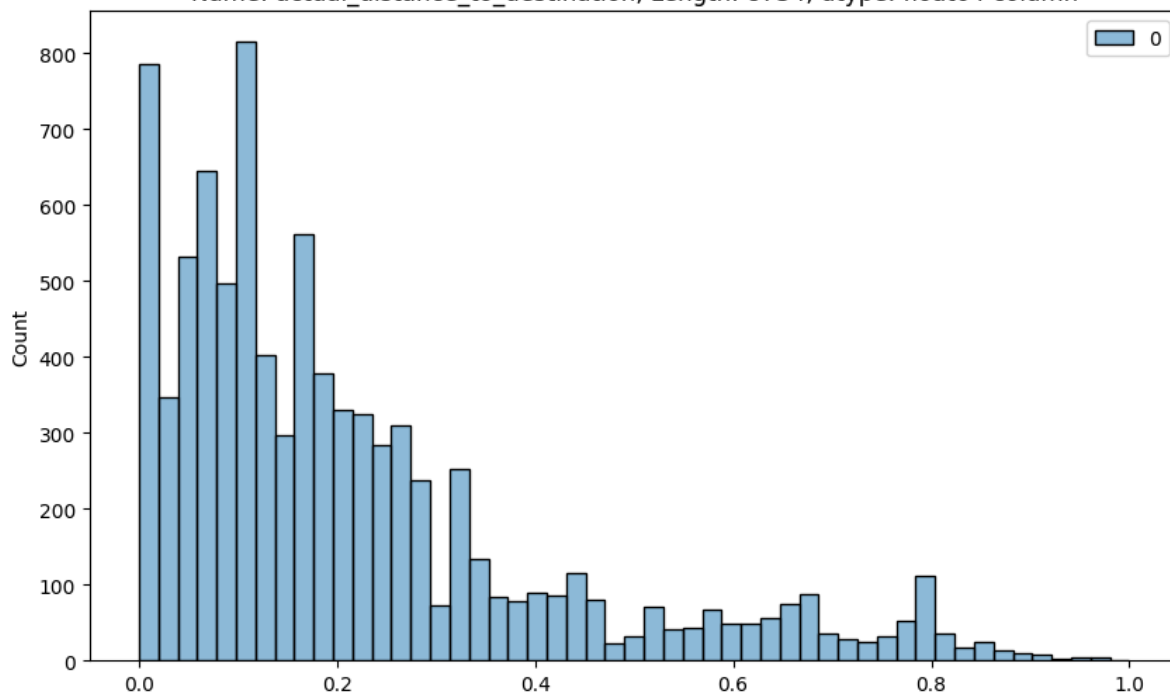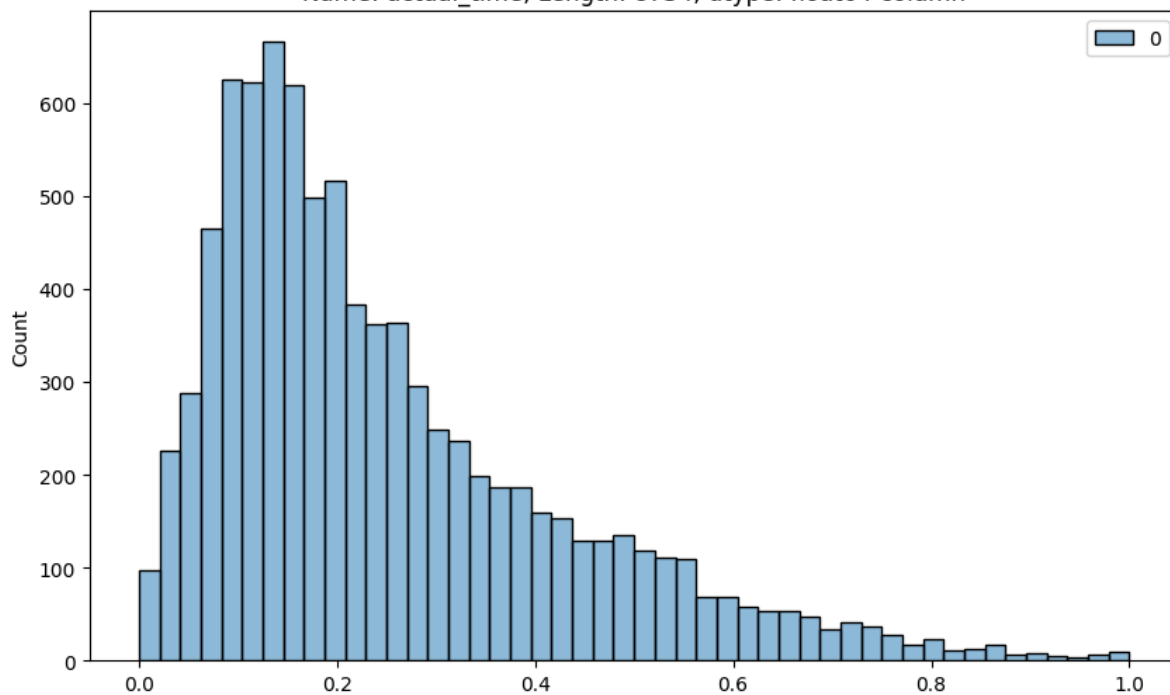


```python
plt.figure(figsize = (10, 6))
scaler = MinMaxScaler()
scaled = scaler.fit_transform(df['actual_time'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Normalized {df['actual_time']} column")
plt.plot()
```

