Business Case: Delhivery - Feature Engineering

```
!wget "https://drive.google.com/uc?export=download&id=1u1DWXZ2ywp-RGVUp1O2ol5lGAM-8SV7D" -O delhivery_data.csv
```

```
--2025-04-07 12:04:04--  https://drive.google.com/uc?export=download&id=1u1DWXZ2ywp-RGVUp1O2ol5lGAM-8SV7D
Resolving drive.google.com (drive.google.com)... 142.250.107.139, 142.250.107.102, 142.250.107.100, ...
Connecting to drive.google.com (drive.google.com)|142.250.107.139|:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://drive.usercontent.google.com/download?id=1u1DWXZ2ywp-RGVUp1O2ol5lGAM-8SV7D&export=download [following]
--2025-04-07 12:04:04--  https://drive.usercontent.google.com/download?id=1u1DWXZ2ywp-RGVUp1O2ol5lGAM-8SV7D&export=download
Resolving drive.usercontent.google.com (drive.usercontent.google.com)... 142.251.188.132, 2607:f8b0:400e:c1b::84
Connecting to drive.usercontent.google.com (drive.usercontent.google.com)|142.251.188.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 55617130 (53M) [application/octet-stream]
Saving to: ‘delhivery_data.csv’

delhivery_data.csv  100%[===================>]  53.04M   210MB/s    in 0.3s

2025-04-07 12:04:08 (210 MB/s) - ‘delhivery_data.csv’ saved [55617130/55617130]
```

1. Introduction

❓ What is Delhivery Business Case Study?

- Delhivery, India's leading and rapidly growing integrated player, has set its sights on creating the commerce operating system. They achieve this by utilizing world-class infrastructure, ensuring the highest quality in logistics operations, and harnessing cutting-edge engineering and technology capabilities.

🎯 Objective:

- The company wants to understand and process the data coming out of data engineering pipelines:
- Clean, sanitize and manipulate data to get useful features out of raw fields
- Make sense out of the raw data and help the data science team to build forecasting models on it

📚 About Data:

📃 Features of the dataset:

1) data - tells whether the data is testing or training data

2) trip_creation_time – Timestamp of trip creation

3) route_schedule_uuid – Unique Id for a particular route schedule route_type – Transportation type

- FTL – Full Truck Load: FTL shipments get to the destination sooner, as the truck is making no other pickups or drop-offs along the way
- Carting: Handling system consisting of small vehicles (carts)

4) trip_uuid - Unique ID given to a particular trip (A trip may include different source and destination centers)

5) source_center - Source ID of trip origin

6) source_name - Source Name of trip origin

7) destination_cente – Destination ID

8) destination_name – Destination Name

9) od_start_time – Trip start time

10) od_end_time – Trip end time

11) start_scan_to_end_scan – Time taken to deliver from source to destination

12) is_cutoff – Unknown field

13) cutoff_factor – Unknown field

14) cutoff_timestamp – Unknown field

15) actual_distance_to_destination – Distance in Kms between source and destination warehouse

16) actual_time – Actual time taken to complete the delivery (Cumulative)

17) osrm_time – An open-source routing engine time calculator which computes the shortest path between points in a given map (Includes usual traffic, distance through major and minor roads) and gives the time (Cumulative)

18) osrm_distance – An open-source routing engine which computes the shortest path between points in a given map (Includes usual traffic, distance through major and minor roads) (Cumulative)

19) factor – Unknown field

20) segment_actual_time – This is a segment time. Time taken by the subset of the package delivery

21) segment_osrm_time – This is the OSRM segment time. Time taken by the subset of the package delivery

22) segment_osrm_distance – This is the OSRM distance. Distance covered by subset of the package delivery

23) segment_factor – Unknown field

2.Exploratory Data Analysis

```
#importing libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import missingno as msno
import warnings
warnings.filterwarnings('ignore')
from scipy import stats
from sklearn.impute import SimpleImputer
from scipy.stats import pearsonr
from sklearn.preprocessing import OneHotEncoder,MinMaxScaler,StandardScaler
```

```
# loading the dataset
df = pd.read_csv('delhivery_data.csv')
```

```
#to view full data
df
```

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
...	...	...	...	...	...	...	...	...	
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144864	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144865	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144866	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	

144867 rows × 24 columns

```
#to view columns

df.columns
#df.keys()== df.columns
```

```
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')
```

```
#view first 5 rows/records
df.head(5)
#view first 5 rows/records default=5
#df.head()
```

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	des
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambh
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambh
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambh
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambh
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambh

5 rows × 24 columns

```
#view last 5 rows/records,deafault=5
df.tail()
#view last 5 rows/records
#df.tail(5)
```

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	des
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurga
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurga
144864	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurga
144865	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurga
144866	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurga

5 rows × 24 columns

```
#To get index of dataframe
df.index
```

```
RangeIndex(start=0, stop=144867, step=1)
```

```
#To get shape information

df.shape

#10886 rows and 12 columns
```

```
(144867, 24)
```

```
# to get dimensional detail of dataframe

df.ndim

#2D
```

```
2
```

```
#Datatype

print(df.dtypes)
```

```
data                          object
trip_creation_time            object
route_schedule_uuid           object
route_type                    object
trip_uuid                     object
source_center                 object
source_name                   object
destination_center            object
destination_name              object
od_start_time                 object
```

```
od_end_time                        object
start_scan_to_end_scan            float64
is_cutoff                            bool
cutoff_factor                       int64
cutoff_timestamp                   object
actual_distance_to_destination    float64
actual_time                       float64
osrm_time                         float64
osrm_distance                     float64
factor                            float64
segment_actual_time               float64
segment_osrm_time                 float64
segment_osrm_distance             float64
segment_factor                    float64
dtype: object
```

```
# Convert columns to categorical types and datetime to Datetime
df['data'] = df['data'].astype('category')
df['route_type'] = df['route_type'].astype('category')
df['is_cutoff'] = df['is_cutoff'].astype('category')
df['trip_creation_time']=pd.to_datetime(df['trip_creation_time'])
df['od_start_time']=pd.to_datetime(df['od_start_time'])
df['od_end_time']=pd.to_datetime(df['od_end_time'])
# Convert column to datetime and round to milliseconds
df['cutoff_timestamp'] = pd.to_datetime(df['cutoff_timestamp'], errors='coerce')
df['cutoff_timestamp'] = df['cutoff_timestamp'].dt.strftime('%Y-%m-%d %H:%M:%S.%f').str[:-3]  # Keeping milliseconds
df['cutoff_timestamp'] = pd.to_datetime(df['cutoff_timestamp'], errors='coerce')
df['actual_time']=pd.to_datetime(df['actual_time'])
df['osrm_time']=pd.to_datetime(df['osrm_time'])
df['segment_actual_time']=pd.to_datetime(df['segment_actual_time'])
df['segment_osrm_time']=pd.to_datetime(df['segment_osrm_time'])

##trip_creation_time,od_start_time,od_end_time,cutoff_timestamp,actual_time,osrm_time,segment_actual_time,segment_osrm_time,


#Datatype

print(df.dtypes)
```

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

```
# to get complete information of each column of dataframe like counts,datatype,memory usage.
#Note: For missing value in each column data type will be object

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  category
 1   trip_creation_time              144867 non-null  datetime64[ns]
 2   route_schedule_uuid             144867 non-null  object
 3   route_type                      144867 non-null  category
 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  datetime64[ns]
 10  od_end_time                     144867 non-null  datetime64[ns]
 11  start_scan_to_end_scan          144867 non-null  float64
 12  is_cutoff                       144867 non-null  category
 13  cutoff_factor                   144867 non-null  int64
 14  cutoff_timestamp                141438 non-null  datetime64[ns]
 15  actual_distance_to_destination  144867 non-null  float64
 16  actual_time                     144867 non-null  datetime64[ns]
 17  osrm_time                       144867 non-null  datetime64[ns]
```

```
 18  osrm_distance                  144867 non-null  float64
 19  factor                         144867 non-null  float64
 20  segment_actual_time            144867 non-null  datetime64[ns]
 21  segment_osrm_time              144867 non-null  datetime64[ns]
 22  segment_osrm_distance          144867 non-null  float64
 23  segment_factor                 144867 non-null  float64
dtypes: category(3), datetime64[ns](8), float64(6), int64(1), object(6)
memory usage: 23.6+ MB
```

Insights

From the above details it is clear that given dataframe is of dimension 2D with 144867 rows and 24 columns.

Also we can also observe that there are missing values for few columns like source_name,destination_name and cutoff_timestamp .

Statistical Summary

```
#for column with datatype as int, df.describe() will give statistical information like count,mean,min,max,std detail for that column.
```

```
df.describe()
```

	trip_creation_time	od_start_time	od_end_time	start_scan_to_end_scan	cutoff_factor	cutoff_timestamp	actual_distance_to_desti
count	144867	144867	144867	144867.000000	144867.000000	141438	144867.
mean	2018-09-22 13:34:23.659819264	2018-09-22 18:02:45.855230720	2018-09-23 10:04:31.395393024	961.262986	232.926567	2018-09-23 03:43:41.794807552	234.
min	2018-09-12 00:00:16.535741	2018-09-12 00:00:16.535741	2018-09-12 00:50:10.814399	20.000000	9.000000	2018-09-12 00:10:27	9.
25%	2018-09-17 03:20:51.775845888	2018-09-17 08:05:40.886155008	2018-09-18 01:48:06.410121984	161.000000	22.000000	2018-09-17 19:52:04.750000128	23.
50%	2018-09-22 04:24:27.932764928	2018-09-22 08:53:00.116656128	2018-09-23 03:13:03.520212992	449.000000	66.000000	2018-09-22 22:02:55	66.
75%	2018-09-27 17:57:56.350054912	2018-09-27 22:41:50.285857024	2018-09-28 12:49:06.054018048	1634.000000	286.000000	2018-09-28 06:37:55	286.
max	2018-10-03 23:59:42.701692	2018-10-06 04:27:23.392375	2018-10-08 03:00:24.353479	7898.000000	1927.000000	2018-10-06 23:44:12	1927.
std	NaN	NaN	NaN	1037.012769	344.755577	NaN	344.

```
#Statistical Summary: Generate a statistical summary of the dataset.
```

```
df.describe(include='all')
```

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center
count	144867	144867	144867	144867	144867	144867	144574	144867
unique	2	NaN	1504	2	14817	1508	1498	1481
top	training	NaN	thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...	FTL	trip-153759210483476123	IND000000ACB	Gurgaon_Bilaspur_HB (Haryana)	IND000000ACB
freq	104858	NaN	1812	99660	101	23347	23347	15192
mean	NaN	2018-09-22 13:34:23.659819264	NaN	NaN	NaN	NaN	NaN	NaN
min	NaN	2018-09-12 00:00:16.535741	NaN	NaN	NaN	NaN	NaN	NaN
25%	NaN	2018-09-17 03:20:51.775845888	NaN	NaN	NaN	NaN	NaN	NaN
50%	NaN	2018-09-22 04:24:27.932764928	NaN	NaN	NaN	NaN	NaN	NaN
75%	NaN	2018-09-27 17:57:56.350054912	NaN	NaN	NaN	NaN	NaN	NaN
max	NaN	2018-10-03 23:59:42.701692	NaN	NaN	NaN	NaN	NaN	NaN
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

11 rows × 24 columns

Insights:

- data: Indicates data purpose (training or others), with the majority as training (104,858 rows).

- trip_creation_time: The timestamp of trip creation, spanning September 12 to October 3, 2018. Useful for temporal trend analysis.
- route_schedule_uuid: Unique route schedule identifiers (1504 unique values), showcasing diverse routing configurations.
- route_type: Categorical column (e.g., FTL), with FTL dominating 99,660 rows, pointing to prevalent freight type.
- trip_uuid: Unique trip IDs (14,817 unique), essential for individual trip tracking and analysis.
- source_center / destination_center: Encoded source and destination hub IDs, indicating traffic between various hub pairs.
- source_name / destination_name: Human-readable hub names, with recurring hubs like "Gurgaon_Bilaspur_HB (Haryana)" frequently appearing.
- od_start_time / od_end_time: Captures operational duration, with the median trip starting on September 22, 2018, at 08:53.
- cutoff_timestamp: Timestamp for cutoffs; spans roughly the same period as the trip lifecycle.
- actual_distance_to_destination: Average trip distance is 234 km, with high variability (standard deviation of 345 km).
- actual_time / osrm_time: Timestamps of actual vs. predicted travel times, often defaulting to 1970-01-01, indicating data anomalies.
- osrm_distance: Predicted trip distances average 284 km, marginally higher than actuals, useful for route optimization.
- factor: Efficiency ratio; averages around 2.12. Negative outliers suggest potential data issues.
- segment_actual_time / segment_osrm_time: Times for route segments, granularity useful for segment-level optimization.
- segment_osrm_distance: Predicted segment distances (mean of 22.8 km), aiding performance metrics.
- segment_factor: Segment efficiency ratio; median is ~1.68, but high variability and negative values might require deeper investigation.

Noteworthy Observations:

1) Categorical Data:

- The data column has two categories (e.g., training, testing), with the majority being training.
- Columns like route_type and trip_uuid have a wide variety of unique values (1504 for route_schedule_uuid and 14817 for trip_uuid).

