Delhivery Analysis by Kumud

July 29, 2025

[1]:
```
import pandas as pd
```

[2]:
```
df = pd.read_csv(r"C:\Users\kpswe\Downloads\delhivery_data.csv")
```

0.1 Basic data cleaning and exploration

[3]:
```
df.head()
```

[3]:
```
       data          trip_creation_time  \
0  training  2018-09-20 02:35:36.476840
1  training  2018-09-20 02:35:36.476840
2  training  2018-09-20 02:35:36.476840
3  training  2018-09-20 02:35:36.476840
4  training  2018-09-20 02:35:36.476840

                                route_schedule_uuid route_type  \
0  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
1  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
2  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
3  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
4  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting

                trip_uuid source_center                source_name  \
0  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
1  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
2  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
3  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
4  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)

  destination_center             destination_name  \
0       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
1       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
2       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
3       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
4       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)

               od_start_time  …           cutoff_timestamp  \
0  2018-09-20 03:21:32.418600  …       2018-09-20 04:27:55
```

```
1  2018-09-20 03:21:32.418600  …         2018-09-20 04:17:55
2  2018-09-20 03:21:32.418600  …  2018-09-20 04:01:19.505586
3  2018-09-20 03:21:32.418600  …         2018-09-20 03:39:57
4  2018-09-20 03:21:32.418600  …         2018-09-20 03:33:55

   actual_distance_to_destination  actual_time  osrm_time osrm_distance  \
0                       10.435660         14.0       11.0       11.9653
1                       18.936842         24.0       20.0       21.7243
2                       27.637279         40.0       28.0       32.5395
3                       36.118028         62.0       40.0       45.5620
4                       39.386040         68.0       44.0       54.2181

     factor  segment_actual_time  segment_osrm_time  segment_osrm_distance  \
0  1.272727                 14.0               11.0                11.9653
1  1.200000                 10.0                9.0                 9.7590
2  1.428571                 16.0                7.0                10.8152
3  1.550000                 21.0               12.0                13.0224
4  1.545455                  6.0                5.0                 3.9153

   segment_factor
0        1.272727
1        1.111111
2        2.285714
3        1.750000
4        1.200000

[5 rows x 24 columns]
```

[4]:
```
df.columns
```

[4]:
```
Index(['data', 'trip_creation_time', 'route_schedule_uuid', 'route_type',
       'trip_uuid', 'source_center', 'source_name', 'destination_center',
       'destination_name', 'od_start_time', 'od_end_time',
       'start_scan_to_end_scan', 'is_cutoff', 'cutoff_factor',
       'cutoff_timestamp', 'actual_distance_to_destination', 'actual_time',
       'osrm_time', 'osrm_distance', 'factor', 'segment_actual_time',
       'segment_osrm_time', 'segment_osrm_distance', 'segment_factor'],
      dtype='object')
```

[5]:
```
print(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
None
```

[6]:
```
print(df.describe(include='all'))
```

```
            data          trip_creation_time  \
count     144867                      144867
unique         2                       14817
top     training  2018-09-28 05:23:15.359220
freq      104858                         101
mean         NaN                         NaN
std          NaN                         NaN
min          NaN                         NaN
25%          NaN                         NaN
50%          NaN                         NaN
75%          NaN                         NaN
max          NaN                         NaN

                                    route_schedule_uuid route_type  \
count                                            144867     144867
unique                                             1504          2
top     thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f…        FTL
freq                                               1812      99660
mean                                                NaN        NaN
std                                                 NaN        NaN
```

```
min                                                   NaN        NaN
25%                                                   NaN        NaN
50%                                                   NaN        NaN
75%                                                   NaN        NaN
max                                                   NaN        NaN

                     trip_uuid source_center                    source_name  \
count                   144867        144867                         144574
unique                   14817          1508                           1498
top     trip-153811219535896559  IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
freq                       101         23347                          23347
mean                       NaN           NaN                            NaN
std                        NaN           NaN                            NaN
min                        NaN           NaN                            NaN
25%                        NaN           NaN                            NaN
50%                        NaN           NaN                            NaN
75%                        NaN           NaN                            NaN
max                        NaN           NaN                            NaN

       destination_center              destination_name  \
count              144867                        144606
unique               1481                          1468
top          IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
freq                15192                         15192
mean                  NaN                           NaN
std                   NaN                           NaN
min                   NaN                           NaN
25%                   NaN                           NaN
50%                   NaN                           NaN
75%                   NaN                           NaN
max                   NaN                           NaN

                    od_start_time  …     cutoff_timestamp  \
count                      144867  …               144867
unique                      26369  …                93180
top     2018-09-21 18:37:09.322207  …  2018-09-24 05:19:20
freq                           81  …                   40
mean                          NaN  …                  NaN
std                           NaN  …                  NaN
min                           NaN  …                  NaN
25%                           NaN  …                  NaN
50%                           NaN  …                  NaN
75%                           NaN  …                  NaN
max                           NaN  …                  NaN

        actual_distance_to_destination    actual_time      osrm_time  \
count                   144867.000000  144867.000000  144867.000000
unique                            NaN            NaN            NaN
```

```
top                                NaN            NaN            NaN
freq                               NaN            NaN            NaN
mean                        234.073372     416.927527     213.868272
std                         344.990009     598.103621     308.011085
min                           9.000045       9.000000       6.000000
25%                          23.355874      51.000000      27.000000
50%                          66.126571     132.000000      64.000000
75%                         286.708875     513.000000     257.000000
max                        1927.447705    4532.000000    1686.000000

        osrm_distance         factor  segment_actual_time  segment_osrm_time  \
count   144867.000000  144867.000000        144867.000000      144867.000000
unique            NaN            NaN                  NaN                NaN
top               NaN            NaN                  NaN                NaN
freq              NaN            NaN                  NaN                NaN
mean       284.771297       2.120107            36.196111          18.507548
std        421.119294       1.715421            53.571158          14.775960
min          9.008200       0.144000          -244.000000           0.000000
25%         29.914700       1.604264            20.000000          11.000000
50%         78.525800       1.857143            29.000000          17.000000
75%        343.193250       2.213483            40.000000          22.000000
max       2326.199100      77.387097          3051.000000        1611.000000

        segment_osrm_distance  segment_factor
count             144867.00000   144867.000000
unique                     NaN             NaN
top                        NaN             NaN
freq                       NaN             NaN
mean                  22.82902        2.218368
std                   17.86066        4.847530
min                    0.00000      -23.444444
25%                   12.07010        1.347826
50%                   23.51300        1.684211
75%                   27.81325        2.250000
max                 2191.40370      574.250000

[11 rows x 24 columns]
```

0.2 Handle Missing Values

[7]:
```
print(df.isnull().sum())
```

```
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
```

[8]:
```
df['destination_name'].mode()[0]
```

[8]:
```
'Gurgaon_Bilaspur_HB (Haryana)'
```

[9]:
```
df.groupby('destination_name')['destination_name'].count().
 ↪sort_values(ascending=False)
```

[9]:
```
destination_name
Gurgaon_Bilaspur_HB (Haryana)        15192
Bangalore_Nelmngla_H (Karnataka)     11019
Bhiwandi_Mankoli_HB (Maharashtra)     5492
Hyderabad_Shamshbd_H (Telangana)      5142
Kolkata_Dankuni_HB (West Bengal)      4892
                                     …
Manthuka_Central_D_1 (Kerala)            1
Hyd_Trimulgherry_Dc (Telangana)          1
Durg_Bhilai_DC (Chhattisgarh)            1
North Delhi (Delhi)                      1
Tilhar_SingCLNY_D (Uttar Pradesh)        1
Name: destination_name, Length: 1468, dtype: int64
```