Out[1515]: []

```
Normalized 0      143.0
           1       59.0
           2       61.0
           3       24.0
           4       64.0
              ...
        8729       38.0
        8730       83.0
        8731       21.0
        8732      282.0
        8733      275.0
Name: actual_time, Length: 8734, dtype: float64 column
```
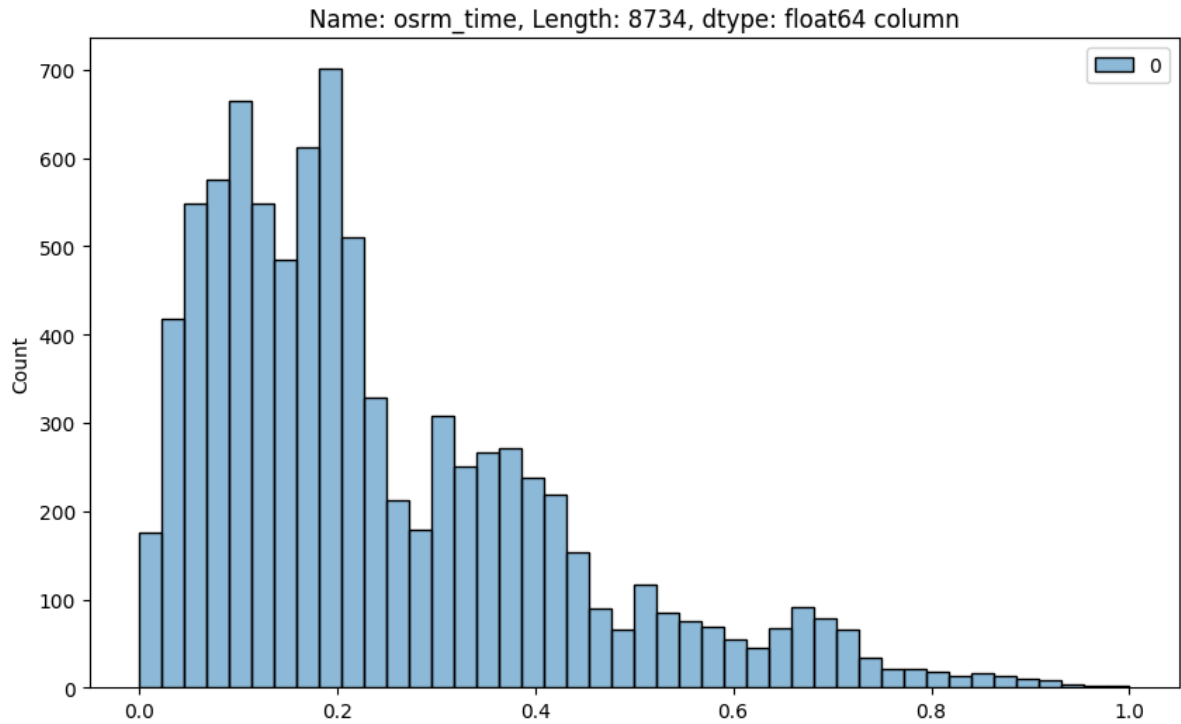


```
In [151…  plt.figure(figsize = (10, 6))
          scaler = MinMaxScaler()
          scaled = scaler.fit_transform(df['osrm_time'].to_numpy().reshape(-1, 1))
          sns.histplot(scaled)
          plt.title(f"Normalized {df['osrm_time']} column")
          plt.plot()
```

Out[1516]: []

```
Normalized 0      68.0
          1      15.0
          2      23.0
          3      13.0
          4      34.0
              ...
       8729     16.0
       8730     62.0
       8731     12.0
       8732     48.0
       8733     68.0
Name: osrm_time, Length: 8734, dtype: float64 column
```



```
In [151…   plt.figure(figsize = (10, 6))
           scaler = MinMaxScaler()
           scaled = scaler.fit_transform(df['osrm_distance'].to_numpy().reshape(-1, 1))
           sns.histplot(scaled)
           plt.title(f"Normalized {df['osrm_distance']} column")
           plt.plot()
```