2) Datetime Columns:

- trip_creation_time, od_start_time, and other datetime fields range from September 12, 2018, to October 6, 2018.
- Mean values for datetime columns indicate the approximate middle range during this period.

3) Distance and Factors:

- actual_distance_to_destination and osrm_distance indicate a mean of 234 km and 284 km respectively, with wide variability (standard deviation around 345 km for actual_distance_to_destination).
- factor and segment_factor measure ratios and seem to average around 2.12 and 2.22 respectively.

4) Potential Anomalies:

- Negative values appear in segment_factor (minimum value is -23.444), which could require attention.
- Some datetime-related columns like actual_time and osrm_time are unusually close to 1970-01-01, suggesting incorrect or missing data entries.

👥 Duplicate Detection

```
df.duplicated().value_counts()
```

	count
False	144867

dtype: int64

🔍 Insights There are no duplicate entries in the dataset

🕵️ Missing Value Analysis

```
missing_values=df.isnull().sum()
missing_values
```

	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	3429
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

Insights:

- There are missing values for few columns like source_name,destination_name and cutoff_timestamp .

```
from sklearn.impute import SimpleImputer

# Handling missing values for categorical columns (source_name and destination_name)
imputer_constant = SimpleImputer(strategy='constant', fill_value='Unknown')  # Define imputer with constant value
df[['source_name']] = imputer_constant.fit_transform(df[['source_name']])  # Use constant to replace missing values
df[['destination_name']] = imputer_constant.fit_transform(df[['destination_name']])
```

```
# Using mean to fill missing datetime values (convert datetime to numerical for imputation)
df[['cutoff_timestamp']] =SimpleImputer(strategy='mean').fit_transform(df[['cutoff_timestamp']])
```

```
missing_values=df.isnull().sum()
missing_values
```

	0
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

Insight:

- Missing values handled.

```
df['cutoff_timestamp'] = pd.to_datetime(df['cutoff_timestamp'], errors='coerce')
df.dtypes
```

	0
data	category
trip_creation_time	datetime64[ns]
route_schedule_uuid	object
route_type	category
trip_uuid	object
source_center	object
source_name	object
destination_center	object
destination_name	object
od_start_time	datetime64[ns]
od_end_time	datetime64[ns]
start_scan_to_end_scan	float64
is_cutoff	category
cutoff_factor	int64
cutoff_timestamp	datetime64[ns]
actual_distance_to_destination	float64
actual_time	datetime64[ns]
osrm_time	datetime64[ns]
osrm_distance	float64
factor	float64
segment_actual_time	datetime64[ns]
segment_osrm_time	datetime64[ns]
segment_osrm_distance	float64
segment_factor	float64

dtype: object

📍 For Non-graphical Analysis:

✅ Sanity Check for columns

```
# checking the unique values for columns
for i in df.columns:
  print('Unique Values in',i,'column are :-')
  print(df[i].unique())
  print('-'*70)
```

```
Length: 747, dtype: datetime64[ns]
----------------------------------------------------------------------
Unique Values in segment_osrm_time column are :-
<DatetimeArray>
['1970-01-01 00:00:00.000000011', '1970-01-01 00:00:00.000000009',
 '1970-01-01 00:00:00.000000007', '1970-01-01 00:00:00.000000012',
 '1970-01-01 00:00:00.000000005', '1970-01-01 00:00:00.000000006',
 '1970-01-01 00:00:00.000000010', '1970-01-01 00:00:00.000000024',
 '1970-01-01 00:00:00.000000027', '1970-01-01 00:00:00.000000026',
 ...
 '1970-01-01 00:00:00.000000185', '1970-01-01 00:00:00.000000158',
 '1970-01-01 00:00:00.000000324', '1970-01-01 00:00:00.000000177',
 '1970-01-01 00:00:00.000000453', '1970-01-01 00:00:00.000000152',
 '1970-01-01 00:00:00.000000176', '1970-01-01 00:00:00.000000737',
 '1970-01-01 00:00:00.000000173', '1970-01-01 00:00:00.000001032']
Length: 214, dtype: datetime64[ns]
----------------------------------------------------------------------
Unique Values in segment_osrm_distance column are :-
[11.9653  9.759  10.8152 ... 20.7053 18.8885  8.8088]
----------------------------------------------------------------------
Unique Values in segment_factor column are :-
[ 1.27272727  1.11111111  2.28571429 ...  2.09322034  5.72
 29.77777778]
----------------------------------------------------------------------
```

```
for i in df.columns:
  print('Value count in',i,'column are :-')
  print(df[i].value_counts())
  print('-'*70)
```

```
1970-01-01 00:00:00.000000024    6188
1970-01-01 00:00:00.000000026    5479
1970-01-01 00:00:00.000000030    4903
1970-01-01 00:00:00.000000027    4439
1970-01-01 00:00:00.000000023    4401
                                 ...
1970-01-01 00:00:00.000000488       1
1970-01-01 00:00:00.000000748       1
1970-01-01 00:00:00.000001020       1
1970-01-01 00:00:00.000001677       1
1970-01-01 00:00:00.000000830       1
Name: count, Length: 747, dtype: int64
----------------------------------------------------------------------
Value count in segment_osrm_time column are :-
segment_osrm_time
1970-01-01 00:00:00.000000016    11483
1970-01-01 00:00:00.000000017    10856
1970-01-01 00:00:00.000000018     8734
1970-01-01 00:00:00.000000019     6925
1970-01-01 00:00:00.000000015     6846
                                 ...
1970-01-01 00:00:00.000000152        1
1970-01-01 00:00:00.000000176        1
1970-01-01 00:00:00.000000737        1
1970-01-01 00:00:00.000000173        1
1970-01-01 00:00:00.000001032        1
Name: count, Length: 214, dtype: int64
----------------------------------------------------------------------
Value count in segment_osrm_distance column are :-
segment_osrm_distance
0.0000     1536
22.6267       8
25.6081       8
24.4061       7
26.6974       7
           ...
20.7053       1
17.3725       1
16.3094       1
23.6371       1
70.0436       1
Name: count, Length: 113799, dtype: int64
----------------------------------------------------------------------
Value count in segment_factor column are :-
segment_factor
 2.000000     6001
 1.500000     4637
 1.000000     2371
 1.666667     2370
-1.000000     2347
               ...
 4.730769        1
 0.228261        1
 13.111111       1
 26.428571       1
 3.756757        1
Name: count, Length: 5675, dtype: int64
----------------------------------------------------------------------
```

Insights: Here are very brief insights for each column:

- Data: 2 unique values (training, test).
- Trip Creation Time: 14,817 unique datetime values.

- Route Schedule UUID: 1,504 unique values.
- Route Type: 2 unique values (Carting, FTL).
- Trip UUID: 14,817 unique identifiers.
- Source Center: 1,508 unique values.
- Source Name: 1,498 unique human-readable hub names.
- Destination Center: 1,481 unique values.
- Destination Name: 1,468 unique human-readable hub names.
- OD Start Time: 26,369 unique datetime values.
- OD End Time: 26,369 unique datetime values.
- Start Scan to End Scan: 2,476 unique numeric durations.
- Is Cutoff: 2 unique values (True, False).
- Cutoff Factor: Highly variable numeric factors with many unique values.
- Cutoff Timestamp: 3,182 unique datetime values.
- Actual Distance to Destination: 144,867 unique distances.
- Actual Time: 3,182 unique datetime values, some anomalies.
- OSRM Time: 1,531 unique predicted timestamps.
- OSRM Distance: 144,867 unique predicted distances.
- Factor: Highly variable efficiency ratios.
- Segment Actual Time: 747 unique datetime values.
- Segment OSRM Time: 214 unique predicted datetime values.
- Segment OSRM Distance: 144,867 unique segment distances.
- Segment Factor: Highly variable segment efficiency ratios.

Business Insights based on Non-Graphical and Visual Analysis

```
# Comments on the range of attributes
print("Comments on the range of attributes:")
for column in df.columns:
    if pd.api.types.is_categorical_dtype(df[column]):
        df[column] = df[column].astype('category').cat.as_ordered()

    print(f"{column}: {df[column].min()} to {df[column].max()}")
```

```
Comments on the range of attributes:
data: test to training
trip_creation_time: 2018-09-12 00:00:16.535741 to 2018-10-03 23:59:42.701692
route_schedule_uuid: thanos::sroute:0007affd-fd01-4cf0-8a4f-90419df059f7 to thanos::sroute:fffa2622-a170-4d08-b60b-38dfbae83869
route_type: Carting to FTL
trip_uuid: trip-153671041653548748 to trip-153861118270144424
source_center: IND000000AAL to IND854335AAA
source_name: AMD_Memnagar (Gujarat) to Zahirabad_Mohim_D (Telangana)
destination_center: IND000000AAL to IND854335AAA
destination_name: AMD_Memnagar (Gujarat) to Zirakpur_DC (Punjab)
od_start_time: 2018-09-12 00:00:16.535741 to 2018-10-06 04:27:23.392375
od_end_time: 2018-09-12 00:50:10.814399 to 2018-10-08 03:00:24.353479
start_scan_to_end_scan: 20.0 to 7898.0
is_cutoff: False to True
cutoff_factor: 9 to 1927
cutoff_timestamp: 2018-09-12 00:10:27 to 2018-10-06 23:44:12
actual_distance_to_destination: 9.00004535977208 to 1927.4477046975032
actual_time: 1970-01-01 00:00:00.000000009 to 1970-01-01 00:00:00.000004532
osrm_time: 1970-01-01 00:00:00.000000006 to 1970-01-01 00:00:00.000001686
osrm_distance: 9.0082 to 2326.1991000000003
factor: 0.144 to 77.38709677419355
segment_actual_time: 1969-12-31 23:59:59.999999756 to 1970-01-01 00:00:00.000003051
segment_osrm_time: 1970-01-01 00:00:00 to 1970-01-01 00:00:00.000001611
segment_osrm_distance: 0.0 to 2191.4037000000003
segment_factor: -23.444444444444443 to 574.25
```

🔍 Comments on the range of attributes:

Data: Ranges from test to training.

Trip Creation Time: 2018-09-12 00:00:16.535741 to 2018-10-03 23:59:42.701692.

Route Schedule UUID: thanos::sroute:0007affd-fd01-4cf0-8a4f-90419df059f7 to thanos::sroute:fffa2622-a170-4d08-b60b-38dfbae83869.

Route Type: Ranges from Carting to FTL.

Trip UUID: trip-153671041653548748 to trip-153861118270144424.

Source Center: IND000000AAL to IND854335AAA.

Source Name: AMD_Memnagar (Gujarat) to Zahirabad_Mohim_D (Telangana).

Destination Center: IND000000AAL to IND854335AAA.

Destination Name: AMD_Memnagar (Gujarat) to Zirakpur_DC (Punjab).

OD Start Time: 2018-09-12 00:00:16.535741 to 2018-10-06 04:27:23.392375.

OD End Time: 2018-09-12 00:50:10.814399 to 2018-10-08 03:00:24.353479.

Start Scan to End Scan: 20.0 to 7898.0.

Is Cutoff: False to True.

Cutoff Factor: Ranges from 9 to 1927.

Cutoff Timestamp: 2018-09-12 00:10:27 to 2018-10-06 23:44:12

Actual Distance to Destination: 9.00004535977208 to 1927.4477046975032.

Actual Time: 1970-01-01 00:00:00.000000009 to 1970-01-01 00:00:00.000004532.

OSRM Time: 1970-01-01 00:00:00.000000006 to 1970-01-01 00:00:00.000001686.

OSRM Distance: 9.0082 to 2326.1991000000003.

Factor: 0.144 to 77.38709677419355.

Segment Actual Time: 1969-12-31 23:59:59.999999756 to 1970-01-01 00:00:00.000003051.

Segment OSRM Time: 1970-01-01 00:00:00 to 1970-01-01 00:00:00.000001611.

Segment OSRM Distance: 0.0 to 2191.4037000000003.