[10]:
```
imputed_df = df.copy(deep=True)
imputed_df.head()
```

[10]:
```
       data          trip_creation_time  \
0  training  2018-09-20 02:35:36.476840
1  training  2018-09-20 02:35:36.476840
2  training  2018-09-20 02:35:36.476840
```

```
3  training  2018-09-20 02:35:36.476840
4  training  2018-09-20 02:35:36.476840

                                route_schedule_uuid route_type  \
0  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
1  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
2  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
3  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
4  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting

                 trip_uuid source_center                source_name  \
0  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
1  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
2  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
3  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
4  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)

  destination_center              destination_name  \
0       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
1       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
2       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
3       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
4       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)

               od_start_time  …            cutoff_timestamp  \
0  2018-09-20 03:21:32.418600  …         2018-09-20 04:27:55
1  2018-09-20 03:21:32.418600  …         2018-09-20 04:17:55
2  2018-09-20 03:21:32.418600  …  2018-09-20 04:01:19.505586
3  2018-09-20 03:21:32.418600  …         2018-09-20 03:39:57
4  2018-09-20 03:21:32.418600  …         2018-09-20 03:33:55

   actual_distance_to_destination  actual_time  osrm_time osrm_distance  \
0                       10.435660         14.0       11.0       11.9653
1                       18.936842         24.0       20.0       21.7243
2                       27.637279         40.0       28.0       32.5395
3                       36.118028         62.0       40.0       45.5620
4                       39.386040         68.0       44.0       54.2181

     factor  segment_actual_time  segment_osrm_time  segment_osrm_distance  \
0  1.272727                 14.0               11.0                11.9653
1  1.200000                 10.0                9.0                 9.7590
2  1.428571                 16.0                7.0                10.8152
3  1.550000                 21.0               12.0                13.0224
4  1.545455                  6.0                5.0                 3.9153

   segment_factor
0        1.272727
```

```
1        1.111111
2        2.285714
3        1.750000
4        1.200000

[5 rows x 24 columns]
```

[11]:
```
imputed_df.isnull().sum()
```

[11]:
```
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
```

[12]:
```
selectd_cols = ['source_name','destination_name']

for col in selectd_cols:
    if imputed_df[col].isnull().any():
        mode_val = imputed_df[col].mode()[0]
        imputed_df[col].fillna(mode_val, inplace=True)
```

```
C:\Users\kpswe\AppData\Local\Temp\ipykernel_19628\1641919377.py:6:
FutureWarning: A value is trying to be set on a copy of a DataFrame or Series
through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work
because the intermediate object on which we are setting values always behaves as
a copy.
```

```
For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.


  imputed_df[col].fillna(mode_val, inplace=True)
```

[13]:
```
imputed_df.isnull().sum()
```

[13]:
```
data                                0
trip_creation_time                  0
route_schedule_uuid                 0
route_type                          0
trip_uuid                           0
source_center                       0
source_name                         0
destination_center                  0
destination_name                    0
od_start_time                       0
od_end_time                         0
start_scan_to_end_scan              0
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
```

0.3 Merging the DF

[14]:
```
agg_df1 = imputed_df.
 ↪groupby(['trip_uuid','source_center','destination_center']).agg({
    'actual_time': 'sum',
    'segment_actual_time': 'sum',
    'osrm_time': 'sum',
    'segment_osrm_time': 'sum',
    'osrm_distance': 'sum',
    'segment_osrm_distance': 'sum',
    'start_scan_to_end_scan': 'max',
```

```
    'od_start_time': 'min',
    'od_end_time': 'max'
}).reset_index()
agg_df1.head()
```

[14]:

	trip_uuid	source_center	destination_center	actual_time	segment_actual_time	osrm_time	segment_osrm_time	osrm_distance	segment_osrm_distance	start_scan_to_end_scan	od_start_time	od_end_time
0	trip-153671041653548748	IND209304AAA	IND000000ACB	6484.0	728.0	3464.0	534.0	4540.1261	670.6205	1260.0	2018-09-12 16:39:46.858469	2018-09-13 13:40:23.123744
1	trip-153671041653548748	IND462022AAA	IND209304AAA	9198.0	820.0	4323.0	474.0	6037.6386	649.8528	999.0	2018-09-12 00:00:16.535741	2018-09-12 16:39:46.858469
2	trip-153671042288605164	IND561203AAB	IND562101AAA	96.0	46.0	55.0	26.0	60.3157	28.1995	58.0	2018-09-12 02:03:09.655591	2018-09-12 03:01:59.598855
3	trip-153671042288605164	IND572101AAA	IND561203AAB	303.0	95.0	155.0	39.0	209.1151	55.9899	122.0	2018-09-12 00:00:22.886430	2018-09-12 02:03:09.655591
4	trip-153671043369099517	IND000000ACB	IND160002AAC	2601.0	608.0	1427.0	231.0	1975.7409	317.7408	834.0	2018-09-14 03:40:17.106733	2018-09-14 17:34:55.442454

[15]:
```
agg_df2 = imputed_df.groupby('trip_uuid').agg({
    'actual_time': 'sum',
    'segment_actual_time': 'sum',
    'osrm_time': 'sum',
    'segment_osrm_time': 'sum',
    'osrm_distance': 'sum',
    'segment_osrm_distance': 'sum',
    'start_scan_to_end_scan': 'max',
    'od_start_time': 'min',
    'od_end_time': 'max'
}).reset_index()
agg_df2.head()
```

[15]:

	trip_uuid	actual_time	segment_actual_time	osrm_time	segment_osrm_time	osrm_distance	segment_osrm_distance	start_scan_to_end_scan	od_start_time	od_end_time
0	trip-153671041653548748	15682.0	1548.0	7787.0	1008.0	10577.7647	1320.4733	1260.0	2018-09-12 00:00:16.535741	2018-09-13 13:40:23.123744
1	trip-153671042288605164	399.0	141.0	210.0	65.0	269.4308	84.1894	122.0	2018-09-12 00:00:22.886430	2018-09-12 03:01:59.598855
2	trip-153671043369099517	112225.0	3308.0	65768.0	1941.0	89447.2488	2545.2678	3099.0	2018-09-12 00:00:33.691250	2018-09-14 17:34:55.442454
3	trip-153671046011330457	82.0	59.0	24.0	16.0	31.6475	19.8766	100.0	2018-09-12 00:01:00.113710	2018-09-12 01:41:29.809822
4	trip-153671052974046625	556.0	340.0	207.0	115.0	266.2914	146.7919	485.0	2018-09-12 00:02:09.740725	2018-09-12 12:00:30.683231

[16]:
```
agg_df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14817 entries, 0 to 14816
Data columns (total 10 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   trip_uuid               14817 non-null  object
 1   actual_time             14817 non-null  float64
 2   segment_actual_time     14817 non-null  float64
 3   osrm_time               14817 non-null  float64
 4   segment_osrm_time       14817 non-null  float64
 5   osrm_distance           14817 non-null  float64
 6   segment_osrm_distance   14817 non-null  float64
 7   start_scan_to_end_scan  14817 non-null  float64
 8   od_start_time           14817 non-null  object
 9   od_end_time             14817 non-null  object
dtypes: float64(7), object(3)
memory usage: 1.1+ MB
```

[17]:
```
agg_df2['od_start_time'] = pd.to_datetime(agg_df2['od_start_time'])
agg_df2['od_end_time'] = pd.to_datetime(agg_df2['od_end_time'])
```

[18]:
```
agg_df2['trip_duration_hours'] = (agg_df2['od_end_time'] -␣
 ↪agg_df2['od_start_time']).dt.total_seconds() / 3600
agg_df2.head()
```

[18]:

	trip_uuid	actual_time	segment_actual_time	osrm_time	segment_osrm_time	osrm_distance	segment_osrm_distance	start_scan_to_end_scan	od_start_time	od_end_time	trip_duration_hours
0	trip-153671041653548748	15682.0	1548.0	7787.0	1008.0	10577.7647	1320.4733	1260.0	2018-09-12 00:00:16.535741	2018-09-13 13:40:23.123744	37.668497
1	trip-153671042288605164	399.0	141.0	210.0	65.0	269.4308	84.1894	122.0	2018-09-12 00:00:22.886430	2018-09-12 03:01:59.598855	3.026865
2	trip-153671043369099517	112225.0	3308.0	65768.0	1941.0	89447.2488	2545.2678	3099.0	2018-09-12 00:00:33.691250	2018-09-14 17:34:55.442454	65.572709
3	trip-153671046011330457	82.0	59.0	24.0	16.0	31.6475	19.8766	100.0	2018-09-12 00:01:00.113710	2018-09-12 01:41:29.809822	1.674916
4	trip-153671052974046625	556.0	340.0	207.0	115.0	266.2914	146.7919	485.0	2018-09-12 00:02:09.740725	2018-09-12 12:00:30.683231	11.972484