Out[1517]:   []

```
Normalized 0      85.1110
           1      19.6800
           2      28.0647
           3      12.0184
           4      28.9203
                   ...
        8729       20.5065
        8730       73.4630
        8731       16.0882
        8732       58.9037
        8733       80.5787
Name: osrm_distance, Length: 8734, dtype: float64 column
```
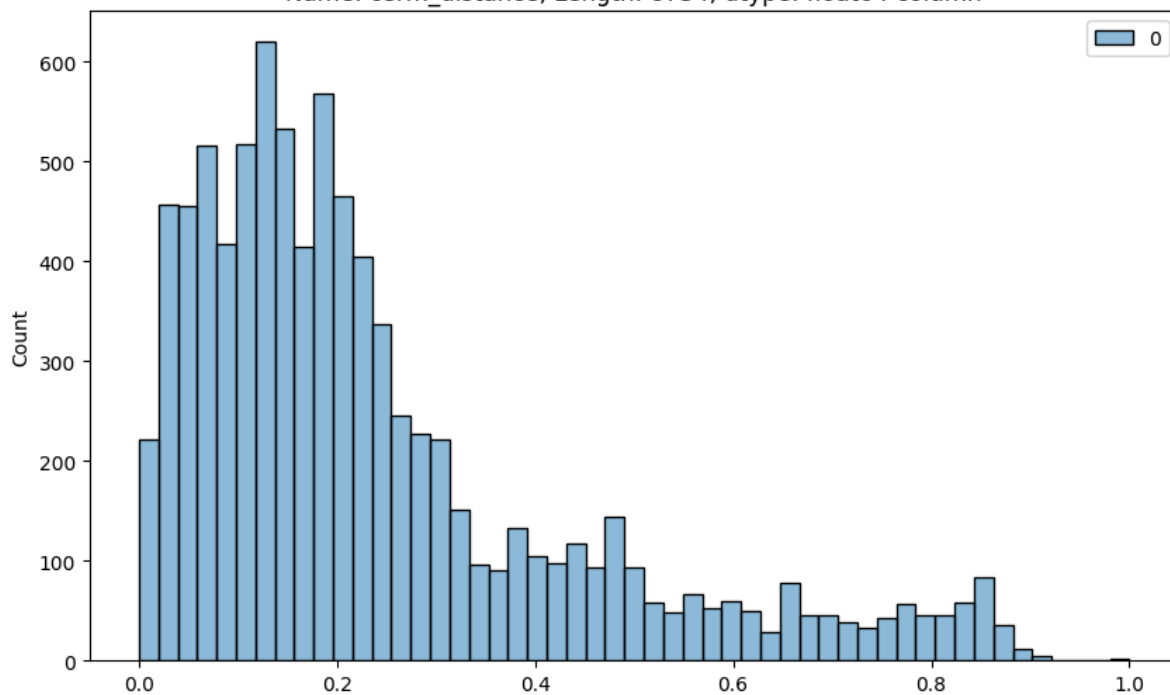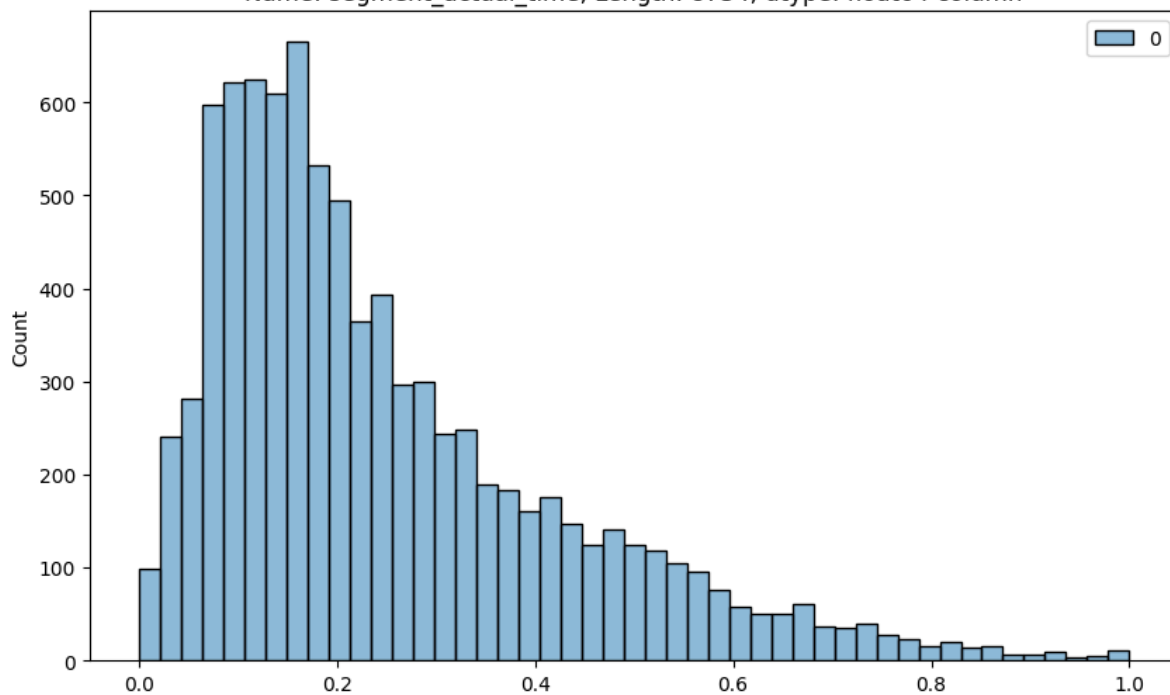


```
In [151…   plt.figure(figsize = (10, 6))
           scaler = MinMaxScaler()
           scaled = scaler.fit_transform(df['segment_actual_time'].to_numpy().reshape(-1, 1))
           sns.histplot(scaled)
           plt.title(f"Normalized {df['segment_actual_time']} column")
           plt.plot()
```

```
Out[1518]:   []
```

```
Normalized 0      141.0
          1       59.0
          2       60.0
          3       24.0
          4       64.0
                   ...
       8729       37.0
       8730       82.0
       8731       21.0
       8732      281.0
       8733      274.0
Name: segment_actual_time, Length: 8734, dtype: float64 column
```
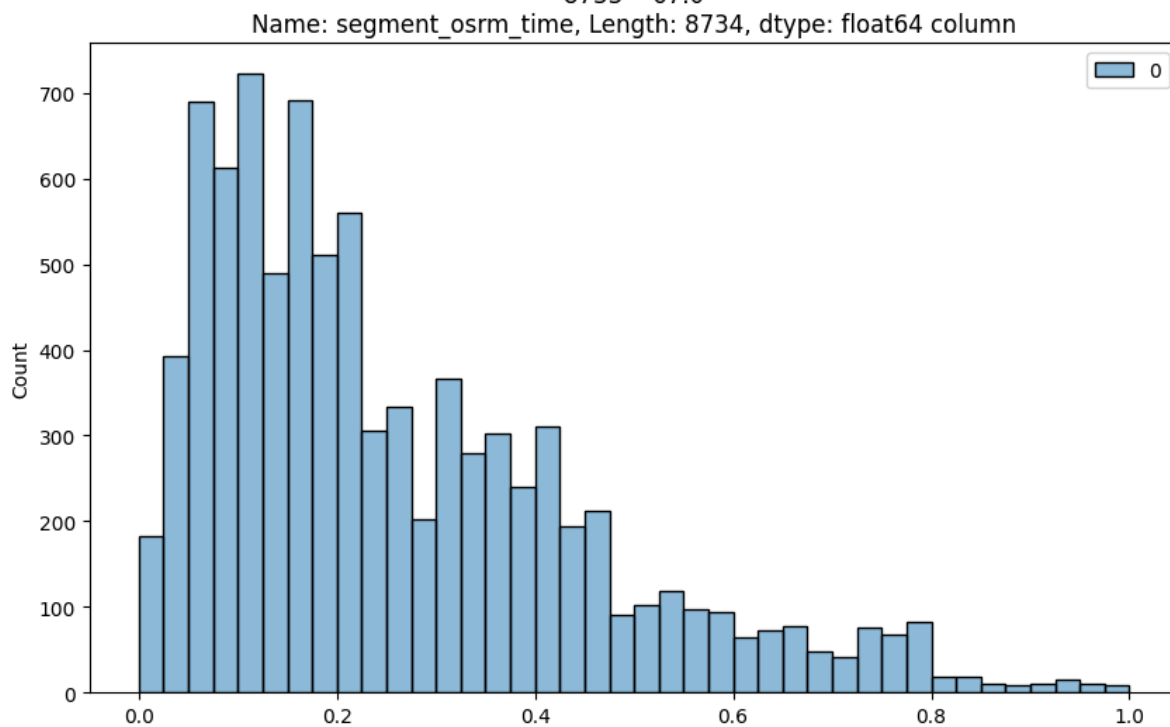


```python
plt.figure(figsize = (10, 6))
scaler = MinMaxScaler()
scaled = scaler.fit_transform(df['segment_osrm_time'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Normalized {df['segment_osrm_time']} column")
plt.plot()
```