Segment Factor: -23.444444444444443 to 574.25.

```
# Comments on the distribution of the variables and relationship between them
def comments_on_distribution(df):
    comments = []

    for column in df.columns:
        comments.append(f"{column}:")
        if pd.api.types.is_numeric_dtype(df[column]):
            comments.append(f" - Distribution: The {column} distribution spans from {df[column].min()} to {df[column].max()}.")
        else:
            comments.append(f" - Distribution: The {column} distribution includes categories: {df[column].unique().tolist()}.")

        # Relationship with other variables
        comments.append(" - Relationship with Other Variables:")
        for other_column in df.columns:
            if column != other_column:
                comments.append(f"   - {other_column}: Relationship analysis between {column} and {other_column}.")

        comments.append("\n")

    return "\n".join(comments)

# Print the comments on distribution and relationships
print(comments_on_distribution(df))
```

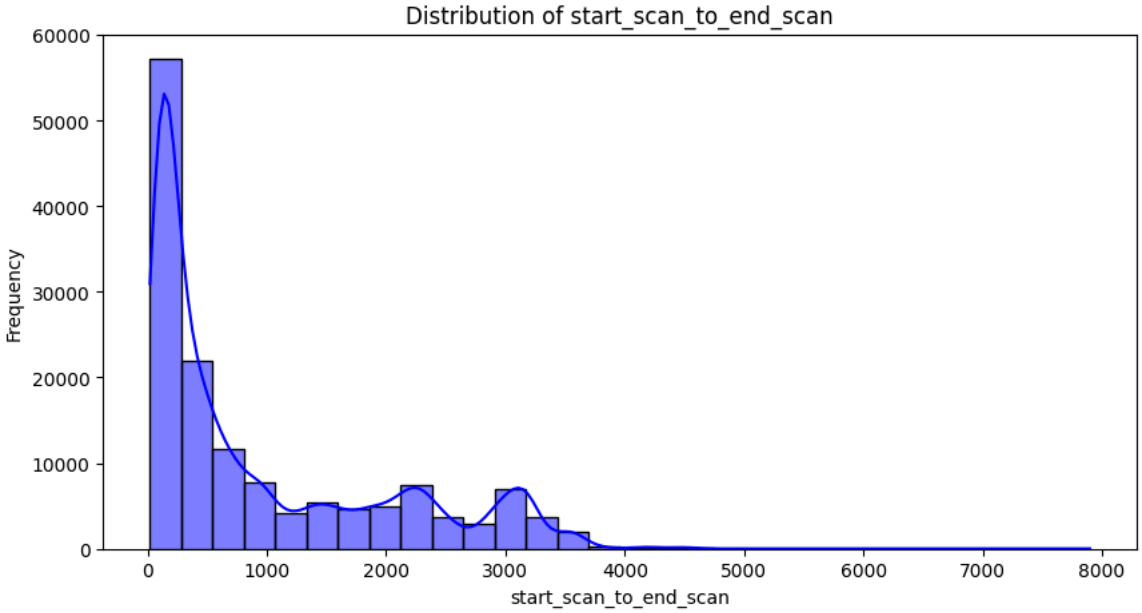
```
        - data: Relationship analysis between segment_factor and data.
        - trip_creation_time: Relationship analysis between segment_factor and trip_creation_time.
        - route_schedule_uuid: Relationship analysis between segment_factor and route_schedule_uuid.
        - route_type: Relationship analysis between segment_factor and route_type.
        - trip_uuid: Relationship analysis between segment_factor and trip_uuid.
        - source_center: Relationship analysis between segment_factor and source_center.
        - source_name: Relationship analysis between segment_factor and source_name.
        - destination_center: Relationship analysis between segment_factor and destination_center.
        - destination_name: Relationship analysis between segment_factor and destination_name.
        - od_start_time: Relationship analysis between segment_factor and od_start_time.
        - od_end_time: Relationship analysis between segment_factor and od_end_time.
        - start_scan_to_end_scan: Relationship analysis between segment_factor and start_scan_to_end_scan.
        - is_cutoff: Relationship analysis between segment_factor and is_cutoff.
        - cutoff_factor: Relationship analysis between segment_factor and cutoff_factor.
        - cutoff_timestamp: Relationship analysis between segment_factor and cutoff_timestamp.
        - actual_distance_to_destination: Relationship analysis between segment_factor and actual_distance_to_destination.
        - actual_time: Relationship analysis between segment_factor and actual_time.
        - osrm_time: Relationship analysis between segment_factor and osrm_time.
        - osrm_distance: Relationship analysis between segment_factor and osrm_distance.
        - factor: Relationship analysis between segment_factor and factor.
        - segment_actual_time: Relationship analysis between segment_factor and segment_actual_time.
        - segment_osrm_time: Relationship analysis between segment_factor and segment_osrm_time.
        - segment_osrm_distance: Relationship analysis between segment_factor and segment_osrm_distance.
```

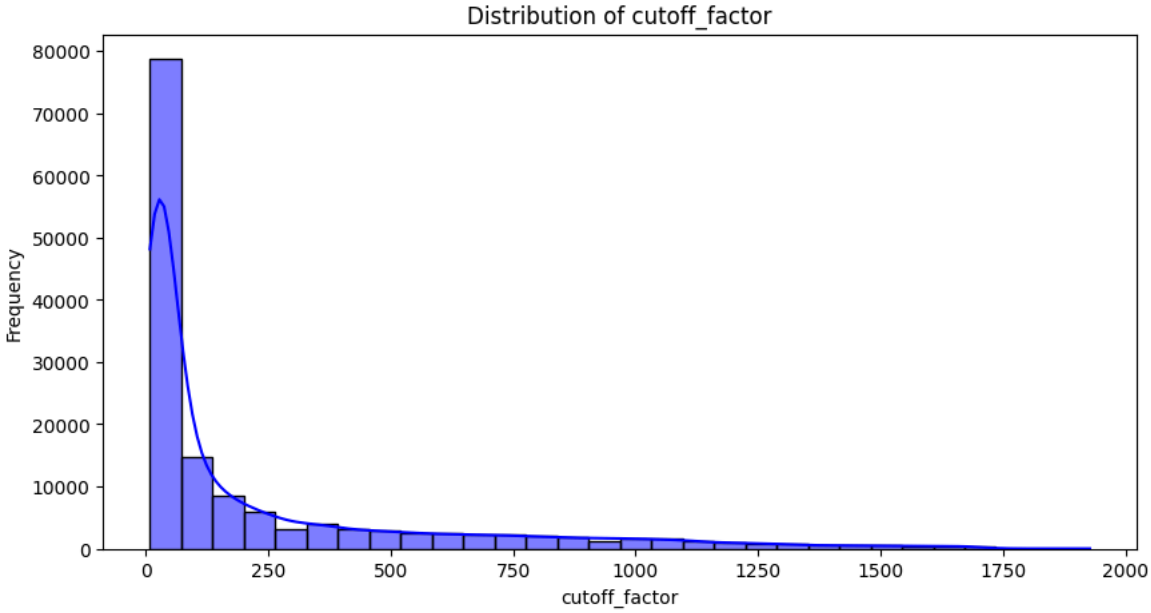
```
# Comments for each univariate and bivariate plot
def generate_comments(df):
    comments = []

    # Univariate comments
    for column in df.columns:
        comments.append(f"Univariate Plot ({column}):")
        if pd.api.types.is_numeric_dtype(df[column]):
            comments.append(f" - The {column} distribution spans from {df[column].min()} to {df[column].max()}.")
        else:
            comments.append(f" - The {column} distribution includes categories: {df[column].unique().tolist()}.")
        comments.append("\n")

    # Bivariate comments
    for i in range(len(df.columns)):
        for j in range(i + 1, len(df.columns)):
            x = df.columns[i]
            y = df.columns[j]
            comments.append(f"Bivariate Plot ({x} vs {y}):")
            comments.append(f" - Relationship analysis between {x} and {y}.")
            comments.append("\n")

    return "\n".join(comments)



# Print the comments on distribution and relationships
print(generate_comments(df))
```

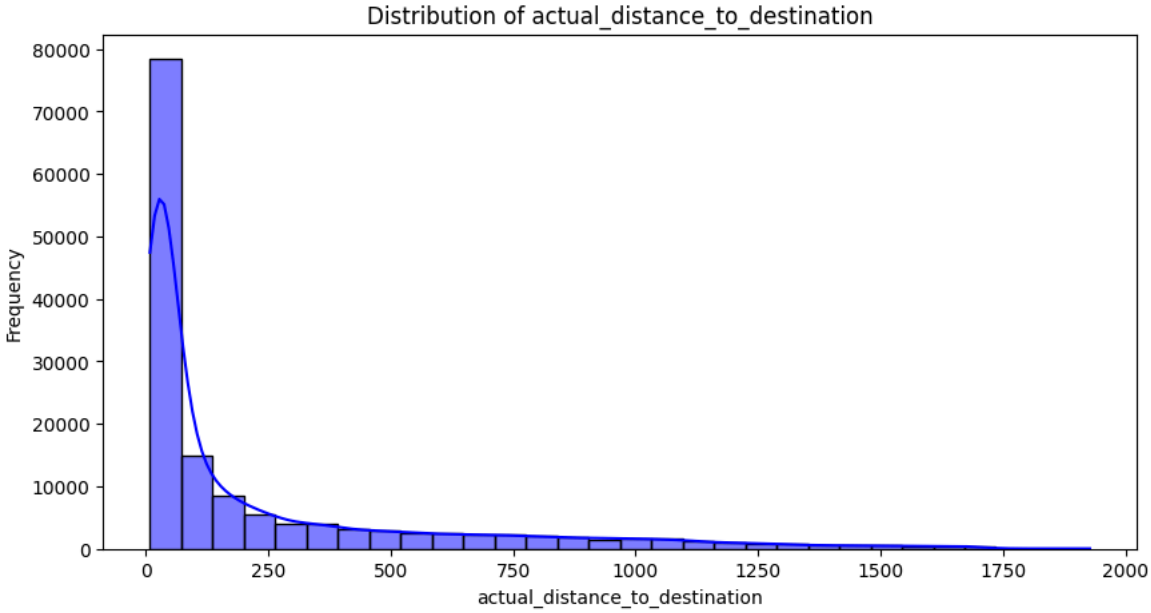
```
 - Relationship analysis between segment_actual_time and segment_osrm_distance.


Bivariate Plot (segment_actual_time vs segment_factor):
 - Relationship analysis between segment_actual_time and segment_factor.


Bivariate Plot (segment_osrm_time vs segment_osrm_distance):
 - Relationship analysis between segment_osrm_time and segment_osrm_distance.


Bivariate Plot (segment_osrm_time vs segment_factor):
 - Relationship analysis between segment_osrm_time and segment_factor.


Bivariate Plot (segment_osrm_distance vs segment_factor):
 - Relationship analysis between segment_osrm_distance and segment_factor.
```

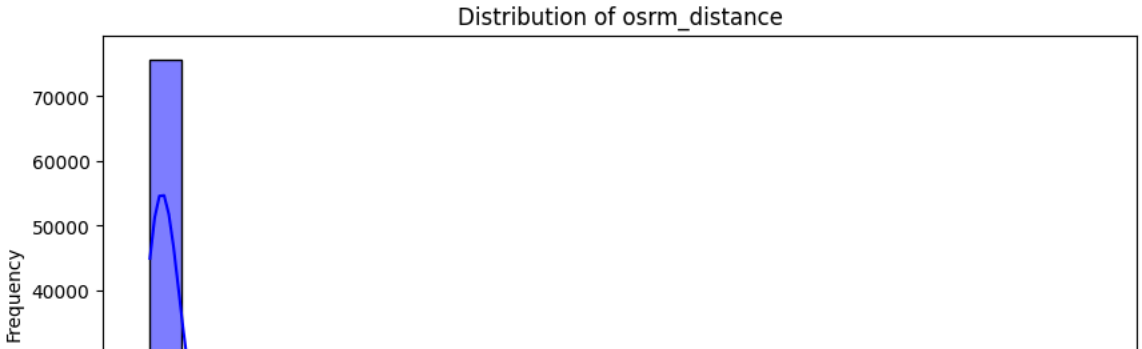
Visual Analysis (distribution plots of all the continuous variable(s), boxplots of all the categorical variables)

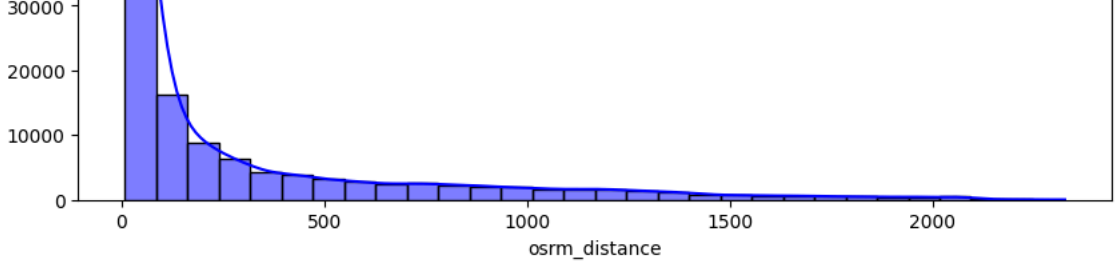
```
# Continuous Variables: Distribution Plots
continuous_columns = [
    'start_scan_to_end_scan', 'cutoff_factor',
    'actual_distance_to_destination', 'osrm_distance', 'factor',
    'segment_osrm_distance', 'segment_factor'
]

for col in continuous_columns:
    plt.figure(figsize=(10, 5))
    sns.histplot(df[col], kde=True, bins=30, color='blue')
    plt.title(f'Distribution of {col}')
    plt.xlabel(col)
    plt.ylabel('Frequency')
    plt.show()

# Categorical Variables: Boxplots
categorical_columns = ['data', 'route_type', 'is_cutoff']

for col in categorical_columns:
    plt.figure(figsize=(10, 5))
    sns.boxplot(x=df[col], y=df['actual_distance_to_destination'], palette='Set2')
    plt.title(f'Boxplot of {col} vs Actual Distance to Destination')
    plt.xlabel(col)
    plt.ylabel('Actual Distance to Destination')
    plt.show()
```