[19]:
```
agg_df2['od_duration'] = (agg_df2['od_end_time'] - agg_df2['od_start_time']).dt.
 ↪total_seconds() / 60
agg_df2.head()
```

[19]:

	trip_uuid	actual_time	segment_actual_time	osrm_time	segment_osrm_time	osrm_distance	segment_osrm_distance	start_scan_to_end_scan	od_start_time	od_end_time	trip_duration_hours
0	trip-153671041653548748	15682.0	1548.0	7787.0	1008.0	10577.7647	1320.4733	1260.0	2018-09-12 00:00:16.535741	2018-09-13 13:40:23.123744	37.668497
1	trip-153671042288605164	399.0	141.0	210.0	65.0	269.4308	84.1894	122.0	2018-09-12 00:00:22.886430	2018-09-12 03:01:59.598855	3.026865
2	trip-153671043369099517	112225.0	3308.0	65768.0	1941.0	89447.2488	2545.2678	3099.0	2018-09-12 00:00:33.691250	2018-09-14 17:34:55.442454	65.572709
3	trip-153671046011330457	82.0	59.0	24.0	16.0	31.6475	19.8766	100.0	2018-09-12 00:01:00.113710	2018-09-12 01:41:29.809822	1.674916
4	trip-153671052974046625	556.0	340.0	207.0	115.0	266.2914	146.7919	485.0	2018-09-12 00:02:09.740725	2018-09-12 12:00:30.683231	11.972484

```
0              1008.0     10577.7647              1320.4733
1                65.0       269.4308                84.1894
2              1941.0     89447.2488              2545.2678
3                16.0        31.6475                19.8766
4               115.0       266.2914               146.7919

   start_scan_to_end_scan             od_start_time  \
0                  1260.0 2018-09-12 00:00:16.535741
1                   122.0 2018-09-12 00:00:22.886430
2                  3099.0 2018-09-12 00:00:33.691250
3                   100.0 2018-09-12 00:01:00.113710
4                   485.0 2018-09-12 00:02:09.740725

                 od_end_time  trip_duration_hours  od_duration
0 2018-09-13 13:40:23.123744            37.668497  2260.109800
1 2018-09-12 03:01:59.598855             3.026865   181.611874
2 2018-09-14 17:34:55.442454            65.572709  3934.362520
3 2018-09-12 01:41:29.809822             1.674916   100.494935
4 2018-09-12 12:00:30.683231            11.972484   718.349042
```

0.4 Feature Engineering

[20]:
```
def extract_city_state(col):
    imputed_df[col + '_city'] = imputed_df[col].str.extract(r'^([^_]*)')
    imputed_df[col + '_state'] = imputed_df[col].str.extract(r'\(([^)]+)\)')

extract_city_state('destination_name')
extract_city_state('source_name')
```

[21]:
```
imputed_df.head()
```

[21]:
```
       data          trip_creation_time  \
0  training  2018-09-20 02:35:36.476840
1  training  2018-09-20 02:35:36.476840
2  training  2018-09-20 02:35:36.476840
3  training  2018-09-20 02:35:36.476840
4  training  2018-09-20 02:35:36.476840

                               route_schedule_uuid route_type  \
0  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
1  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
2  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
3  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting
4  thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3…    Carting

                 trip_uuid source_center                source_name  \
0  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
```

```
1  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
2  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
3  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)
4  trip-153741093647649320  IND388121AAA  Anand_VUNagar_DC (Gujarat)

  destination_center              destination_name  \
0       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
1       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
2       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
3       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
4       IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)

               od_start_time  … osrm_distance    factor  \
0  2018-09-20 03:21:32.418600  …       11.9653  1.272727
1  2018-09-20 03:21:32.418600  …       21.7243  1.200000
2  2018-09-20 03:21:32.418600  …       32.5395  1.428571
3  2018-09-20 03:21:32.418600  …       45.5620  1.550000
4  2018-09-20 03:21:32.418600  …       54.2181  1.545455

   segment_actual_time  segment_osrm_time segment_osrm_distance  \
0                 14.0               11.0               11.9653
1                 10.0                9.0                9.7590
2                 16.0                7.0               10.8152
3                 21.0               12.0               13.0224
4                  6.0                5.0                3.9153

   segment_factor  destination_name_city  destination_name_state  \
0        1.272727               Khambhat                 Gujarat
1        1.111111               Khambhat                 Gujarat
2        2.285714               Khambhat                 Gujarat
3        1.750000               Khambhat                 Gujarat
4        1.200000               Khambhat                 Gujarat

   source_name_city  source_name_state
0             Anand            Gujarat
1             Anand            Gujarat
2             Anand            Gujarat
3             Anand            Gujarat
4             Anand            Gujarat

[5 rows x 28 columns]
```

[22]:
```
imputed_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 144867 entries, 0 to 144866
Data columns (total 28 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                     144867 non-null  object
 7   destination_center              144867 non-null  object
 8   destination_name                144867 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
 24  destination_name_city           144867 non-null  object
 25  destination_name_state          144867 non-null  object
 26  source_name_city                144867 non-null  object
 27  source_name_state               144867 non-null  object
dtypes: bool(1), float64(10), int64(1), object(16)
memory usage: 30.0+ MB
```

0.5 Hypothesis Testing / Visual Analysis

[23]:
```python
def compare_paired_samples_from_df(df, col1, col2, alpha=0.05, test_name=""):
    """
    Accepts a DataFrame and column names, runs Shapiro and chooses paired test.
    """
    from scipy.stats import shapiro, ttest_rel, wilcoxon

    # Extract columns safely
    s1 = df[col1].dropna()
    s2 = df[col2].dropna()

    # Align them in case dropna caused mismatched indexes
    s1, s2 = s1.align(s2, join='inner')
```

```
    # Shapiro-Wilk normality tests
    normal1 = shapiro(s1).pvalue > alpha
    normal2 = shapiro(s2).pvalue > alpha

    if normal1 and normal2:
        stat, p = ttest_rel(s1, s2)
        test_used = 'ttest_rel'
        print("Data is Normally distributed")
    else:
        stat, p = wilcoxon(s1, s2)
        test_used = 'wilcoxon'
        print("Data is not Normally distributed")

    print(f"\n {test_name or f'{col1} vs {col2}'}:")
    print(f"Used Test: {test_used}")
    print(f"P-Value: {p:.5f}")
    if p < alpha:
        print("Significant difference - reject null hypothesis.")
    else:
        print("No significant difference - fail to reject null hypothesis.")

    return test_used, p
```

[24]:
```
compare_paired_samples_from_df(agg_df2, 'od_duration',␣
↪'start_scan_to_end_scan', test_name="Duration vs Scan Duration")
```

```
Data is not Normally distributed

 Duration vs Scan Duration:
Used Test: wilcoxon
P-Value: 0.00000
Significant difference - reject null hypothesis.

C:\ProgramData\anaconda3\Lib\site-packages\scipy\stats\_axis_nan_policy.py:531:
UserWarning: scipy.stats.shapiro: For N > 5000, computed p-value may not be
accurate. Current N is 14817.
  res = hypotest_fun_out(*samples, **kwds)
```

[24]:
```
('wilcoxon', 0.0)
```

[25]:
```
compare_paired_samples_from_df(agg_df2, 'actual_time', 'osrm_time',␣
↪test_name="Actual vs OSRM time")
```

```
Data is not Normally distributed

 Actual vs OSRM time:
Used Test: wilcoxon
P-Value: 0.00000
Significant difference - reject null hypothesis.
```

[25]: ('wilcoxon', 0.0)

[26]:
```
compare_paired_samples_from_df(agg_df2, 'actual_time', 'segment_actual_time',␣
↪test_name="Actual vs Segment actual time")
```

```
Data is not Normally distributed

 Actual vs Segment actual time:
Used Test: wilcoxon
P-Value: 0.00000
Significant difference - reject null hypothesis.
```

[26]: ('wilcoxon', 0.0)

[27]:
```
compare_paired_samples_from_df(agg_df2, 'osrm_distance',␣
↪'segment_osrm_distance', test_name="OSRM vs Segment OSRM distance")
```

```
Data is not Normally distributed

 OSRM vs Segment OSRM distance:
Used Test: wilcoxon
P-Value: 0.00000
Significant difference - reject null hypothesis.
```

[27]: ('wilcoxon', 0.0)