Out[1519]:   []

```
Normalized 0      65.0
            1      16.0
            2      23.0
            3      13.0
            4      34.0
                  ...
         8729   16.0
         8730   62.0
         8731   11.0
         8732   88.0
         8733   67.0
Name: segment_osrm_time, Length: 8734, dtype: float64 column
```
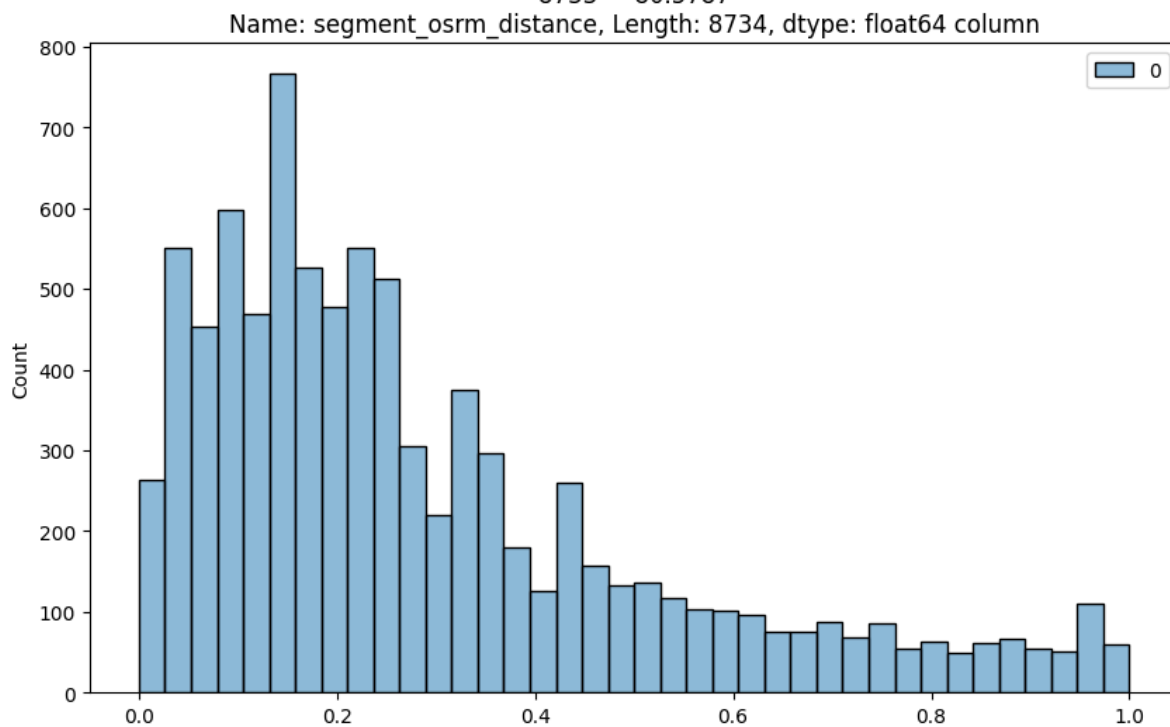


```python
plt.figure(figsize = (10, 6))
scaler = MinMaxScaler()
scaled = scaler.fit_transform(df['segment_osrm_distance'].to_numpy().reshape(-1, 1)
sns.histplot(scaled)
plt.title(f"Normalized {df['segment_osrm_distance']} column")
plt.plot()
```

Out[1520]:   []

```
Normalized 0      84.1894
           1      19.8766
           2      28.0647
           3      12.0184
           4      28.9203
                    ...
        8729      20.5065
        8730      64.8551
        8731      16.0883
        8732     104.8866
        8733      80.5787
Name: segment_osrm_distance, Length: 8734, dtype: float64 column
```
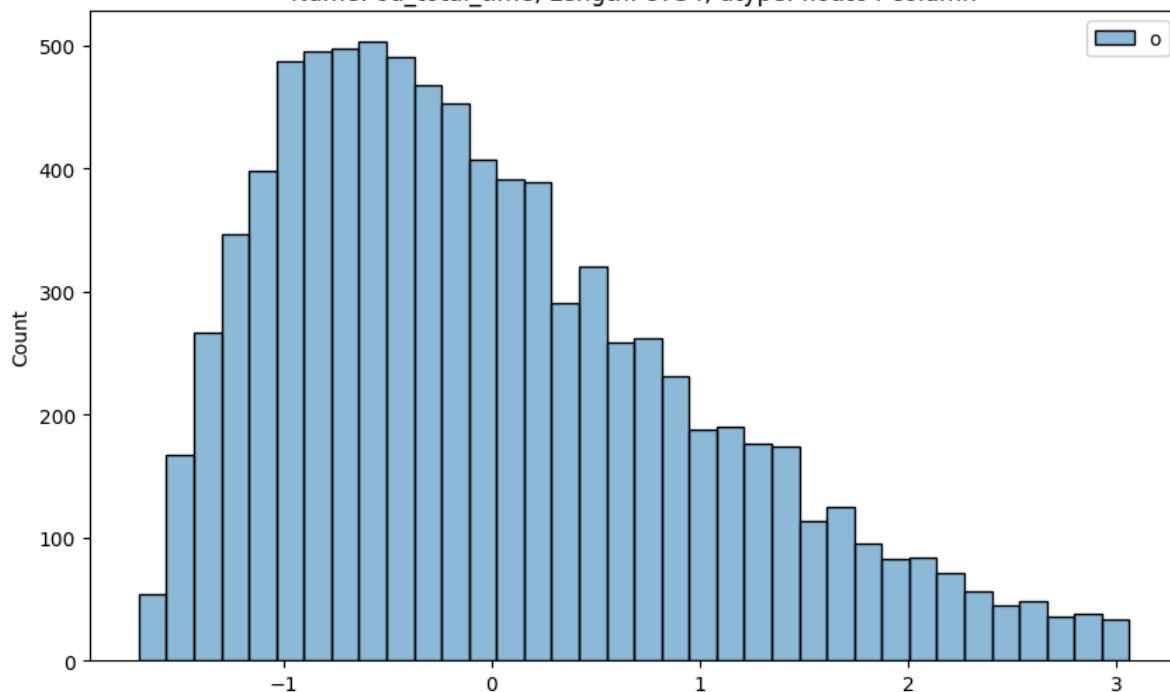

Standardization of numerical columns

```python
In [152…  plt.figure(figsize = (10, 6))
          scaler = StandardScaler()
          scaled = scaler.fit_transform(df['od_total_time'].to_numpy().reshape(-1, 1))
          sns.histplot(scaled)
          plt.title(f"Standardized {df['od_total_time']} column")
          plt.legend('od_total_time')
          plt.plot()
```