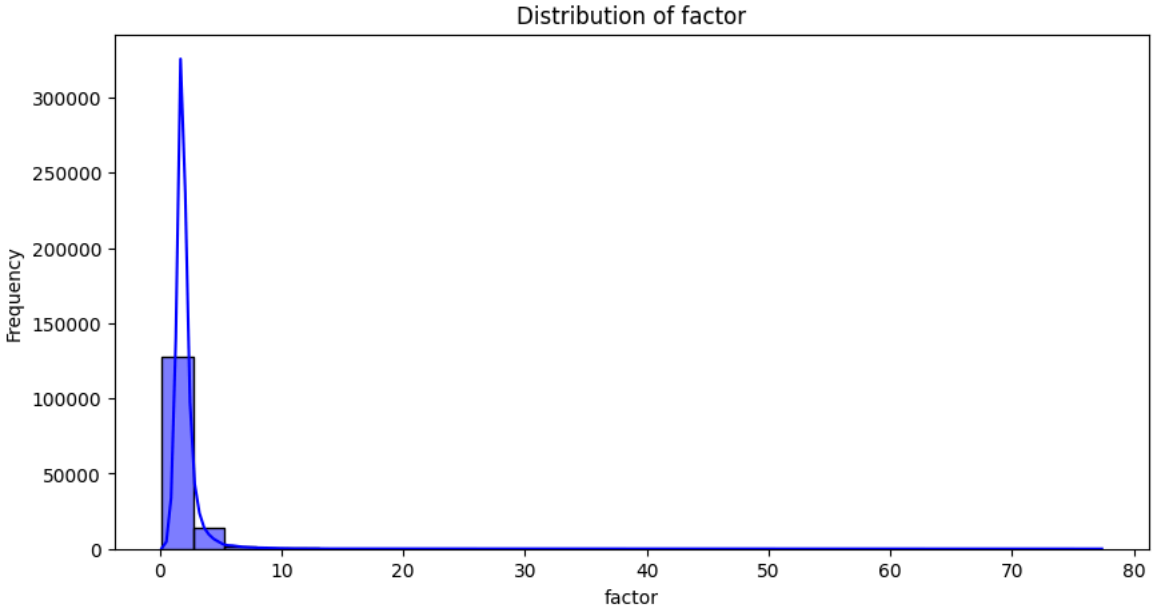

Distribution of start_scan_to_end_scan
Frequency
start_scan_to_end_scan
0
10000
20000
30000
40000
50000
60000
0
1000
2000
3000
4000
5000
6000
7000
8000

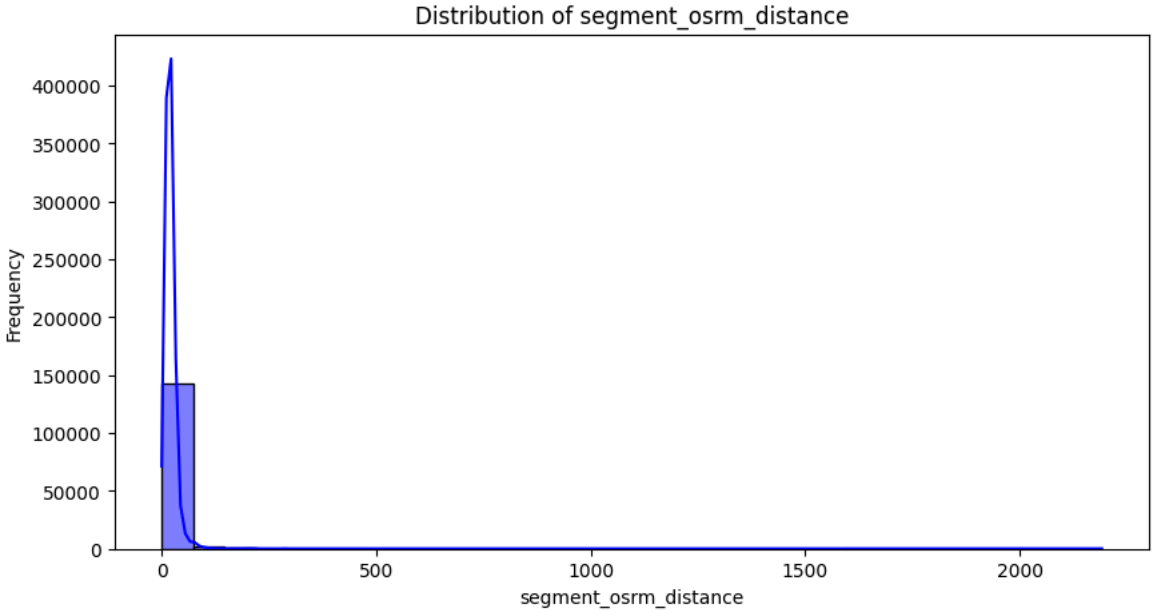

Distribution of cutoff_factor
Frequency
cutoff_factor
0
10000
20000
30000
40000
50000
60000
70000
80000
0
250
500
750
1000
1250
1500
1750
2000

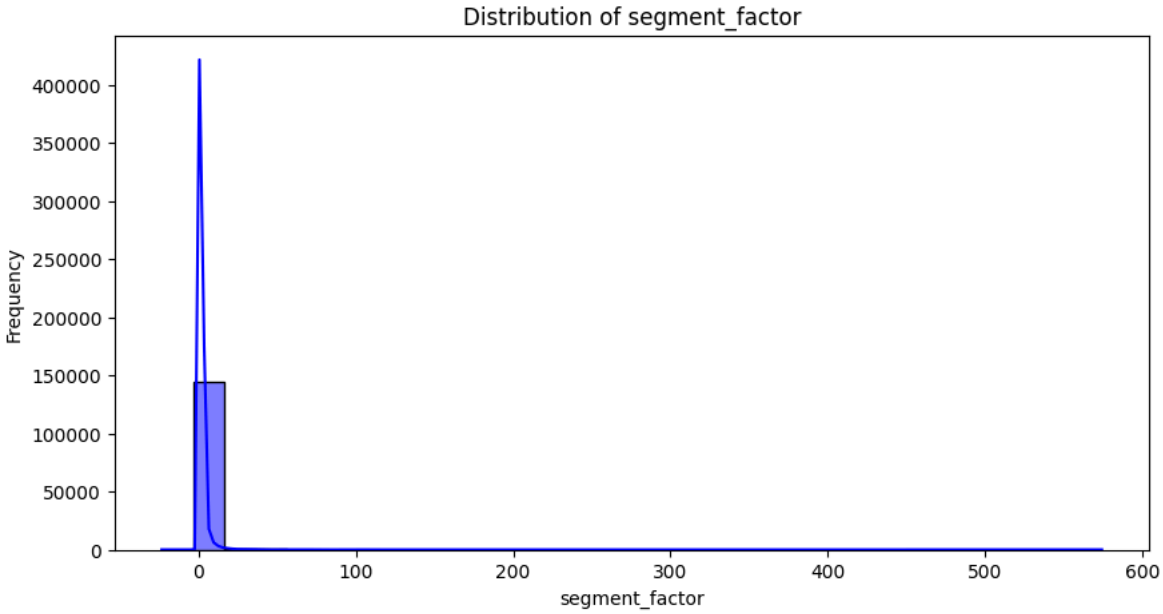

Distribution of actual_distance_to_destination
Frequency
actual_distance_to_destination
0
10000
20000
30000
40000
50000
60000
70000
80000
0
250
500
750
1000
1250
1500
1750
2000

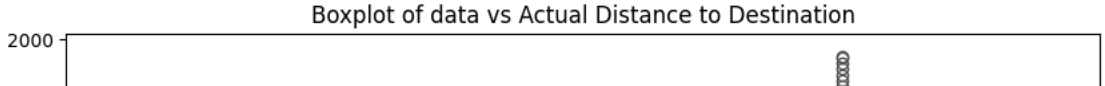

Distribution of osrm_distance
Frequency
40000
50000
60000
70000

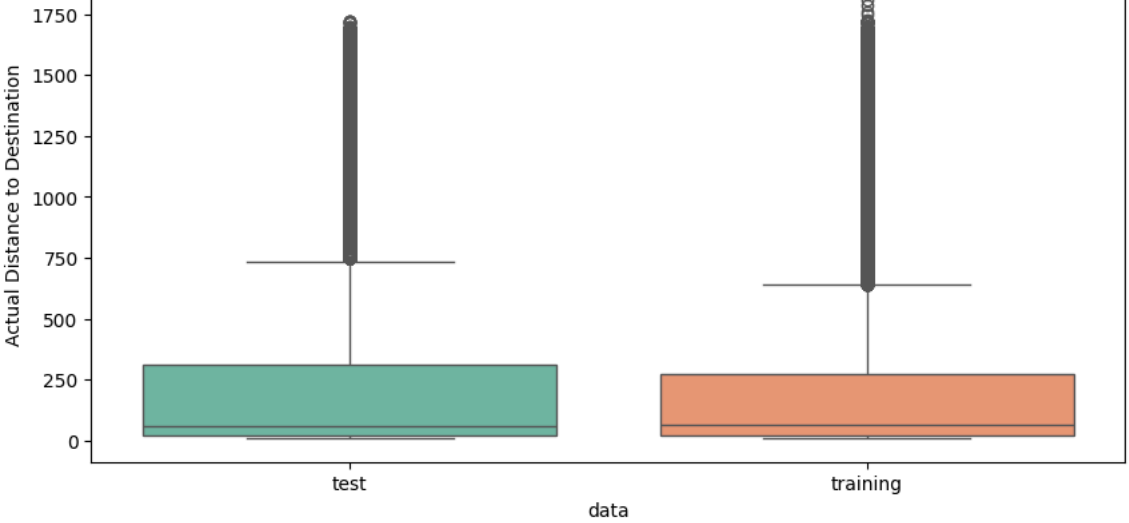

30000
20000
10000
0
0
500
1000
1500
2000
osrm_distance

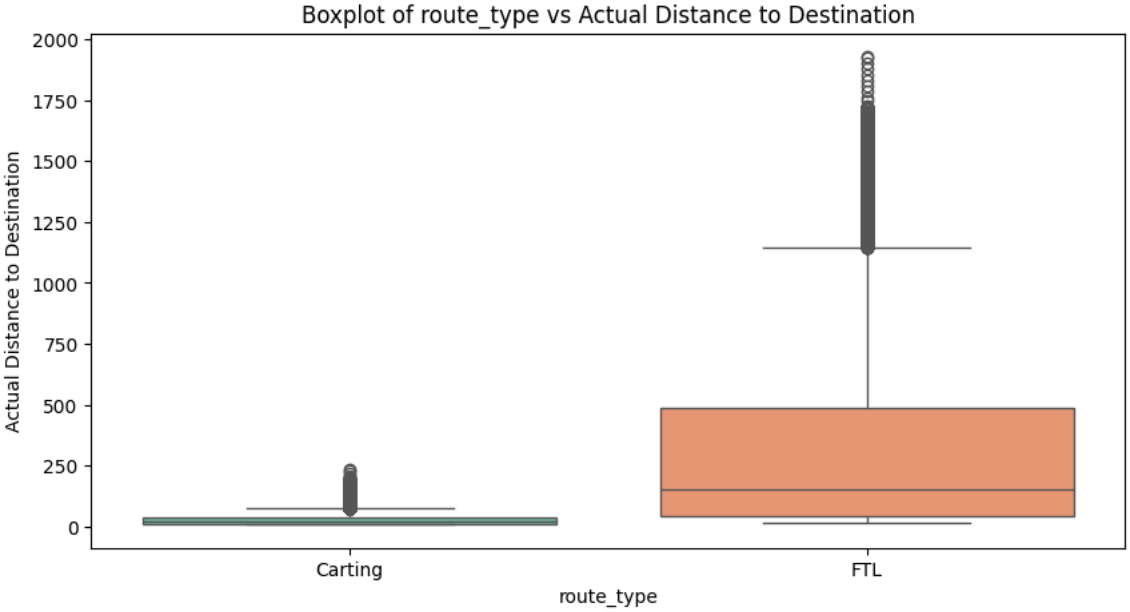

Distribution of factor
Frequency
300000
250000
200000
150000
100000
50000
0
0
10
20
30
40
50
60
70
80
factor

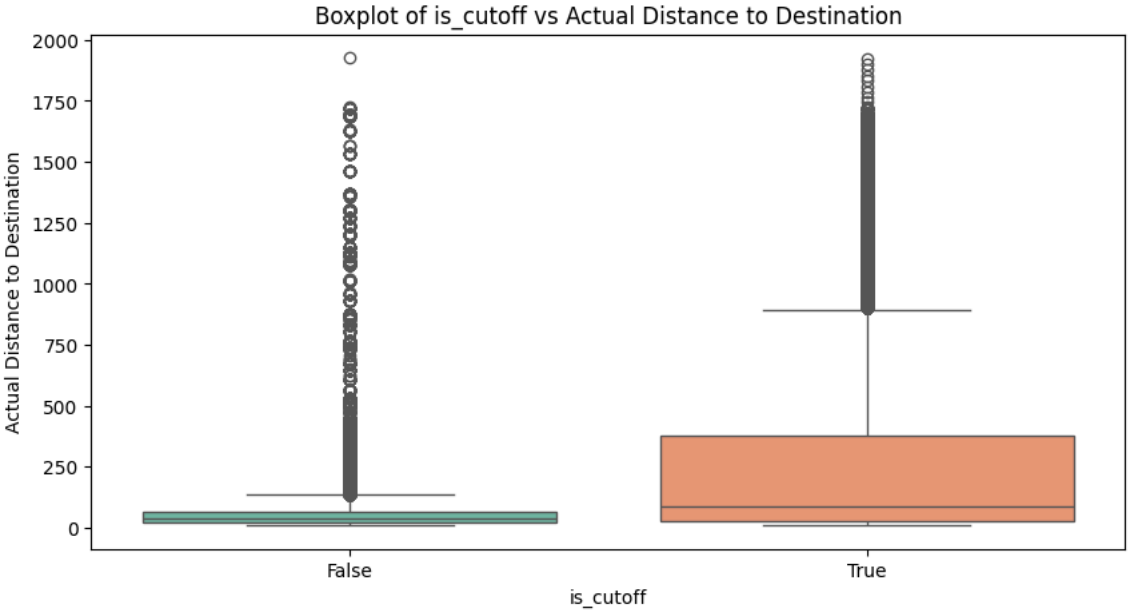

Distribution of segment_osrm_distance
Frequency
400000
350000
300000
250000
200000
150000
100000
50000
0
0
500
1000
1500
2000
segment_osrm_distance