[28]:
```
compare_paired_samples_from_df(agg_df2, 'osrm_time', 'segment_osrm_time',␣
↪test_name="OSRM vs Segment OSRM Time")
```

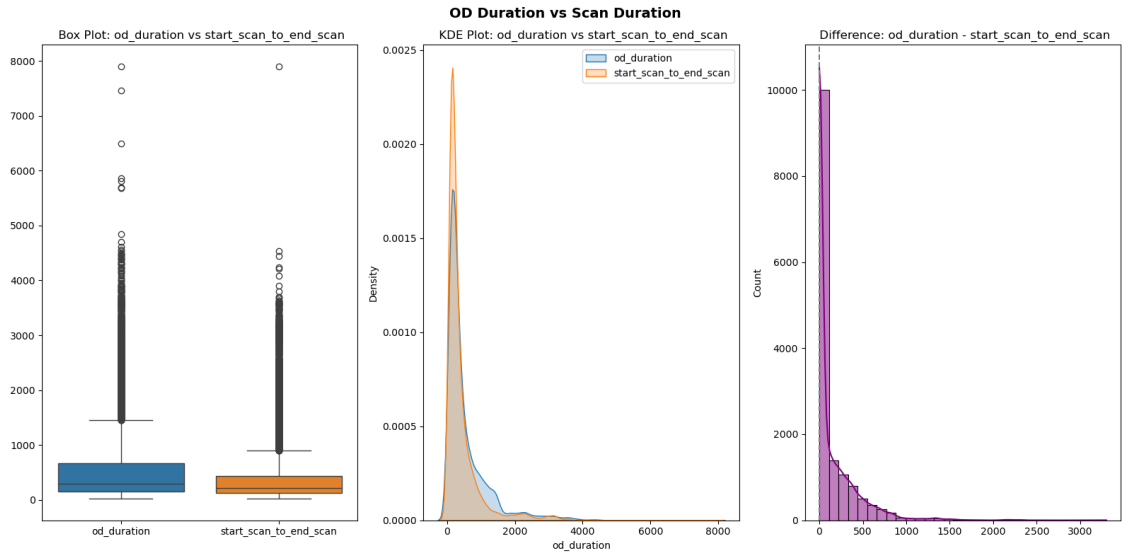
```
Data is not Normally distributed

 OSRM vs Segment OSRM Time:
Used Test: wilcoxon
P-Value: 0.00000
Significant difference - reject null hypothesis.
```

[28]: ('wilcoxon', 0.0)

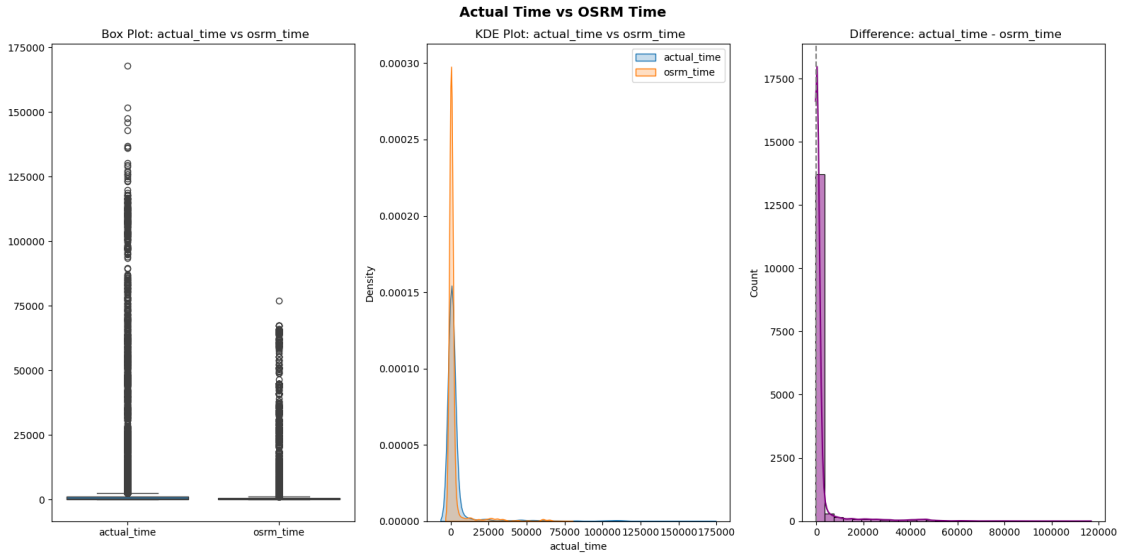
[29]:
```
import matplotlib.pyplot as plt
import seaborn as sns

def plot_paired_distributions(df, col1, col2, title=""):
    """
    Plot box plot, KDE, and difference distribution between two paired columns.
    """
    plt.figure(figsize=(16, 8))

    # Box plot
    plt.subplot(1, 3, 1)
```

```
    sns.boxplot(data=df[[col1, col2]])
    plt.title(f' Box Plot: {col1} vs {col2}')

    # KDE plot
    plt.subplot(1, 3, 2)
    sns.kdeplot(df[col1].dropna(), label=col1, fill=True)
    sns.kdeplot(df[col2].dropna(), label=col2, fill=True)
    plt.title(f' KDE Plot: {col1} vs {col2}')
    plt.legend()

    # Difference distribution
    plt.subplot(1, 3, 3)
    diff = df[col1] - df[col2]
    sns.histplot(diff.dropna(), kde=True, bins=30, color='purple')
    plt.axvline(0, linestyle='--', color='gray')
    plt.title(f' Difference: {col1} - {col2}')

    plt.suptitle(title, fontsize=14, fontweight='bold')
    plt.tight_layout()
    plt.show()
```

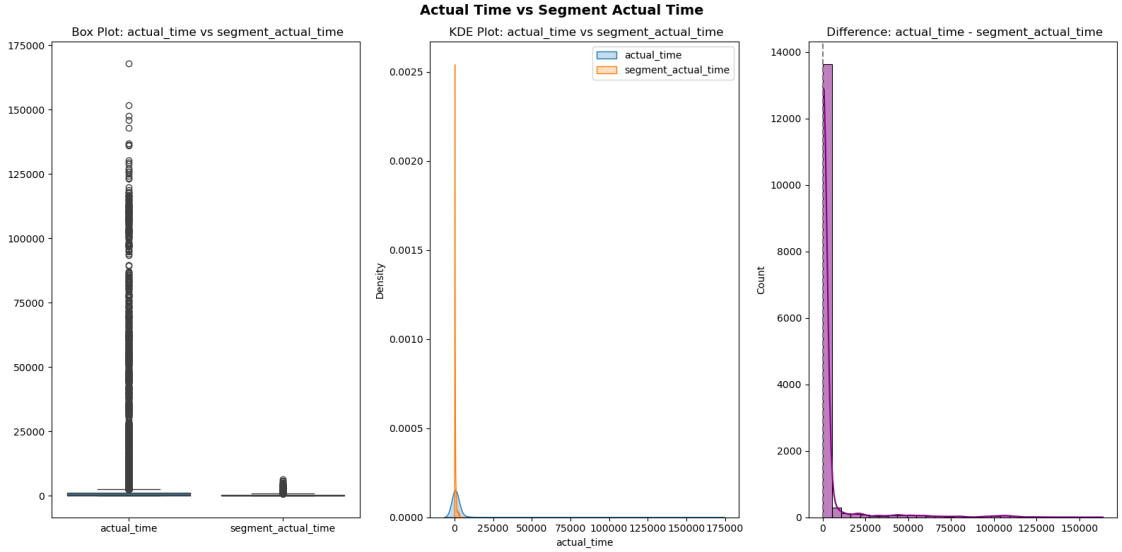
[30]:
```
plot_paired_distributions(agg_df2, 'od_duration', 'start_scan_to_end_scan',␣
↪title="OD Duration vs Scan Duration")
```