Out[1521]: []

```
Standardized 0       181.61
             1       100.49
             2       190.49
             3        98.01
             4       146.84
                        ...
          8729        88.22
          8730       258.03
          8731        60.59
          8732       422.12
          8733       354.40
Name: od_total_time, Length: 8734, dtype: float64 column
```
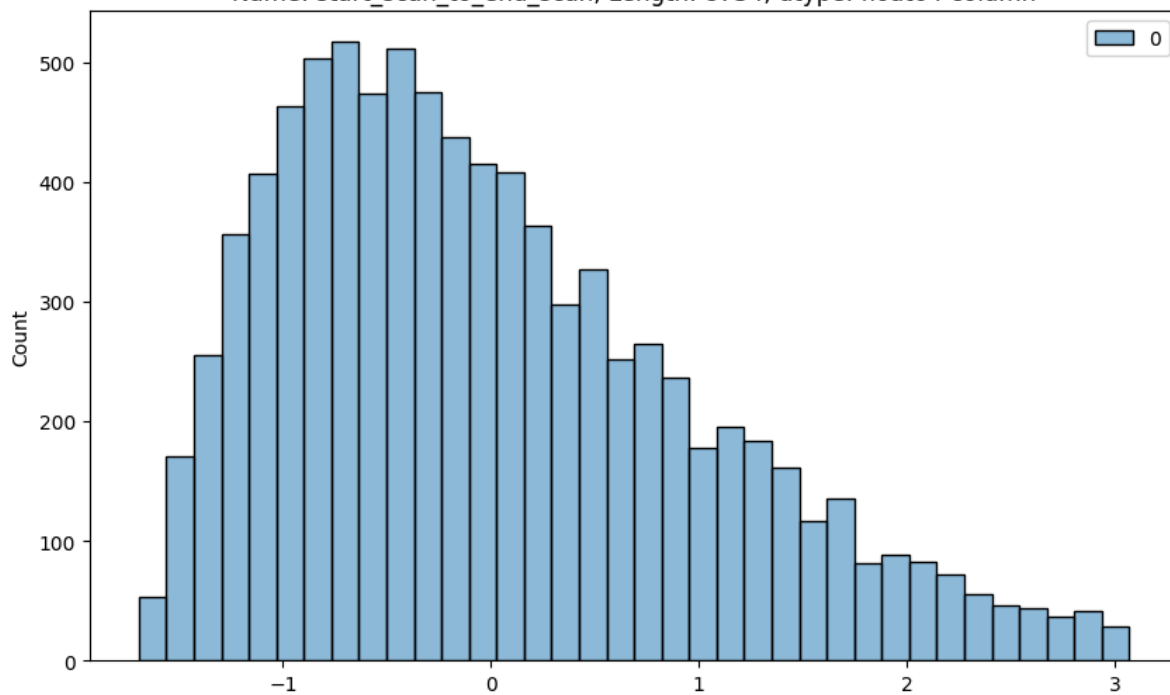


```python
plt.figure(figsize = (10, 6))
scaler = StandardScaler()
scaled = scaler.fit_transform(df['start_scan_to_end_scan'].to_numpy().reshape(-1, 1
sns.histplot(scaled)
plt.title(f"Standardized {df['start_scan_to_end_scan']} column")
plt.plot()
```

Out[1522]:   []

```
Standardized 0      180.0
             1      100.0
             2      189.0
             3       98.0
             4      146.0
                    ...
          8729       88.0
          8730      257.0
          8731       60.0
          8732      421.0
          8733      353.0
Name: start_scan_to_end_scan, Length: 8734, dtype: float64 column
```
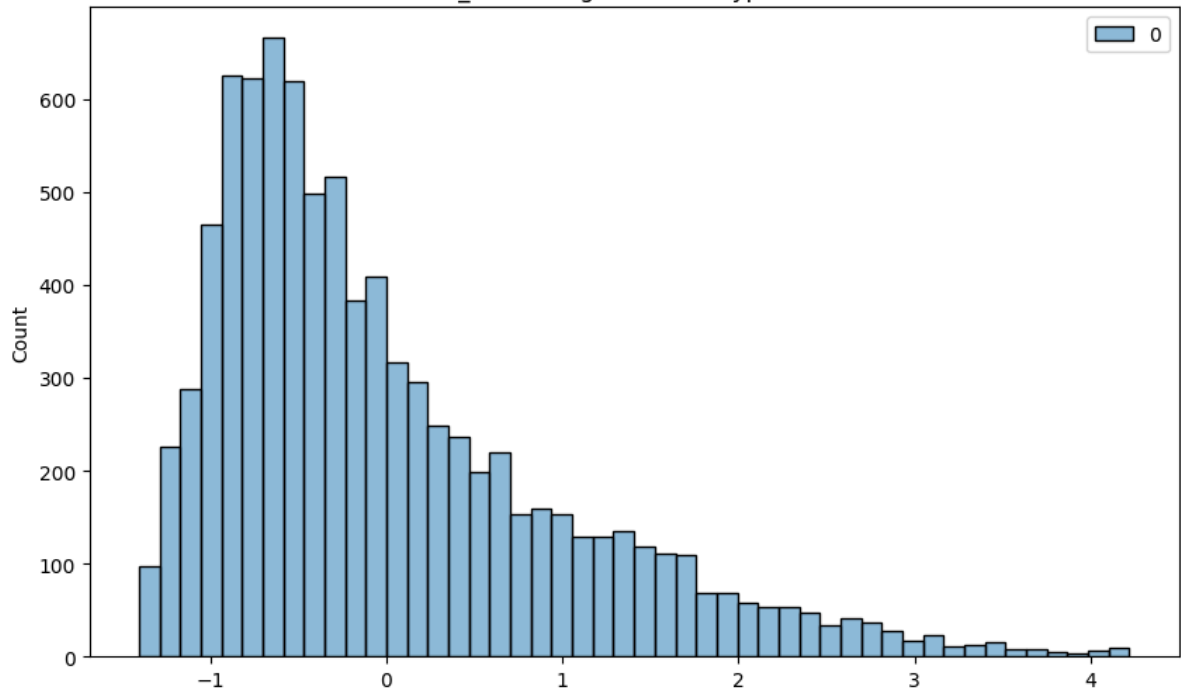

In [152…
```python
plt.figure(figsize = (10, 6))
scaler = StandardScaler()
scaled = scaler.fit_transform(df['actual_time'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Standardized {df['actual_time']} column")
plt.plot()
```