Distribution of segment_factor
Frequency
400000
350000
300000
250000
200000
150000
100000
50000
0
0
100
200
300
400
500
600
segment_factor


Boxplot of data vs Actual Distance to Destination
2000

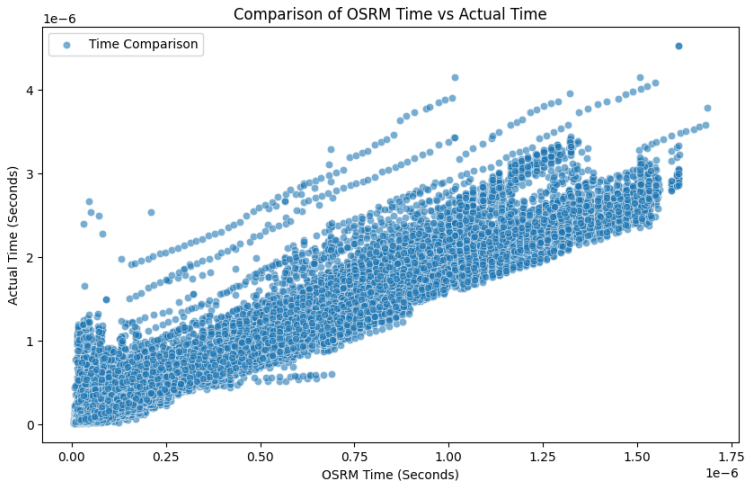

Actual Distance to Destination
1750
1500
1250
1000
750
500
250
0
test
training
data

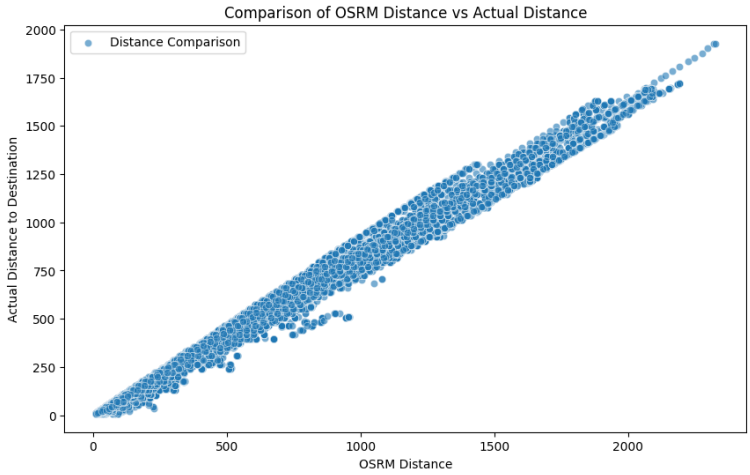

Boxplot of route_type vs Actual Distance to Destination
Actual Distance to Destination
2000
1750
1500
1250
1000
750
500
250
0
Carting
FTL
route_type

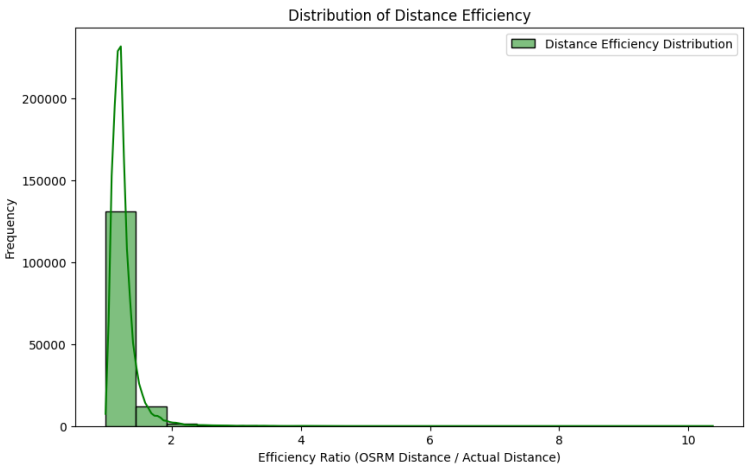

Boxplot of is_cutoff vs Actual Distance to Destination
Actual Distance to Destination
2000
1750
1500
1250
1000
750
500
250
0
False
True
is_cutoff

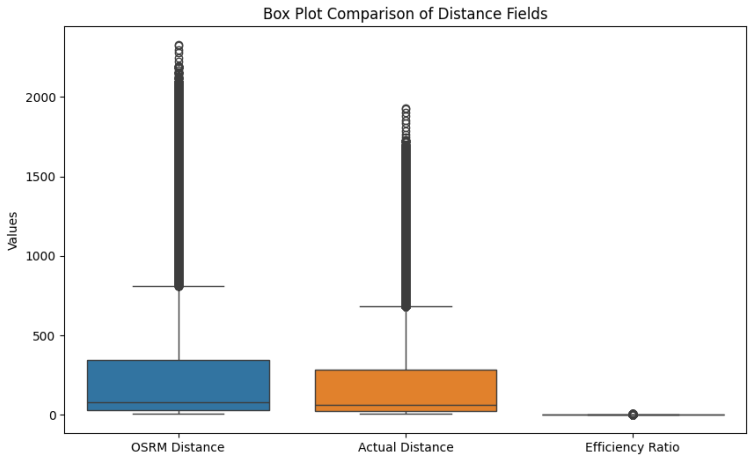
Comparison & Visualization of time and distance fields

```
# Convert time columns to numeric (epoch)
df['actual_time_seconds'] = (pd.to_datetime(df['actual_time']) - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)
df['osrm_time_seconds'] = (pd.to_datetime(df['osrm_time']) - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)

# Calculate distance efficiency
df['distance_efficiency'] = df['osrm_distance'] / df['actual_distance_to_destination']

# Visualization 1 - Time Comparison (Scatter Plot)
plt.figure(figsize=(10, 6))
sns.scatterplot(x='osrm_time_seconds', y='actual_time_seconds', data=df, alpha=0.6, label='Time Comparison')
plt.xlabel('OSRM Time (Seconds)')
plt.ylabel('Actual Time (Seconds)')
plt.title('Comparison of OSRM Time vs Actual Time')
plt.legend()
plt.show()

# Visualization 2 - Distance Comparison (Scatter Plot)
plt.figure(figsize=(10, 6))
sns.scatterplot(x='osrm_distance', y='actual_distance_to_destination', data=df, alpha=0.6, label='Distance Comparison')
plt.xlabel('OSRM Distance')
plt.ylabel('Actual Distance to Destination')
plt.title('Comparison of OSRM Distance vs Actual Distance')
plt.legend()
plt.show()

# Visualization 3 - Distance Efficiency Distribution (Histogram)
plt.figure(figsize=(10, 6))
sns.histplot(df['distance_efficiency'], bins=20, kde=True, color='green', label='Distance Efficiency Distribution')
plt.xlabel('Efficiency Ratio (OSRM Distance / Actual Distance)')
plt.ylabel('Frequency')
plt.title('Distribution of Distance Efficiency')
plt.legend()
plt.show()

# Visualization 4 - Box Plot Comparison
plt.figure(figsize=(10, 6))
sns.boxplot(data=df[['osrm_distance', 'actual_distance_to_destination', 'distance_efficiency']])
plt.xticks(ticks=[0, 1, 2], labels=['OSRM Distance', 'Actual Distance', 'Efficiency Ratio'])
plt.ylabel('Values')
plt.title('Box Plot Comparison of Distance Fields')
plt.show()
```