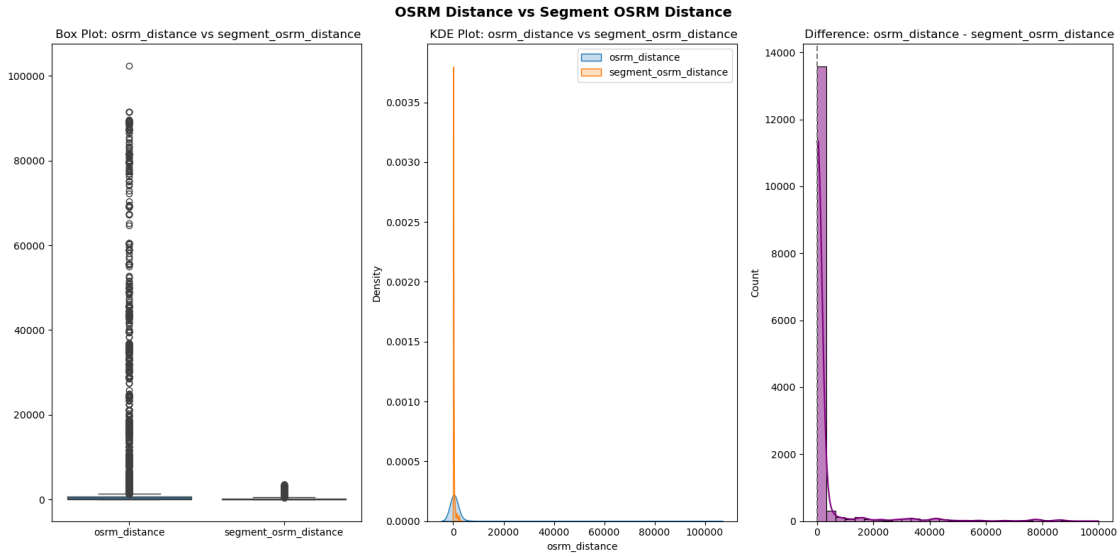
[31]:
```
plot_paired_distributions(agg_df2, 'actual_time', 'osrm_time', title="Actual␣
↪Time vs OSRM Time")
```



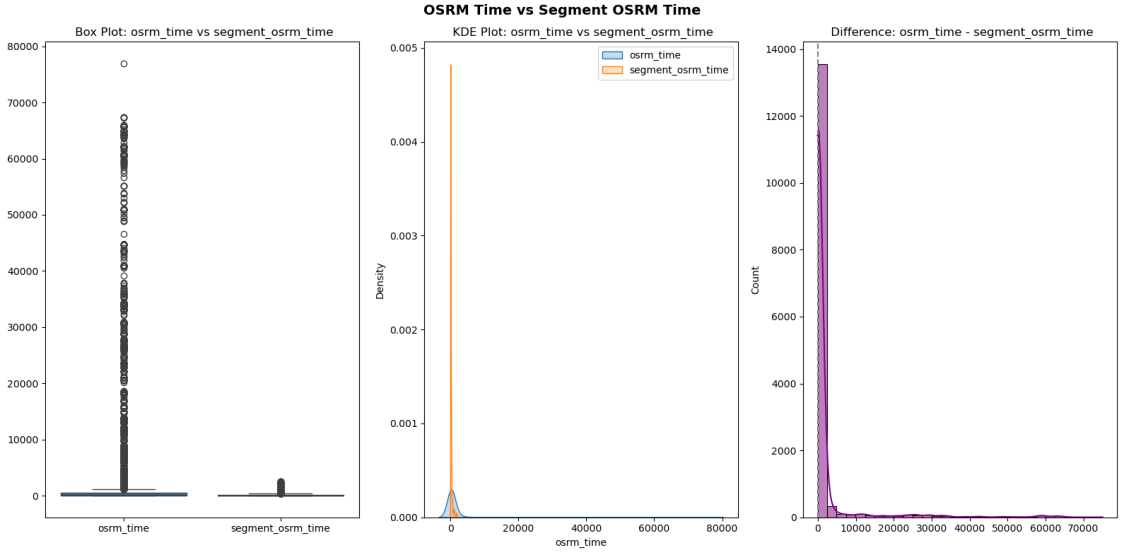
[32]:
```
plot_paired_distributions(agg_df2, 'actual_time', 'segment_actual_time',␣
↪title="Actual Time vs Segment Actual Time")
```



[33]:
```
plot_paired_distributions(agg_df2, 'osrm_distance', 'segment_osrm_distance',␣
↪title="OSRM Distance vs Segment OSRM Distance")
```



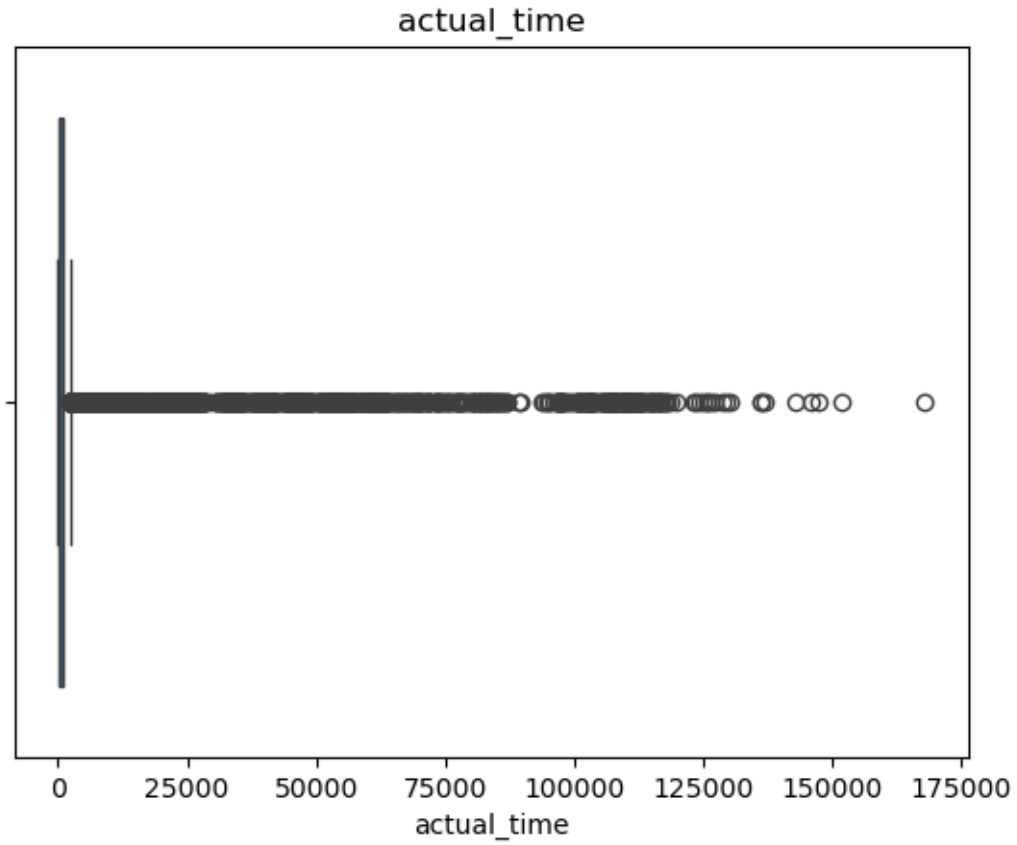
[34]:
```
plot_paired_distributions(agg_df2, 'osrm_time', 'segment_osrm_time',␣
↪title="OSRM Time vs Segment OSRM Time")
```