Out[1523]:   []

```
Standardized 0      143.0
             1       59.0
             2       61.0
             3       24.0
             4       64.0
                      ...
          8729       38.0
          8730       83.0
          8731       21.0
          8732      282.0
          8733      275.0
Name: actual_time, Length: 8734, dtype: float64 column
```
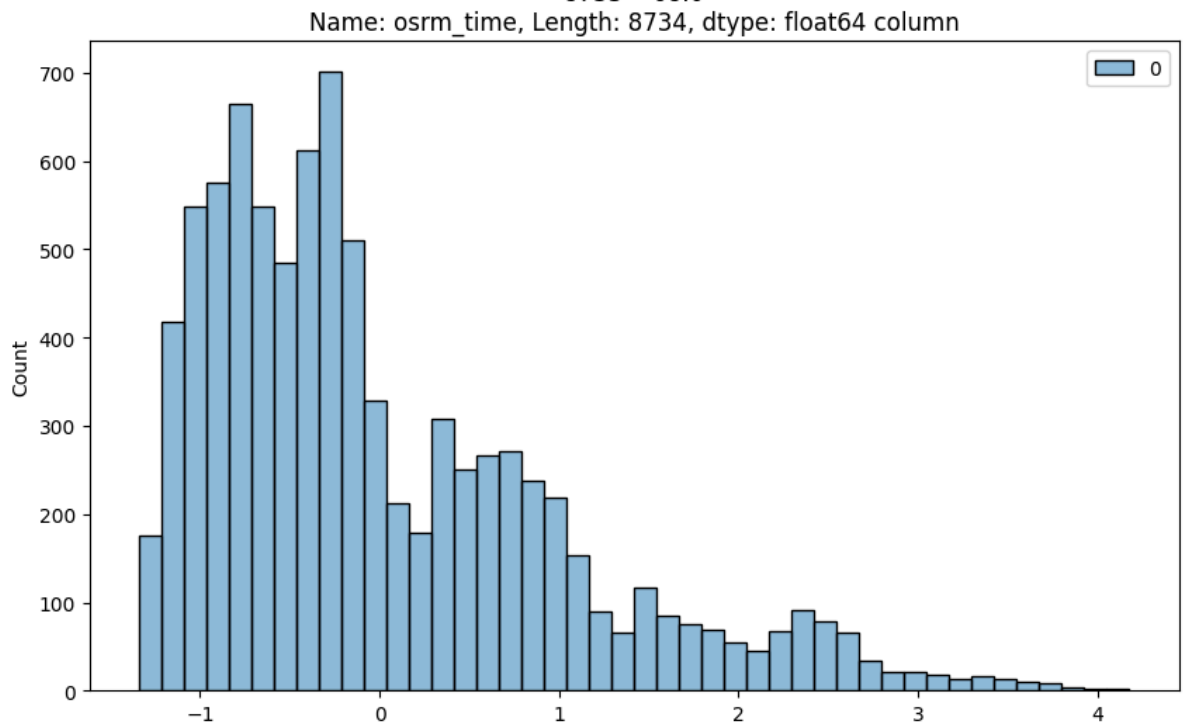

In [152…
```python
plt.figure(figsize = (10, 6))
scaler = StandardScaler()
scaled = scaler.fit_transform(df['osrm_time'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Standardized {df['osrm_time']} column")
plt.plot()
```

Out[1524]:   []

```
Standardized 0      68.0
             1      15.0
             2      23.0
             3      13.0
             4      34.0
                    ...
          8729      16.0
          8730      62.0
          8731      12.0
          8732      48.0
          8733      68.0
Name: osrm_time, Length: 8734, dtype: float64 column
```


In [152…
```python
plt.figure(figsize = (10, 6))
scaler = StandardScaler()
scaled = scaler.fit_transform(df['osrm_distance'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Standardized {df['osrm_distance']} column")
plt.plot()
```