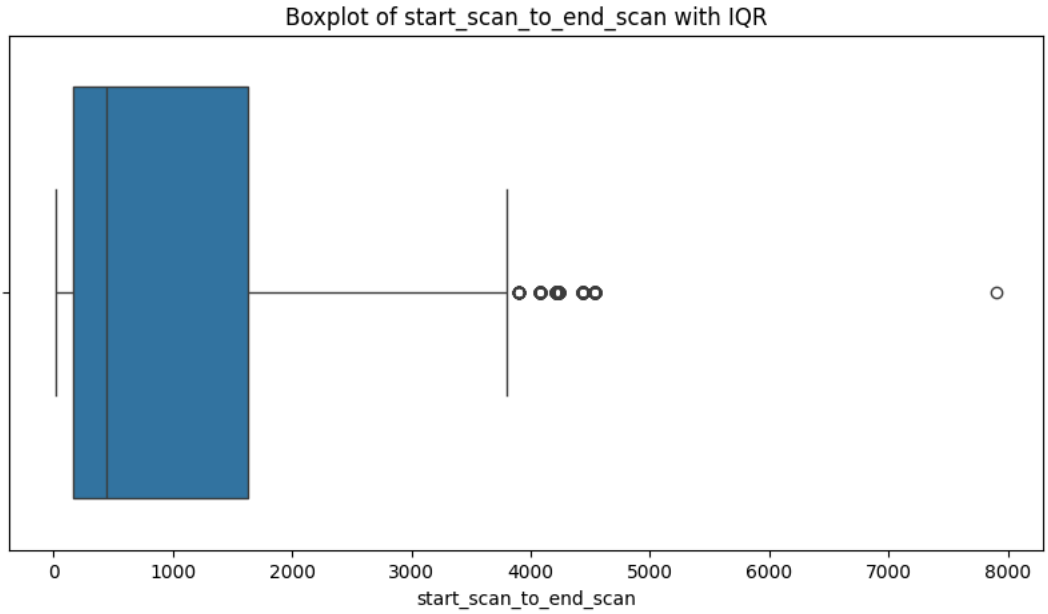


2.Check for Outliers and deal with them accordingly

```
continuous_columns = [
    'start_scan_to_end_scan', 'cutoff_factor',
    'actual_distance_to_destination', 'osrm_distance', 'factor',
    'segment_osrm_distance', 'segment_factor'
]

#i)You can use Boxplot, Interquartile Range (IQR

# IQR Method to find outliers and plot boxplots
for column in continuous_columns:
    Q1 = df[column].quantile(0.25)
    Q3 = df[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    print(column)
    print("Q1: ",Q1)
    print("Q3: ",Q3)
    print("IQR: ",IQR)
    print("Lower Bound: ", lower_bound)
    print("Upper Bound: ", upper_bound)
    outliers = df[(df[column] < lower_bound) | (df[column] > upper_bound)]
    print(f"Outliers in {column} (IQR Method): ")
    if outliers.empty:
      print("No,Outliers Detected for ", column , "\n")
    else:
      print(outliers)
      plt.figure(figsize=(10, 5))
      sns.boxplot(x=df[column])
      plt.title(f'Boxplot of {column} with IQR')
      plt.show()

      df_new = df[(df[column] >= lower_bound) & (df[column] <= upper_bound)]

      # Print the new dataframe without outliers
      print("Dataframe without outliers for", column)
      print(df_new)
```

```
start_scan_to_end_scan
Q1:  161.0
Q3:  1634.0
IQR:  1473.0
Lower Bound:  -2048.5
Upper Bound:  3843.5
Outliers in start_scan_to_end_scan (IQR Method):
            data         trip_creation_time  \
32950   training 2018-09-13 01:28:45.326644
32951   training 2018-09-13 01:28:45.326644
32952   training 2018-09-13 01:28:45.326644
32953   training 2018-09-13 01:28:45.326644
32954   training 2018-09-13 01:28:45.326644
...          ...                       ...
79524   training 2018-09-19 13:44:58.665210
79525   training 2018-09-19 13:44:58.665210
79526   training 2018-09-19 13:44:58.665210
79527   training 2018-09-19 13:44:58.665210
123196      test 2018-10-01 23:35:54.432745

                                     route_schedule_uuid route_type  \
32950   thanos::sroute:6b87651c-fdf4-432f-bf80-0e394f3...        FTL
32951   thanos::sroute:6b87651c-fdf4-432f-bf80-0e394f3...        FTL
32952   thanos::sroute:6b87651c-fdf4-432f-bf80-0e394f3...        FTL
32953   thanos::sroute:6b87651c-fdf4-432f-bf80-0e394f3...        FTL
32954   thanos::sroute:6b87651c-fdf4-432f-bf80-0e394f3...        FTL
...                                                   ...        ...
79524   thanos::sroute:bc7dbb1d-9379-4674-b8d3-f9c3b96...        FTL
79525   thanos::sroute:bc7dbb1d-9379-4674-b8d3-f9c3b96...        FTL
79526   thanos::sroute:bc7dbb1d-9379-4674-b8d3-f9c3b96...        FTL
79527   thanos::sroute:bc7dbb1d-9379-4674-b8d3-f9c3b96...        FTL
123196  thanos::sroute:4316e05f-b4cc-4ea7-b801-62a93ae...    Carting

                     trip_uuid source_center  \
32950   trip-153680212532637033  IND712311AAA
32951   trip-153680212532637033  IND712311AAA
32952   trip-153680212532637033  IND712311AAA
32953   trip-153680212532637033  IND712311AAA
32954   trip-153680212532637033  IND712311AAA
...                         ...           ...
79524   trip-153736469866480991  IND000000ACB
79525   trip-153736469866480991  IND000000ACB
79526   trip-153736469866480991  IND000000ACB
79527   trip-153736469866480991  IND000000ACB
123196  trip-153843695443252828  IND764071AAB

                              source_name destination_center  \
32950      Kolkata_Dankuni_HB (West Bengal)       IND781018AAB
32951      Kolkata_Dankuni_HB (West Bengal)       IND781018AAB
32952      Kolkata_Dankuni_HB (West Bengal)       IND781018AAB
32953      Kolkata_Dankuni_HB (West Bengal)       IND781018AAB
32954      Kolkata_Dankuni_HB (West Bengal)       IND781018AAB
...                                     ...                ...
79524         Gurgaon_Bilaspur_HB (Haryana)       IND712311AAA
79525         Gurgaon_Bilaspur_HB (Haryana)       IND712311AAA
79526         Gurgaon_Bilaspur_HB (Haryana)       IND712311AAA
79527         Gurgaon_Bilaspur_HB (Haryana)       IND712311AAA
123196  Pappadahandi_Central_DPP_2 (Orissa)       IND530012AAA

                                destination_name              od_start_time  \
32950                        Guwahati_Hub (Assam) 2018-09-13 01:28:45.326644
32951                        Guwahati_Hub (Assam) 2018-09-13 01:28:45.326644
32952                        Guwahati_Hub (Assam) 2018-09-13 01:28:45.326644
32953                        Guwahati_Hub (Assam) 2018-09-13 01:28:45.326644
32954                        Guwahati_Hub (Assam) 2018-09-13 01:28:45.326644
...                                            ...                       ...
79524            Kolkata_Dankuni_HB (West Bengal) 2018-09-19 13:44:58.665210
79525            Kolkata_Dankuni_HB (West Bengal) 2018-09-19 13:44:58.665210
79526            Kolkata_Dankuni_HB (West Bengal) 2018-09-19 13:44:58.665210
79527            Kolkata_Dankuni_HB (West Bengal) 2018-09-19 13:44:58.665210
123196  Visakhapatnam_Gajuwaka_IP (Andhra Pradesh) 2018-10-02 15:21:51.236205

        ...    cutoff_timestamp  actual_distance_to_destination  \
32950   ... 2018-09-15 09:59:21                       22.370663
32951   ... 2018-09-15 09:11:23                       44.402060
32952   ... 2018-09-15 08:11:31                       68.958075
32953   ... 2018-09-15 07:27:32                       89.822481
32954   ... 2018-09-15 05:47:23                      110.360839
...     ...                 ...                            ...
79524   ... 2018-09-19 18:47:20                     1254.334355
79525   ... 2018-09-19 18:21:19                     1276.644697
79526   ... 2018-09-19 17:53:20                     1300.489196
79527   ... 2018-09-19 13:49:19                     1300.480089
123196  ... 2018-10-02 15:32:30                      196.451691

                        actual_time                     osrm_time  \
32950  1970-01-01 00:00:00.000000057 1970-01-01 00:00:00.000000036
32951  1970-01-01 00:00:00.000000105 1970-01-01 00:00:00.000000064
32952  1970-01-01 00:00:00.000000165 1970-01-01 00:00:00.000000092
32953  1970-01-01 00:00:00.000000209 1970-01-01 00:00:00.000000102
32954  1970-01-01 00:00:00.000000309 1970-01-01 00:00:00.000000121
...                              ...                           ...
79524  1970-01-01 00:00:00.000003854 1970-01-01 00:00:00.000000973
79525  1970-01-01 00:00:00.000003880 1970-01-01 00:00:00.000000990
79526  1970-01-01 00:00:00.000003908 1970-01-01 00:00:00.000001010
79527  1970-01-01 00:00:00.000004152 1970-01-01 00:00:00.000001015
123196 1970-01-01 00:00:00.000002541 1970-01-01 00:00:00.000000211
```

```
123196 1970-01-01 00:00:00.000002541 1970-01-01 00:00:00.000000211

       osrm_distance     factor           segment_actual_time  \
32950        39.3493   1.583333 1970-01-01 00:00:00.000000057
32951        78.6763   1.640625 1970-01-01 00:00:00.000000047
32952       118.7763   1.793478 1970-01-01 00:00:00.000000059
32953       133.1381   2.049020 1970-01-01 00:00:00.000000043
32954       159.9273   2.553719 1970-01-01 00:00:00.000000100
...              ...        ...                            ...
79524      1377.4674   3.960946 1970-01-01 00:00:00.000000052
79525      1400.5472   3.919192 1970-01-01 00:00:00.000000026
79526      1426.5927   3.869307 1970-01-01 00:00:00.000000027
79527      1427.0313   4.090640 1970-01-01 00:00:00.000000244
123196      293.3271  12.042654 1970-01-01 00:00:00.000002541

                   segment_osrm_time  segment_osrm_distance  segment_factor
32950  1970-01-01 00:00:00.000000036                39.3493        1.583333
32951  1970-01-01 00:00:00.000000027                39.3270        1.740741
32952  1970-01-01 00:00:00.000000028                40.1000        2.107143
32953  1970-01-01 00:00:00.000000025                35.8611        1.720000
32954  1970-01-01 00:00:00.000000018                26.7892        5.555556
...                              ...                    ...             ...
79524  1970-01-01 00:00:00.000000024                33.8311        2.166667
79525  1970-01-01 00:00:00.000000017                23.0798        1.529412
79526  1970-01-01 00:00:00.000000019                26.0456        1.421053
79527  1970-01-01 00:00:00.000000015                15.5571       16.266667
123196 1970-01-01 00:00:00.000000211               293.3271       12.042654

[373 rows x 24 columns]
```

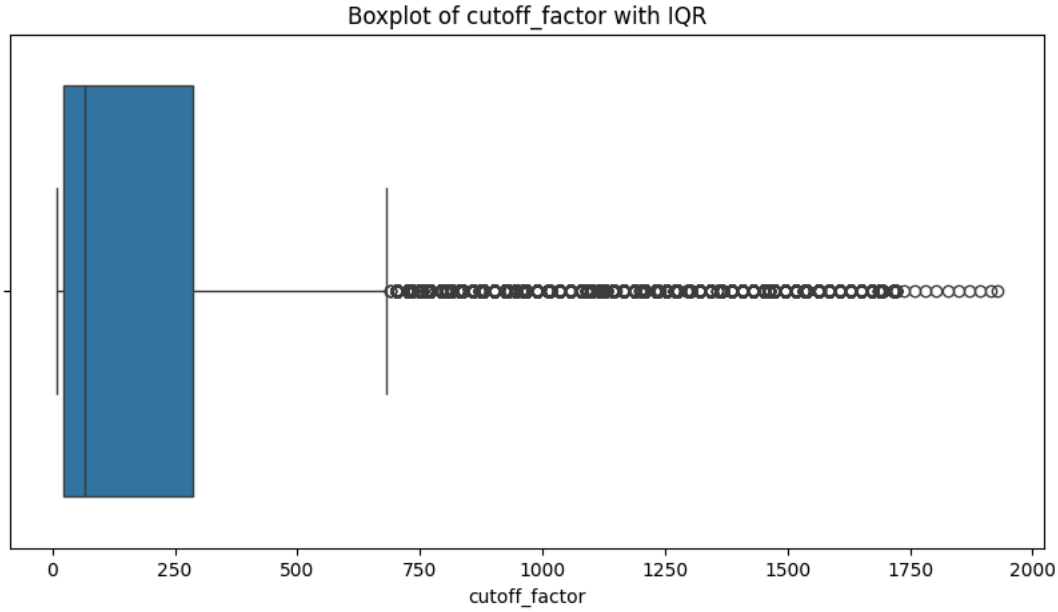



```
Dataframe without outliers for start_scan_to_end_scan
            data         trip_creation_time  \
0       training 2018-09-20 02:35:36.476840
1       training 2018-09-20 02:35:36.476840
2       training 2018-09-20 02:35:36.476840
3       training 2018-09-20 02:35:36.476840
4       training 2018-09-20 02:35:36.476840
...          ...                        ...
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231
144866  training 2018-09-20 16:24:28.436231

                                     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
...                                                   ...        ...
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144866  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

       destination_center               destination_name  \
0            IND388620AAB  Khambhat_MotvdDPP_D (Gujarat)
```

```
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)
...                   ...                            ...
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144866       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                    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  ...                 NaT
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
...                           ...  ...                 ...
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19
144866 2018-09-20 16:24:28.436231  ...                 NaT

        actual_distance_to_destination                   actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                            ...
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158
144866                       70.039010 1970-01-01 00:00:00.000000426

                           osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...           ...       ...
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245
144866 1970-01-01 00:00:00.000000095       88.7319  4.484211

                  segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                            ...
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027
144866 1970-01-01 00:00:00.000000268 1970-01-01 00:00:00.000000009

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630
144866                 8.8088       29.777778

[144494 rows x 24 columns]
cutoff_factor
Q1:  22.0
Q3:  286.0
IQR:  264.0
Lower Bound:  -374.0
Upper Bound:  682.0
Outliers in cutoff_factor (IQR Method):
            data         trip_creation_time  \
402     training 2018-09-25 15:06:59.975279
403     training 2018-09-25 15:06:59.975279
404     training 2018-09-25 15:06:59.975279
405     training 2018-09-25 15:06:59.975279
406     training 2018-09-25 15:06:59.975279
...          ...                        ...
144796      test 2018-10-01 18:17:37.047270
144797      test 2018-10-01 18:17:37.047270
144798      test 2018-10-01 18:17:37.047270
144799      test 2018-10-01 18:17:37.047270
144800      test 2018-10-01 18:17:37.047270
```

```
                                   route_schedule_uuid route_type  \
402     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
403     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
404     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
405     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
406     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
...                                                   ...        ...
144796  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144797  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144798  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144799  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144800  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL

                     trip_uuid source_center  \
402     trip-153788801997503817  IND825409AAA
403     trip-153788801997503817  IND825409AAA
404     trip-153788801997503817  IND825409AAA
405     trip-153788801997503817  IND825409AAA
406     trip-153788801997503817  IND825409AAA
...                         ...           ...
144796  trip-153841785704702048  IND000000ACB
144797  trip-153841785704702048  IND000000ACB
144798  trip-153841785704702048  IND000000ACB
144799  trip-153841785704702048  IND000000ACB
144800  trip-153841785704702048  IND000000ACB

                              source_name destination_center  \
402     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
403     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
404     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
405     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
406     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
...                                   ...                ...
144796      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144797      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144798      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144799      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144800      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA

                       destination_name              od_start_time  ...  \
402        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
403        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
404        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
405        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
406        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
...                                 ...                        ...  ...
144796  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144797  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144798  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144799  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144800  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...

          cutoff_timestamp  actual_distance_to_destination  \
402    2018-09-26 11:05:50                      704.090688
403    2018-09-26 10:29:56                      726.181078
404    2018-09-26 09:41:53                      748.332196
405    2018-09-26 09:11:54                      770.365887
406    2018-09-26 08:45:27                      796.335857
...                    ...                             ...
144796 2018-10-02 14:08:12                     1611.171536
144797 2018-10-02 13:32:23                     1633.419313
144798 2018-10-02 13:08:12                     1650.202066
144799 2018-10-02 12:32:14                     1673.310381
144800 2018-10-02 11:44:15                     1689.639499

                        actual_time                     osrm_time  \
402    1970-01-01 00:00:00.000001071 1970-01-01 00:00:00.000000571
403    1970-01-01 00:00:00.000001106 1970-01-01 00:00:00.000000590
404    1970-01-01 00:00:00.000001154 1970-01-01 00:00:00.000000606
405    1970-01-01 00:00:00.000001184 1970-01-01 00:00:00.000000630
406    1970-01-01 00:00:00.000001211 1970-01-01 00:00:00.000000641
...                              ...                            ...
144796 1970-01-01 00:00:00.000002640 1970-01-01 00:00:00.000001492
144797 1970-01-01 00:00:00.000002675 1970-01-01 00:00:00.000001512
144798 1970-01-01 00:00:00.000002700 1970-01-01 00:00:00.000001532
144799 1970-01-01 00:00:00.000002736 1970-01-01 00:00:00.000001549
144800 1970-01-01 00:00:00.000002784 1970-01-01 00:00:00.000001508

       osrm_distance    factor           segment_actual_time  \
402         767.6109  1.875657 1970-01-01 00:00:00.000000033
403         793.4319  1.874576 1970-01-01 00:00:00.000000035
404         816.5632  1.904290 1970-01-01 00:00:00.000000048
405         850.4080  1.879365 1970-01-01 00:00:00.000000029
406         865.7213  1.889236 1970-01-01 00:00:00.000000026
...              ...       ...                            ...
144796     1980.0975  1.769437 1970-01-01 00:00:00.000000035
144797     2008.9586  1.769180 1970-01-01 00:00:00.000000035
144798     2036.3992  1.762402 1970-01-01 00:00:00.000000024
144799     2059.0195  1.766301 1970-01-01 00:00:00.000000035
144800     2063.7663  1.846154 1970-01-01 00:00:00.000000047

                   segment_osrm_time  segment_osrm_distance  segment_factor
402    1970-01-01 00:00:00.000000020                29.2733        1.650000
403    1970-01-01 00:00:00.000000018                25.8210        1.944444
404    1970-01-01 00:00:00.000000016                23.1313        3.000000
405    1970-01-01 00:00:00.000000023                33.8448        1.260870
406    1970-01-01 00:00:00.000000026                37.8017        1.000000
```