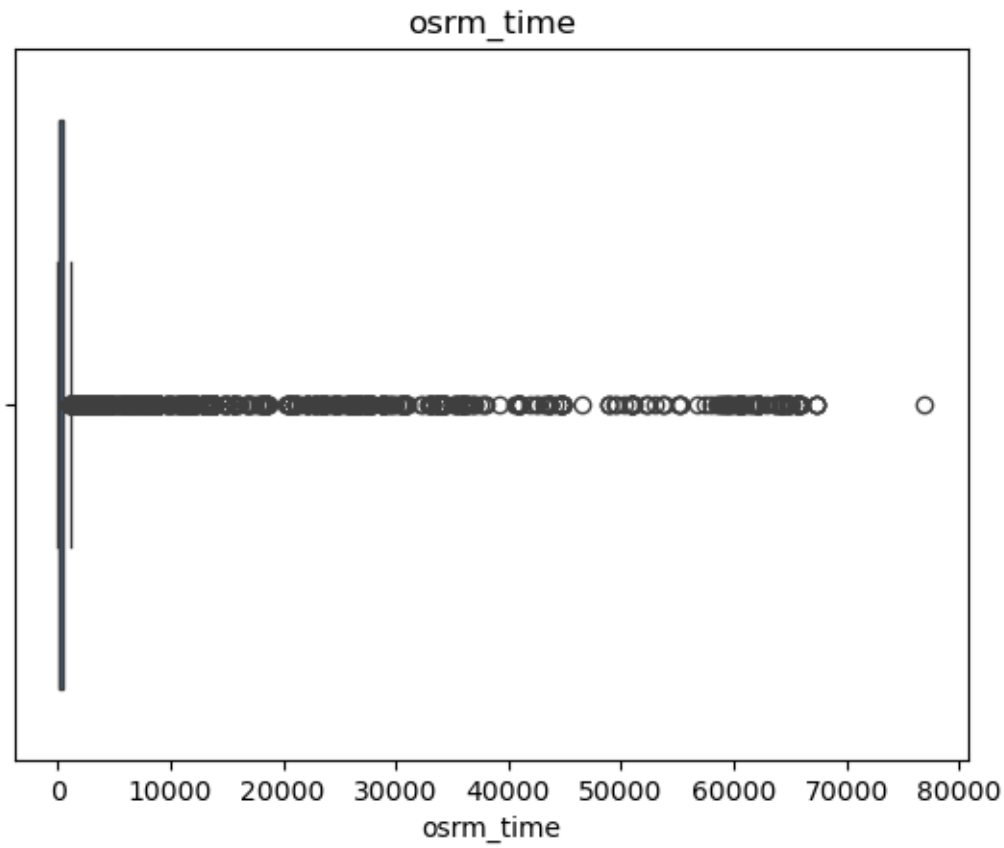


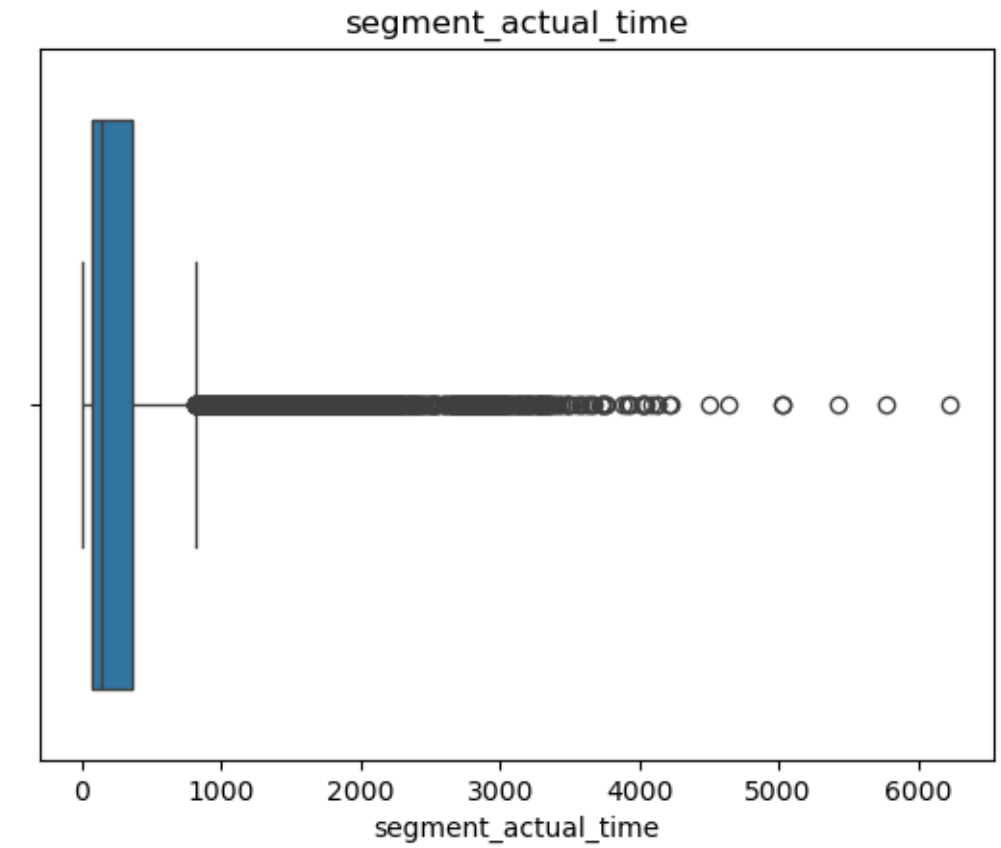
0.6 Outlier Detection and Handling

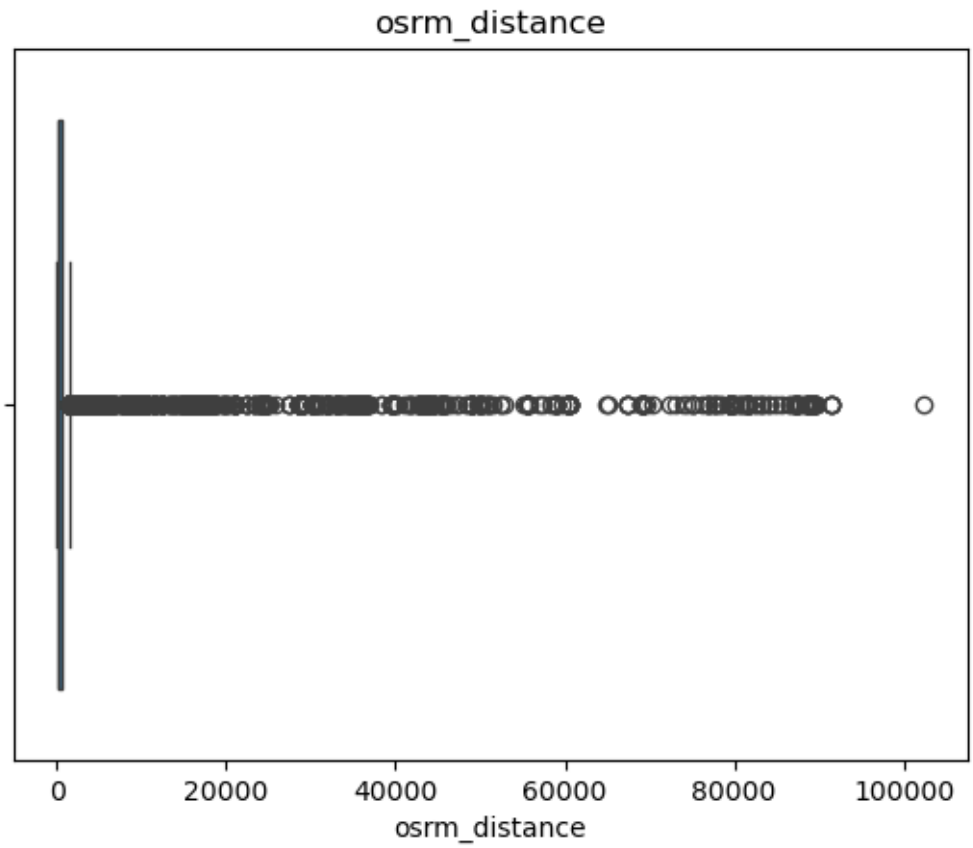
[35]:
```
num_cols = ['actual_time', 'osrm_time', 'segment_actual_time', 'osrm_distance']

for col in num_cols:
    sns.boxplot(x=agg_df2[col])
```

```
    plt.title(col)
    plt.show()
```