Out[1525]:   []

```
Standardized 0      85.1110
            1      19.6800
            2      28.0647
            3      12.0184
            4      28.9203
                   ...
         8729      20.5065
         8730      73.4630
         8731      16.0882
         8732      58.9037
         8733      80.5787
Name: osrm_distance, Length: 8734, dtype: float64 column
```
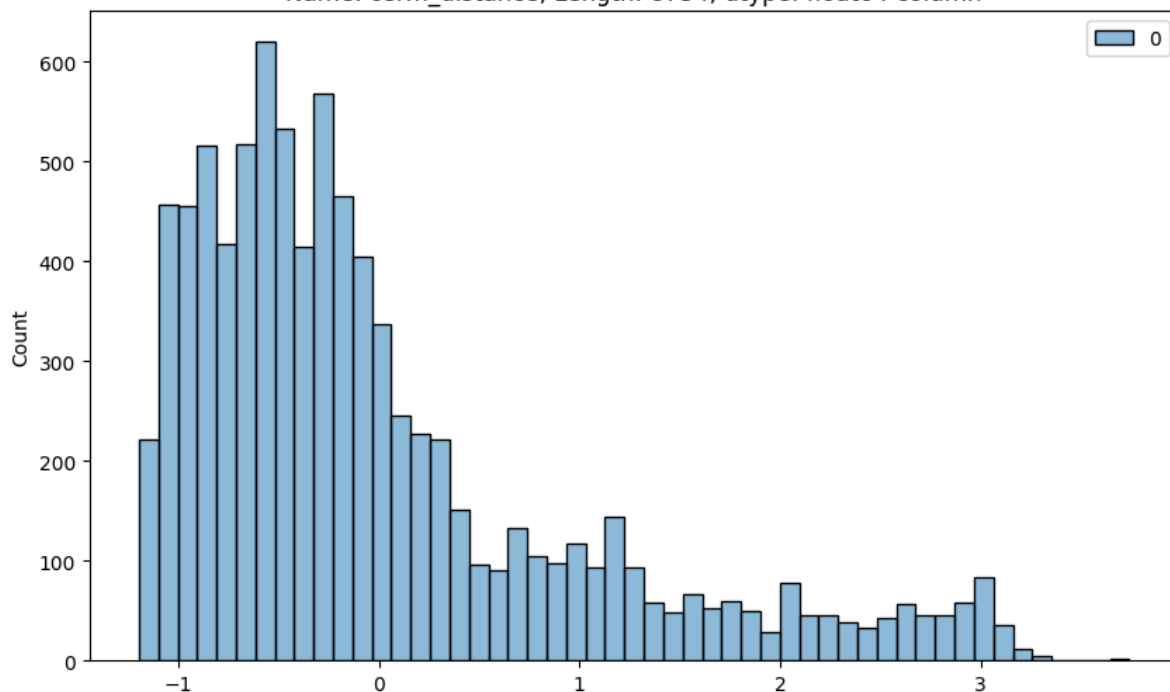


```
In [152…  plt.figure(figsize = (10, 6))
          scaler = StandardScaler()
          scaled = scaler.fit_transform(df['segment_actual_time'].to_numpy().reshape(-1, 1))
          sns.histplot(scaled)
          plt.title(f"Standardized {df['segment_actual_time']} column")
          plt.plot()
```

```
Out[1526]:   []
```

```
Standardized 0     141.0
            1      59.0
            2      60.0
            3      24.0
            4      64.0
                    ...
            8729    37.0
            8730    82.0
            8731    21.0
            8732   281.0
            8733   274.0
Name: segment_actual_time, Length: 8734, dtype: float64 column
```
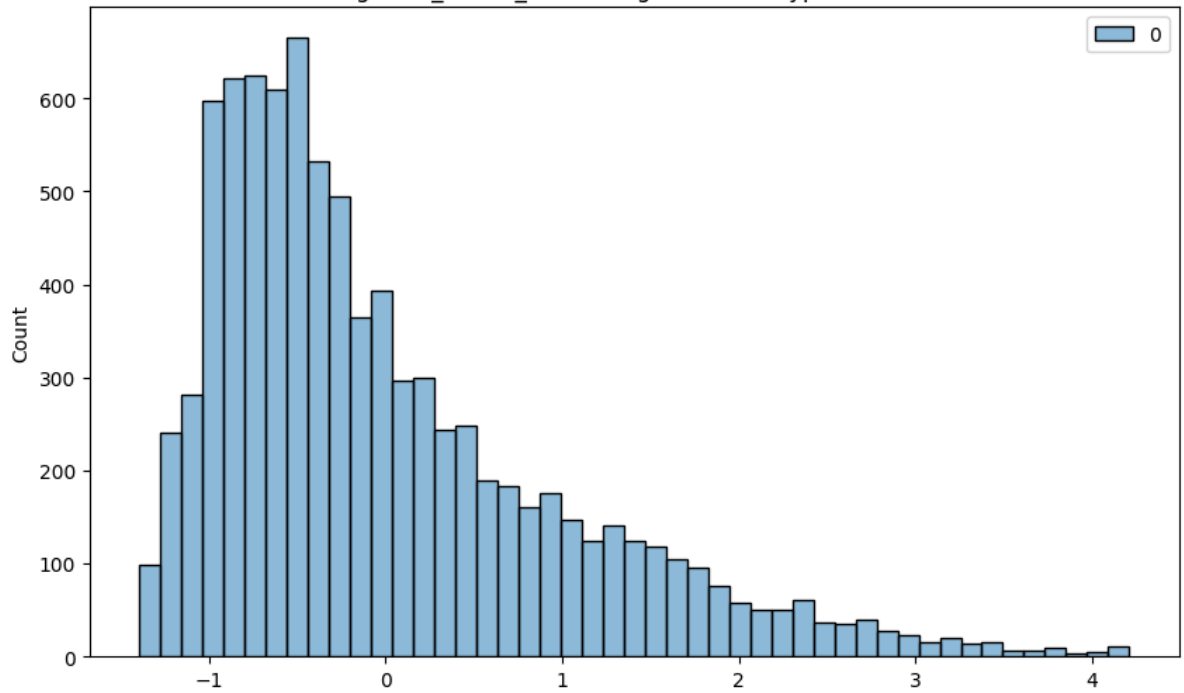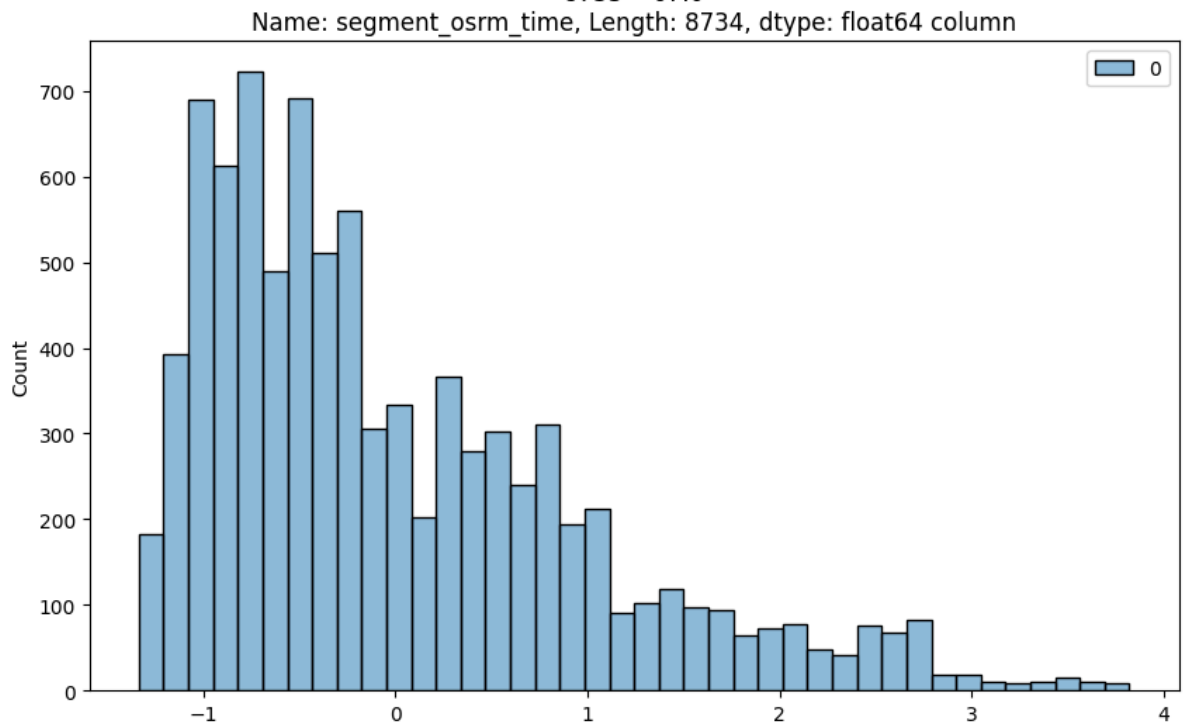


```python
plt.figure(figsize = (10, 6))
scaler = StandardScaler()
scaled = scaler.fit_transform(df['segment_osrm_time'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Standardized {df['segment_osrm_time']} column")
plt.plot()
```