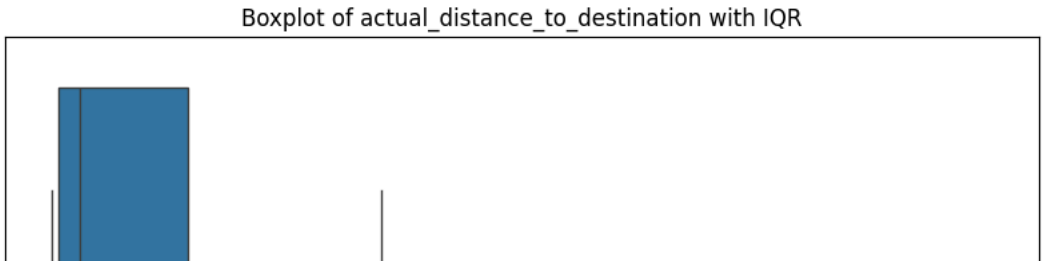
```
...                                 ...                 ...          ...
144796 1970-01-01 00:00:00.000000020                28.5248     1.750000
144797 1970-01-01 00:00:00.000000020                28.8611     1.750000
144798 1970-01-01 00:00:00.000000019                27.4406     1.263158
144799 1970-01-01 00:00:00.000000026                36.7167     1.346154
144800 1970-01-01 00:00:00.000000035                41.4651     1.342857

[17246 rows x 24 columns]
```

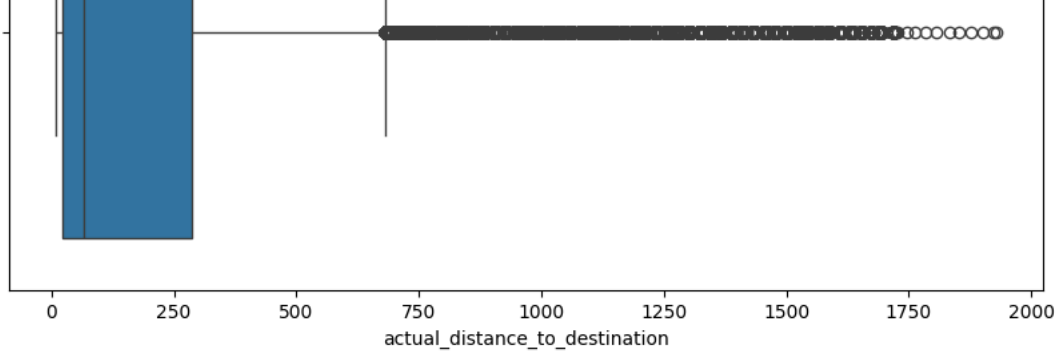



```
Dataframe without outliers for cutoff_factor
            data         trip_creation_time  \
0       training 2018-09-20 02:35:36.476840
1       training 2018-09-20 02:35:36.476840
2       training 2018-09-20 02:35:36.476840
3       training 2018-09-20 02:35:36.476840
4       training 2018-09-20 02:35:36.476840
...          ...                         ...
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231
144866  training 2018-09-20 16:24:28.436231

                                   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
...                                                   ...        ...
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144866  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

       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)
...                   ...                           ...
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144866       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                    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  ...                 NaT
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
...                           ...  ...                 ...
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:21:18
```

```
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19
144866 2018-09-20 16:24:28.436231  ...                 NaT

        actual_distance_to_destination                   actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                            ...
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158
144866                       70.039010 1970-01-01 00:00:00.000000426

                          osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...           ...       ...
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245
144866 1970-01-01 00:00:00.000000095       88.7319  4.484211

                 segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                            ...
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027
144866 1970-01-01 00:00:00.000000268 1970-01-01 00:00:00.000000009

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630
144866                 8.8088       29.777778

[127621 rows x 24 columns]
actual_distance_to_destination
Q1:  23.355874361432974
Q3:  286.7088745976663
IQR:  263.3530002362333
Lower Bound:  -371.6736259929169
Upper Bound:  681.7383749520162
Outliers in actual_distance_to_destination (IQR Method):
            data         trip_creation_time  \
401     training 2018-09-25 15:06:59.975279
402     training 2018-09-25 15:06:59.975279
403     training 2018-09-25 15:06:59.975279
404     training 2018-09-25 15:06:59.975279
405     training 2018-09-25 15:06:59.975279
...          ...                        ...
144796      test 2018-10-01 18:17:37.047270
144797      test 2018-10-01 18:17:37.047270
144798      test 2018-10-01 18:17:37.047270
144799      test 2018-10-01 18:17:37.047270
144800      test 2018-10-01 18:17:37.047270

                                     route_schedule_uuid route_type  \
401     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
402     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
403     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
404     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
405     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
...                                               ...        ...
144796  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144797  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144798  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144799  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144800  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL

                      trip_uuid source_center  \
401     trip-153788801997503817  IND825409AAA
402     trip-153788801997503817  IND825409AAA
403     trip-153788801997503817  IND825409AAA
404     trip-153788801997503817  IND825409AAA
405     trip-153788801997503817  IND825409AAA
```

```
...                          ...            ...
144796  trip-153841785704702048  IND000000ACB
144797  trip-153841785704702048  IND000000ACB
144798  trip-153841785704702048  IND000000ACB
144799  trip-153841785704702048  IND000000ACB
144800  trip-153841785704702048  IND000000ACB

                            source_name destination_center  \
401     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
402     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
403     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
404     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
405     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
...                                   ...                ...
144796      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144797      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144798      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144799      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144800      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA

                       destination_name              od_start_time  ...  \
401        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
402        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
403        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
404        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
405        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
...                                 ...                         ...  ...
144796  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144797  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144798  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144799  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144800  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...

          cutoff_timestamp  actual_distance_to_destination  \
401    2018-09-26 11:38:50                      682.175085
402    2018-09-26 11:05:50                      704.090688
403    2018-09-26 10:29:56                      726.181078
404    2018-09-26 09:41:53                      748.332196
405    2018-09-26 09:11:54                      770.365887
...                    ...                             ...
144796 2018-10-02 14:08:12                     1611.171536
144797 2018-10-02 13:32:23                     1633.419313
144798 2018-10-02 13:08:12                     1650.202066
144799 2018-10-02 12:32:14                     1673.310381
144800 2018-10-02 11:44:15                     1689.639499

                        actual_time                     osrm_time  \
401    1970-01-01 00:00:00.000001038 1970-01-01 00:00:00.000000558
402    1970-01-01 00:00:00.000001071 1970-01-01 00:00:00.000000571
403    1970-01-01 00:00:00.000001106 1970-01-01 00:00:00.000000590
404    1970-01-01 00:00:00.000001154 1970-01-01 00:00:00.000000606
405    1970-01-01 00:00:00.000001184 1970-01-01 00:00:00.000000630
...                              ...                            ...
144796 1970-01-01 00:00:00.000002640 1970-01-01 00:00:00.000001492
144797 1970-01-01 00:00:00.000002675 1970-01-01 00:00:00.000001512
144798 1970-01-01 00:00:00.000002700 1970-01-01 00:00:00.000001532
144799 1970-01-01 00:00:00.000002736 1970-01-01 00:00:00.000001549
144800 1970-01-01 00:00:00.000002784 1970-01-01 00:00:00.000001508

       osrm_distance    factor           segment_actual_time  \
401         746.9369  1.860215 1970-01-01 00:00:00.000000048
402         767.6109  1.875657 1970-01-01 00:00:00.000000033
403         793.4319  1.874576 1970-01-01 00:00:00.000000035
404         816.5632  1.904290 1970-01-01 00:00:00.000000048
405         850.4080  1.879365 1970-01-01 00:00:00.000000029
...              ...       ...                            ...
144796     1980.0975  1.769437 1970-01-01 00:00:00.000000035
144797     2008.9586  1.769180 1970-01-01 00:00:00.000000035
144798     2036.3992  1.762402 1970-01-01 00:00:00.000000024
144799     2059.0195  1.766301 1970-01-01 00:00:00.000000035
144800     2063.7663  1.846154 1970-01-01 00:00:00.000000047

                   segment_osrm_time  segment_osrm_distance  segment_factor
401    1970-01-01 00:00:00.000000023                32.2516        2.086957
402    1970-01-01 00:00:00.000000020                29.2733        1.650000
403    1970-01-01 00:00:00.000000018                25.8210        1.944444
404    1970-01-01 00:00:00.000000016                23.1313        3.000000
405    1970-01-01 00:00:00.000000023                33.8448        1.260870
...                              ...                    ...             ...
144796 1970-01-01 00:00:00.000000020                28.5248        1.750000
144797 1970-01-01 00:00:00.000000020                28.8611        1.750000
144798 1970-01-01 00:00:00.000000019                27.4406        1.263158
144799 1970-01-01 00:00:00.000000026                36.7167        1.346154
144800 1970-01-01 00:00:00.000000035                41.4651        1.342857

[17992 rows x 24 columns]
```