osrm_time
0
10000
20000
30000
40000
50000
60000
70000
80000
osrm_time


segment_actual_time
0
1000
2000
3000
4000
5000
6000
segment_actual_time

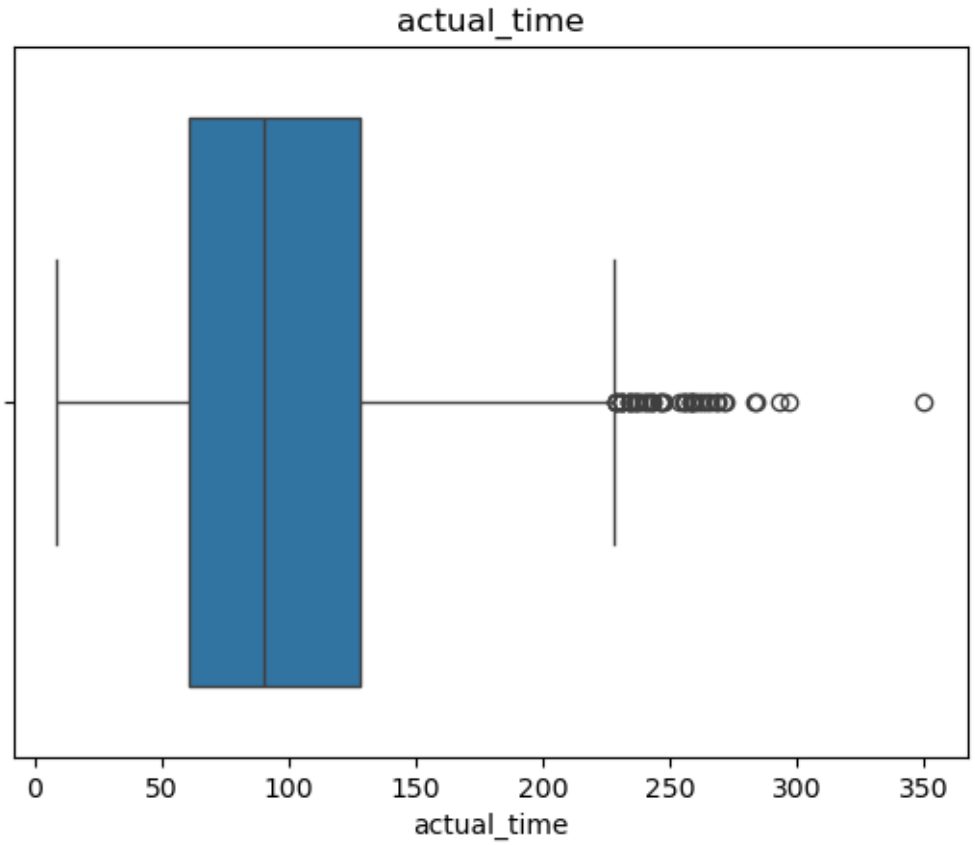


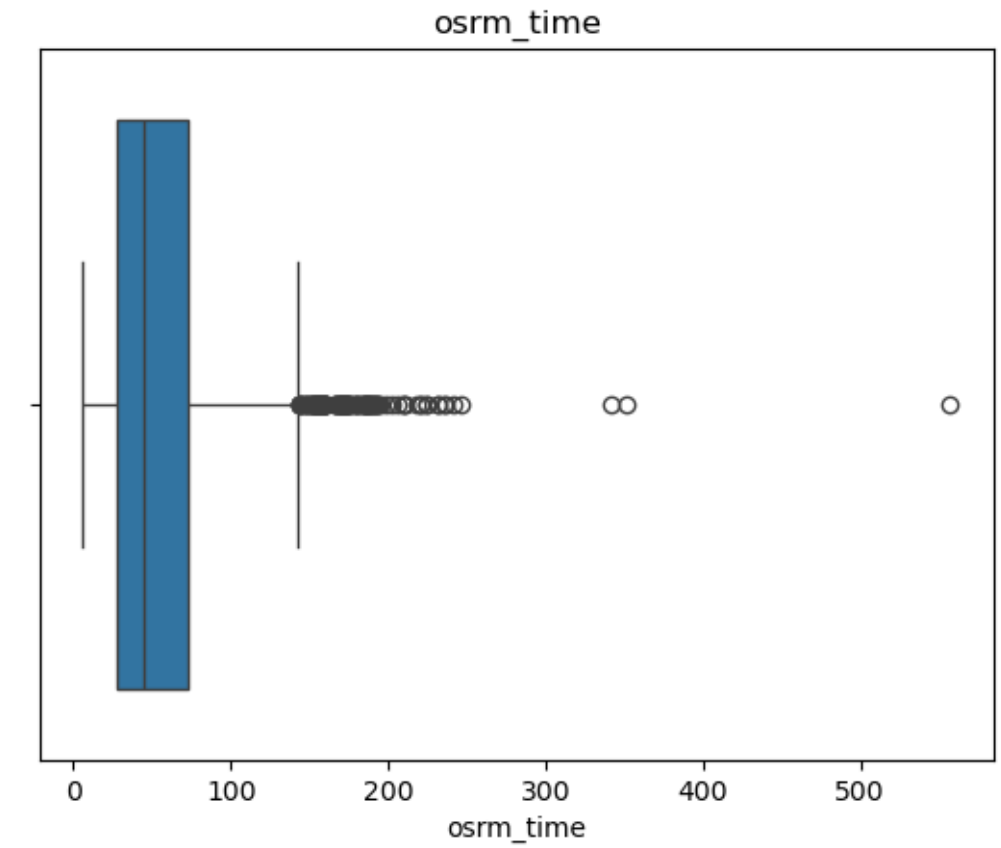
[36]:
```
def iqr_filter(agg_df2, col):
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    return agg_df2[(imputed_df[col] >= Q1 - 1.5 * IQR) & (agg_df2[col] <= Q3 +␣
 ↪1.5 * IQR)]

for col in num_cols:
    agg_df2 = iqr_filter(agg_df2, col)
```

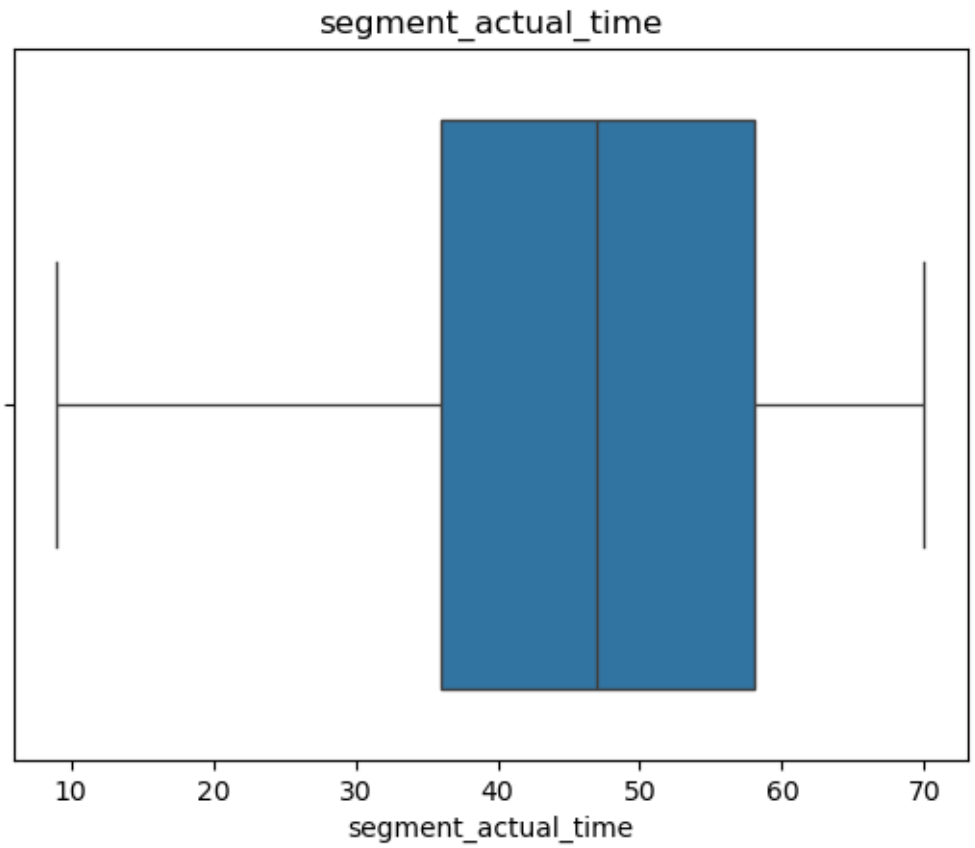
```
C:\Users\kpswe\AppData\Local\Temp\ipykernel_19628\1519824835.py:5: UserWarning:
Boolean Series key will be reindexed to match DataFrame index.
  return agg_df2[(imputed_df[col] >= Q1 - 1.5 * IQR) & (agg_df2[col] <= Q3 + 1.5
* IQR)]
C:\Users\kpswe\AppData\Local\Temp\ipykernel_19628\1519824835.py:5: UserWarning:
Boolean Series key will be reindexed to match DataFrame index.
  return agg_df2[(imputed_df[col] >= Q1 - 1.5 * IQR) & (agg_df2[col] <= Q3 + 1.5
* IQR)]
C:\Users\kpswe\AppData\Local\Temp\ipykernel_19628\1519824835.py:5: UserWarning:
Boolean Series key will be reindexed to match DataFrame index.
```