Out[1527]:   []

```
Standardized  0      65.0
              1      16.0
              2      23.0
              3      13.0
              4      34.0
                      ...
           8729      16.0
           8730      62.0
           8731      11.0
           8732      88.0
           8733      67.0
Name: segment_osrm_time, Length: 8734, dtype: float64 column
```



```python
plt.figure(figsize = (10, 6))
scaler = StandardScaler()
scaled = scaler.fit_transform(df['segment_osrm_distance'].to_numpy().reshape(-1, 1))
sns.histplot(scaled)
plt.title(f"Standardized {df['segment_osrm_distance']} column")
plt.plot()
```

Out[1528]:   []

```
Standardized 0       84.1894
            1       19.8766
            2       28.0647
            3       12.0184
            4       28.9203
                     ...
         8729       20.5065
         8730       64.8551
         8731       16.0883
         8732      104.8866
         8733       80.5787
Name: segment_osrm_distance, Length: 8734, dtype: float64 column
```
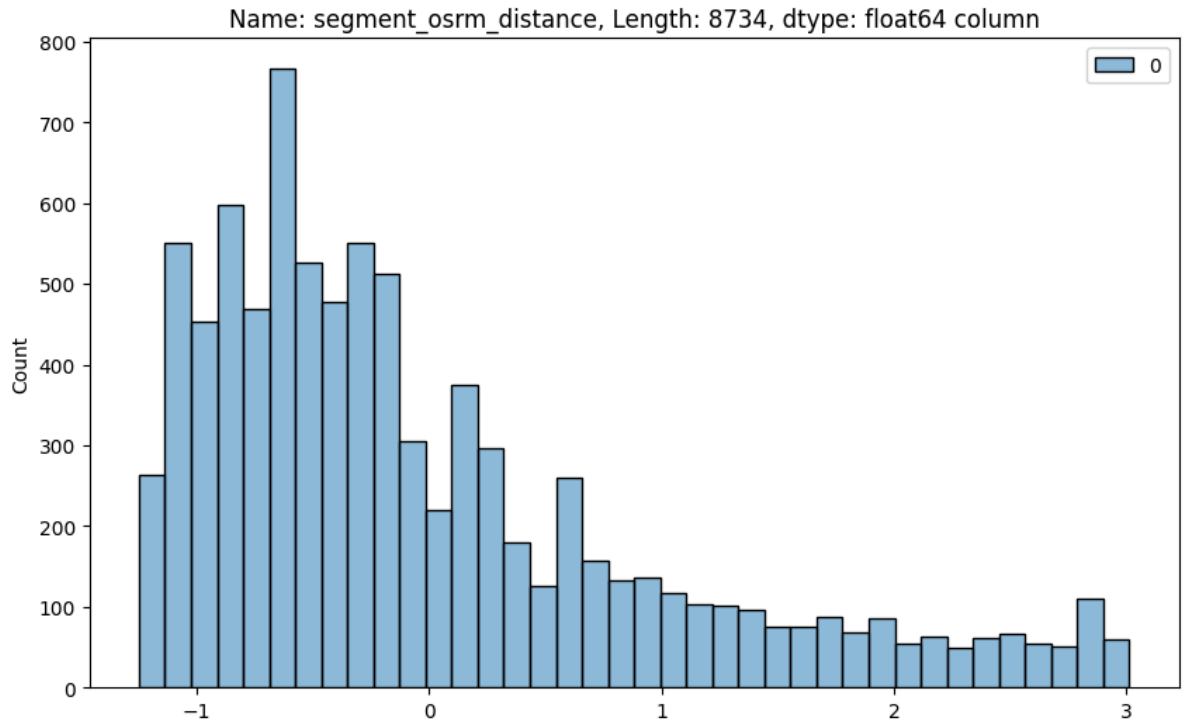

In [152…   df

Out[1529]:

	trip_uuid	data	route_type	start_scan_to_end_scan	actual_distance_to_destination
0	trip-153671042288605164	1	0	180.0	73.18691
1	trip-153671046011330457	1	0	100.0	17.175274
2	trip-153671055416136166	1	0	189.0	24.59704
3	trip-153671066201138152	1	0	98.0	9.10051
4	trip-153671066826362165	1	0	146.0	22.42421
...	...	...	...	...	.
8729	trip-153861091843037040	0	0	88.0	17.76024
8730	trip-153861095625827784	0	0	257.0	57.76233
8731	trip-153861104386292051	0	0	60.0	15.513784
8732	trip-153861106442901555	0	0	421.0	38.68483
8733	trip-153861118270144424	0	1	353.0	66.08153

8734 rows × 22 columns
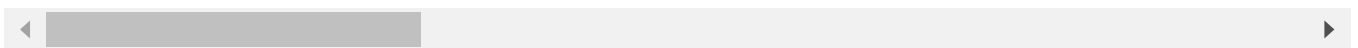
Business Insights

- The data is given from the period '2018-09-12 00:00:16' to '2018-10-08 03:00:24'.
- There are about 14817 unique trip IDs, 1508 unique source centers, 1481 unique destination_centers, 690 unique source cities, 806 unique destination cities.
- Most of the data is for testing than for training.
- Most common route type is Carting.
- The number of trips start increasing after the noon, becomes maximum at 10 P.M and then start decreasing.
- Most orders come mid-month. That means customers usually make more orders in the mid of the month.
- Most orders are sourced from the states like Maharashtra, Karnataka, Haryana, Tamil Nadu, Telangana
- Maximum number of trips originated from Mumbai city followed by Gurgaon Delhi, Bengaluru and Bhiwandi. That means that the seller base is strong in these cities.
- Maximum number of trips ended in Maharashtra state followed by Karnataka, Haryana, Tamil Nadu and Uttar Pradesh. That means that the number of orders placed in these states is significantly high.

- Most orders in terms of destination are coming from cities like bengaluru, mumbai, gurgaon, bangalore, Delhi.
- Maximum number of trips ended in Mumbai city followed by Bengaluru, Gurgaon, Delhi and Chennai. That means that the number of orders placed in these cities is significantly high.

- Most orders in terms of destination are coming from cities like bengaluru, mumbai, gurgaon, bangalore, Delhi.
- Maximum number of trips ended in Mumbai city followed by Bengaluru, Gurgaon, Delhi and Chennai. That means that the number of orders placed in these cities is significantly high.