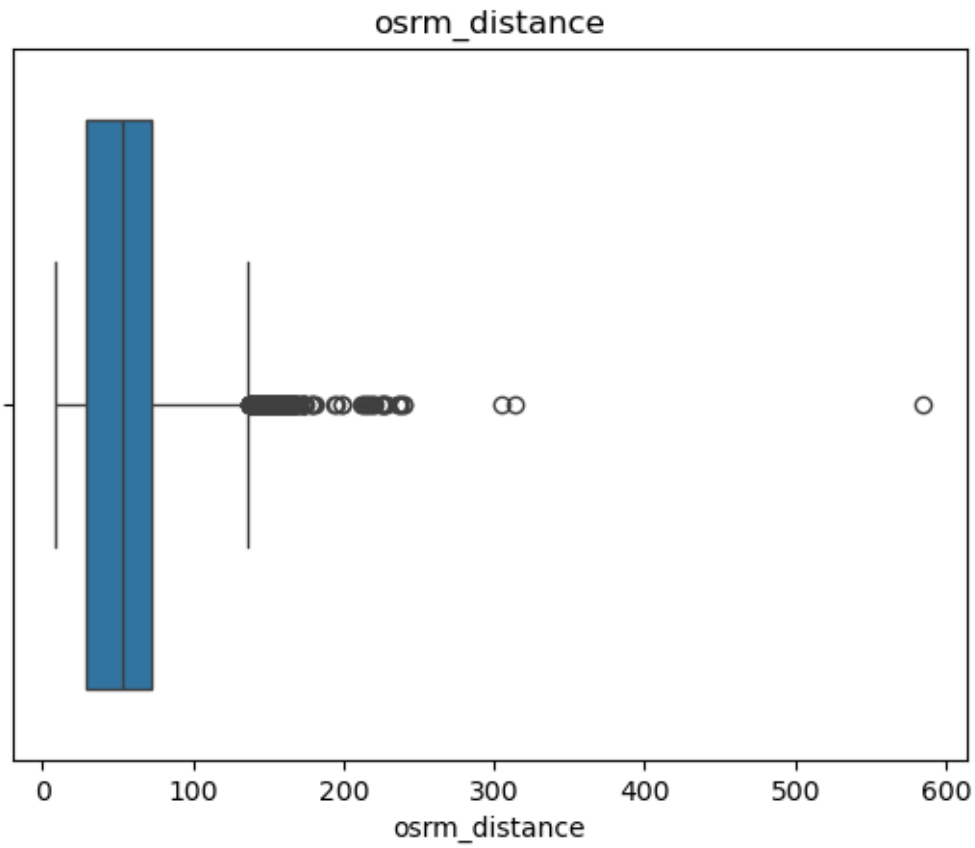
```
  return agg_df2[(imputed_df[col] >= Q1 - 1.5 * IQR) & (agg_df2[col] <= Q3 + 1.5
* IQR)]
C:\Users\kpswe\AppData\Local\Temp\ipykernel_19628\1519824835.py:5: UserWarning:
Boolean Series key will be reindexed to match DataFrame index.
  return agg_df2[(imputed_df[col] >= Q1 - 1.5 * IQR) & (agg_df2[col] <= Q3 + 1.5
* IQR)]
```

[37]:
```
for col in num_cols:
    sns.boxplot(x=agg_df2[col])
    plt.title(col)
    plt.show()
```




osrm_time
0
100
200
300
400
500
osrm_time


segment_actual_time
10
20
30
40
50
60
70
segment_actual_time



0.7 Encoding Categorical Variables

[38]:
```
imputed_df = pd.get_dummies(imputed_df, columns=['route_type',␣
↪'source_name_state', 'destination_name_state'], drop_first=True)
imputed_df.head()
```

[38]:

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

```
  source_center                source_name destination_center  \
0  IND388121AAA  Anand_VUNagar_DC (Gujarat)       IND388620AAB
1  IND388121AAA  Anand_VUNagar_DC (Gujarat)       IND388620AAB
2  IND388121AAA  Anand_VUNagar_DC (Gujarat)       IND388620AAB
3  IND388121AAA  Anand_VUNagar_DC (Gujarat)       IND388620AAB
4  IND388121AAA  Anand_VUNagar_DC (Gujarat)       IND388620AAB

               destination_name              od_start_time  \
0  Khambhat_MotvdDPP_D (Gujarat)  2018-09-20 03:21:32.418600
1  Khambhat_MotvdDPP_D (Gujarat)  2018-09-20 03:21:32.418600
2  Khambhat_MotvdDPP_D (Gujarat)  2018-09-20 03:21:32.418600
3  Khambhat_MotvdDPP_D (Gujarat)  2018-09-20 03:21:32.418600
4  Khambhat_MotvdDPP_D (Gujarat)  2018-09-20 03:21:32.418600

                 od_end_time  …  destination_name_state_Orissa  \
0  2018-09-20 04:47:45.236797  …                         False
1  2018-09-20 04:47:45.236797  …                         False
2  2018-09-20 04:47:45.236797  …                         False
3  2018-09-20 04:47:45.236797  …                         False
4  2018-09-20 04:47:45.236797  …                         False

   destination_name_state_Pondicherry  destination_name_state_Punjab  \
0                               False                          False
1                               False                          False
2                               False                          False
3                               False                          False
4                               False                          False

  destination_name_state_Rajasthan  destination_name_state_Tamil Nadu  \
0                            False                              False
1                            False                              False
2                            False                              False
3                            False                              False
4                            False                              False

   destination_name_state_Telangana  destination_name_state_Tripura  \
0                             False                          False
1                             False                          False
2                             False                          False
3                             False                          False
4                             False                          False

   destination_name_state_Uttar Pradesh  destination_name_state_Uttarakhand  \
0                                 False                               False
1                                 False                               False
2                                 False                               False
```

```
3                                   False                                   False
4                                   False                                   False

   destination_name_state_West Bengal
0                                False
1                                False
2                                False
3                                False
4                                False

[5 rows x 87 columns]
```

0.8 Normalize/ Standardize the numerical features using MinMaxScaler or StandardScaler.

[39]:
```
from sklearn.preprocessing import MinMaxScaler, StandardScaler

scaler = StandardScaler()  # or MinMaxScaler()
scaled_cols = ['actual_time', 'osrm_time', 'osrm_distance', 'od_duration',␣
 ↪'start_scan_to_end_scan']

agg_df2[scaled_cols] = scaler.fit_transform(agg_df2[scaled_cols])
agg_df2.head()
```

[39]:
```
                trip_uuid  actual_time  segment_actual_time  osrm_time  \
3  trip-153671046011330457    -0.322600                 59.0  -0.820316
5  trip-153671055416136166    -0.115434                 60.0  -0.662502
6  trip-153671066201138152    -1.524157                 24.0  -1.109641
7  trip-153671066826362165     0.506061                 64.0   0.258079
9  trip-153671079956500691    -1.296275                 23.0  -1.030734

   segment_osrm_time  osrm_distance  segment_osrm_distance  \
3               16.0      -0.696935                19.8766
5               23.0      -0.527441                28.0647
6               13.0      -1.205045                12.0184
7               34.0      -0.110619                28.9203
9               14.0      -0.999457                16.0860

   start_scan_to_end_scan             od_start_time  \
3               -0.336671 2018-09-12 00:01:00.113710
5               -0.070176 2018-09-12 00:02:34.161600
6               -0.355050 2018-09-12 00:04:22.011653
7               -0.538840 2018-09-12 00:04:28.263977
9               -0.805336 2018-09-12 00:06:39.565253

                 od_end_time  trip_duration_hours  od_duration
3 2018-09-12 01:41:29.809822             1.674916    -0.367897
```

```
5 2018-09-12 03:13:03.432532            3.174797     0.428979
6 2018-09-12 01:42:22.349694            1.633427    -0.389940
7 2018-09-12 03:00:55.163423            2.940805     0.304661
9 2018-09-12 00:55:59.568645            0.822223    -0.820927
```