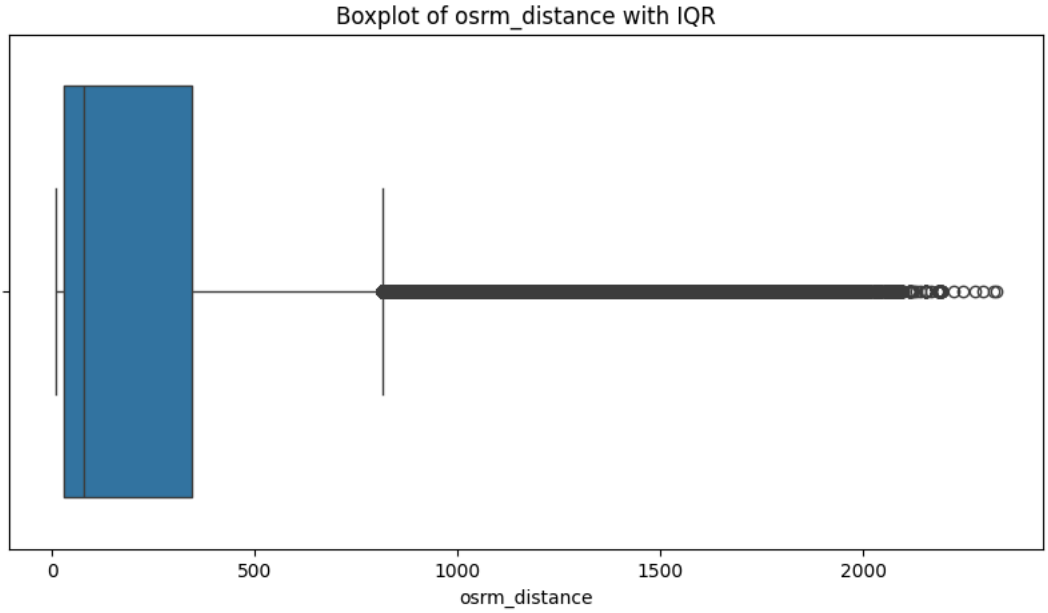





```
Dataframe without outliers for actual_distance_to_destination
            data         trip_creation_time  \
0       training 2018-09-20 02:35:36.476840
1       training 2018-09-20 02:35:36.476840
2       training 2018-09-20 02:35:36.476840
3       training 2018-09-20 02:35:36.476840
4       training 2018-09-20 02:35:36.476840
...          ...                        ...
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231
144866  training 2018-09-20 16:24:28.436231

                                   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
...                                                   ...        ...
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144866  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

       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)
...                    ...                           ...
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144866       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                    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  ...                 NaT
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
...                           ...  ...                 ...
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19
144866 2018-09-20 16:24:28.436231  ...                 NaT

        actual_distance_to_destination                    actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                           ...
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158
144866                       70.039010 1970-01-01 00:00:00.000000426

                          osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7242  1.200000
```

```
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...          ...       ...
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245
144866 1970-01-01 00:00:00.000000095       88.7319  4.484211

                  segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                            ...
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027
144866 1970-01-01 00:00:00.000000268 1970-01-01 00:00:00.000000009

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630
144866                 8.8088       29.777778

[126875 rows x 24 columns]
osrm_distance
Q1:  29.9147
Q3:  343.19325000000003
IQR:  313.27855000000005
Lower Bound:  -440.0031250000001
Upper Bound:  813.1110750000001
Outliers in osrm_distance (IQR Method):
            data         trip_creation_time  \
404     training 2018-09-25 15:06:59.975279
405     training 2018-09-25 15:06:59.975279
406     training 2018-09-25 15:06:59.975279
407     training 2018-09-25 15:06:59.975279
408     training 2018-09-25 15:06:59.975279
...          ...                        ...
144796      test 2018-10-01 18:17:37.047270
144797      test 2018-10-01 18:17:37.047270
144798      test 2018-10-01 18:17:37.047270
144799      test 2018-10-01 18:17:37.047270
144800      test 2018-10-01 18:17:37.047270

                                     route_schedule_uuid route_type  \
404     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
405     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
406     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
407     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
408     thanos::sroute:51d8aa58-9b5b-4bc2-81e9-bb284c6...        FTL
...                                               ...        ...
144796  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144797  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144798  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144799  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144800  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL

                      trip_uuid source_center  \
404     trip-153788801997503817  IND825409AAA
405     trip-153788801997503817  IND825409AAA
406     trip-153788801997503817  IND825409AAA
407     trip-153788801997503817  IND825409AAA
408     trip-153788801997503817  IND825409AAA
...                         ...           ...
144796  trip-153841785704702048  IND000000ACB
144797  trip-153841785704702048  IND000000ACB
144798  trip-153841785704702048  IND000000ACB
144799  trip-153841785704702048  IND000000ACB
144800  trip-153841785704702048  IND000000ACB

                              source_name destination_center  \
404     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
405     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
406     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
407     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
408     JhumriTlya_RadhaCpx_D (Jharkhand)       IND000000ACB
...                                   ...                ...
144796      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144797      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144798      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144799      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144800      Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA

                         destination_name               od_start_time  \
```

```
                         destination_name              od_start_time  ...  \
404        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
405        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
406        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
407        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
408        Gurgaon_Bilaspur_HB (Haryana) 2018-09-26 03:15:43.970231  ...
...                                 ...                          ...  ...
144796  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144797  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144798  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144799  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144800  Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...

          cutoff_timestamp  actual_distance_to_destination  \
404    2018-09-26 09:41:53                      748.332196
405    2018-09-26 09:11:54                      770.365887
406    2018-09-26 08:45:27                      796.335857
407    2018-09-26 08:15:26                      815.152708
408    2018-09-26 07:39:27                      836.016287
...                    ...                             ...
144796 2018-10-02 14:08:12                     1611.171536
144797 2018-10-02 13:32:23                     1633.419313
144798 2018-10-02 13:08:12                     1650.202066
144799 2018-10-02 12:32:14                     1673.310381
144800 2018-10-02 11:44:15                     1689.639499

                        actual_time                     osrm_time  \
404    1970-01-01 00:00:00.000001154 1970-01-01 00:00:00.000000606
405    1970-01-01 00:00:00.000001184 1970-01-01 00:00:00.000000630
406    1970-01-01 00:00:00.000001211 1970-01-01 00:00:00.000000641
407    1970-01-01 00:00:00.000001241 1970-01-01 00:00:00.000000655
408    1970-01-01 00:00:00.000001277 1970-01-01 00:00:00.000000671
...                              ...                             ...
144796 1970-01-01 00:00:00.000002640 1970-01-01 00:00:00.000001492
144797 1970-01-01 00:00:00.000002675 1970-01-01 00:00:00.000001512
144798 1970-01-01 00:00:00.000002700 1970-01-01 00:00:00.000001532
144799 1970-01-01 00:00:00.000002736 1970-01-01 00:00:00.000001549
144800 1970-01-01 00:00:00.000002784 1970-01-01 00:00:00.000001508

       osrm_distance    factor           segment_actual_time  \
404         816.5632  1.904290 1970-01-01 00:00:00.000000048
405         850.4080  1.879365 1970-01-01 00:00:00.000000029
406         865.7213  1.889236 1970-01-01 00:00:00.000000026
407         886.1183  1.894656 1970-01-01 00:00:00.000000030
408         908.4596  1.903130 1970-01-01 00:00:00.000000035
...              ...       ...                            ...
144796     1980.0975  1.769437 1970-01-01 00:00:00.000000035
144797     2008.9586  1.769180 1970-01-01 00:00:00.000000035
144798     2036.3992  1.762402 1970-01-01 00:00:00.000000024
144799     2059.0195  1.766301 1970-01-01 00:00:00.000000035
144800     2063.7663  1.846154 1970-01-01 00:00:00.000000047

                   segment_osrm_time  segment_osrm_distance  segment_factor
404    1970-01-01 00:00:00.000000016                23.1313        3.000000
405    1970-01-01 00:00:00.000000023                33.8448        1.260870
406    1970-01-01 00:00:00.000000026                37.8017        1.000000
407    1970-01-01 00:00:00.000000014                20.3970        2.142857
408    1970-01-01 00:00:00.000000015                22.3413        2.333333
...                              ...                    ...             ...
144796 1970-01-01 00:00:00.000000020                28.5248        1.750000
144797 1970-01-01 00:00:00.000000020                28.8611        1.750000
144798 1970-01-01 00:00:00.000000019                27.4406        1.263158
144799 1970-01-01 00:00:00.000000026                36.7167        1.346154
144800 1970-01-01 00:00:00.000000035                41.4651        1.342857

[17816 rows x 24 columns]
```

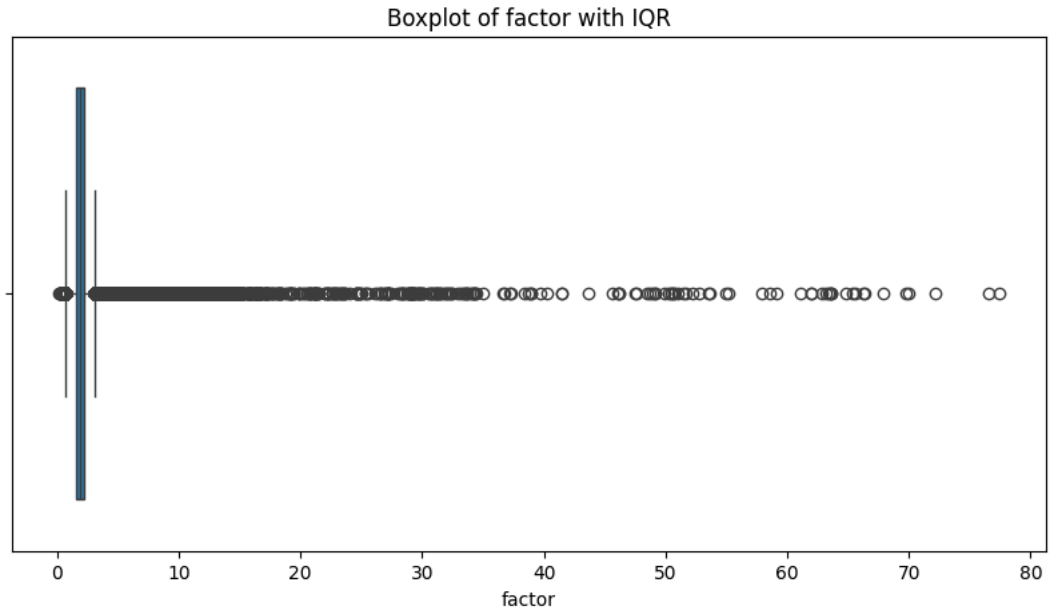



```
Dataframe without outliers for osrm_distance
            data         trip_creation_time  \
0       training 2018-09-20 02:35:36.476840
1       training 2018-09-20 02:35:36.476840
2       training 2018-09-20 02:35:36.476840
```

```
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
...          ...                        ...
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231
144866  training 2018-09-20 16:24:28.436231

                                   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
...                                                   ...        ...
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144866  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

       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)
...                   ...                           ...
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144866       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                   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  ...                 NaT
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
...                           ...  ...                 ...
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19
144866 2018-09-20 16:24:28.436231  ...                 NaT

        actual_distance_to_destination                actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                           ...
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158
144866                       70.039010 1970-01-01 00:00:00.000000426

                          osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...           ...       ...
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245
144866 1970-01-01 00:00:00.000000095       88.7319  4.484211

                 segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                           ...
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000030 1970-01-01 00:00:00.000000024
```

```
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027
144866 1970-01-01 00:00:00.000000268 1970-01-01 00:00:00.000000009

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630
144866                 8.8088       29.777778

[127051 rows x 24 columns]
factor
Q1:  1.6042638714304989
Q3:  2.213483146067416
IQR:  0.6092192746369172
Lower Bound:  0.6904349594751231
Upper Bound:  3.1273120580227918
Outliers in factor (IQR Method):
            data         trip_creation_time  \
15      training 2018-09-14 15:42:46.437249
16      training 2018-09-14 15:42:46.437249
76          test 2018-09-27 14:16:14.819357
77          test 2018-09-27 14:16:14.819357
78          test 2018-09-27 14:16:14.819357
...          ...                        ...
144634  training 2018-09-18 00:34:51.206487
144658  training 2018-09-12 00:14:49.629525
144848  training 2018-09-14 18:45:34.164734
144854  training 2018-09-17 11:35:28.838714
144866  training 2018-09-20 16:24:28.436231

                                     route_schedule_uuid route_type  \
15      thanos::sroute:a16bfa03-3462-4bce-9c82-5784c7d...    Carting
16      thanos::sroute:a16bfa03-3462-4bce-9c82-5784c7d...    Carting
76      thanos::sroute:1283977c-889a-4e96-b632-5ba1a69...    Carting
77      thanos::sroute:1283977c-889a-4e96-b632-5ba1a69...    Carting
78      thanos::sroute:1283977c-889a-4e96-b632-5ba1a69...    Carting
...                                                   ...        ...
144634  thanos::sroute:387e7ab9-2237-48b1-af49-2508ce2...        FTL
144658  thanos::sroute:b62ab3ed-c60b-47d2-8c91-fe62135...    Carting
144848  thanos::sroute:40b6dc9c-faa1-4753-8bc8-ac8c3e0...    Carting
144854  thanos::sroute:d8f74492-4484-412a-887a-61c8e6b...    Carting
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting

                     trip_uuid source_center  \
15      trip-153693976643699843  IND400011AAA
16      trip-153693976643699843  IND400011AAA
76      trip-153805777481903807  IND600056AAB
77      trip-153805777481903807  IND600056AAB
78      trip-153805777481903807  IND600056AAB
...                         ...           ...
144634  trip-153723089120625505  IND151001AAA
144658  trip-153671128962918389  IND302014AAB
144848  trip-153695073416451616  IND400102AAB
144854  trip-153718412883843340  IND600056AAB
144866  trip-153746066843555182  IND131028AAB

                            source_name destination_center  \
15           LowerParel_CP (Maharashtra)       IND400072AAD
16           LowerParel_CP (Maharashtra)       IND400072AAD
76       MAA_Poonamallee_HB (Tamil Nadu)       IND600032AAB
77       MAA_Poonamallee_HB (Tamil Nadu)       IND600032AAB
78       MAA_Poonamallee_HB (Tamil Nadu)       IND600032AAB
...                                  ...                ...
144634             Bhatinda_DPC (Punjab)       IND151302AAA
144658    Jaipur_Central_I_7 (Rajasthan)       IND302026AAA
144848  Mumbai_Jogeshwri_L (Maharashtra)       IND421302AAG
144854   MAA_Poonamallee_HB (Tamil Nadu)       IND600032AAB
144866        Sonipat_Kundli_H (Haryana)       IND000000ACB

                        destination_name              od_start_time  ...  \
15       Mumbai_Chndivli_PC (Maharashtra) 2018-09-14 15:42:46.437249  ...
16       Mumbai_Chndivli_PC (Maharashtra) 2018-09-14 15:42:46.437249  ...
76               Chennai_Hub (Tamil Nadu) 2018-09-27 14:16:14.819357  ...
77               Chennai_Hub (Tamil Nadu) 2018-09-27 14:16:14.819357  ...
78               Chennai_Hub (Tamil Nadu) 2018-09-27 14:16:14.819357  ...
...                                   ...                        ...  ...
144634    TalwandiSabo_Wardno3_D (Punjab) 2018-09-18 00:34:51.206487  ...
144658     Jaipur_Bhankrot_DC (Rajasthan) 2018-09-12 00:14:49.629525  ...
144848  Bhiwandi_Mankoli_HB (Maharashtra) 2018-09-14 18:45:34.164734  ...
144854           Chennai_Hub (Tamil Nadu) 2018-09-17 11:35:28.838714  ...
144866      Gurgaon_Bilaspur_HB (Haryana) 2018-09-20 16:24:28.436231  ...

          cutoff_timestamp  actual_distance_to_destination  \
15     2018-09-14 16:29:54                        9.355852
16     2018-09-14 16:15:53                       16.431273
76     2018-09-27 15:59:36                        9.285856
77     2018-09-27 15:32:36                       18.094962
78     2018-09-27 15:17:37                       27.106207
...                    ...                             ...
144634 2018-09-18 01:28:11                       22.479429
```

```
144634 2018-09-18 01:28:11                        23.479429
144658 2018-09-12 00:35:21                        14.202707
144848 2018-09-14 19:19:54                        23.034042
144854 2018-09-17 12:57:20                         9.169115
144866                 NaT                        70.039010

                        actual_time                     osrm_time  \
15     1970-01-01 00:00:00.000000046 1970-01-01 00:00:00.000000011
16     1970-01-01 00:00:00.000000060 1970-01-01 00:00:00.000000016
76     1970-01-01 00:00:00.000000042 1970-01-01 00:00:00.000000010
77     1970-01-01 00:00:00.000000069 1970-01-01 00:00:00.000000018
78     1970-01-01 00:00:00.000000084 1970-01-01 00:00:00.000000026
...                              ...                           ...
144634 1970-01-01 00:00:00.000000025 1970-01-01 00:00:00.000000040
144658 1970-01-01 00:00:00.000000086 1970-01-01 00:00:00.000000018
144848 1970-01-01 00:00:00.000000344 1970-01-01 00:00:00.000000031
144854 1970-01-01 00:00:00.000000032 1970-01-01 00:00:00.000000010
144866 1970-01-01 00:00:00.000000426 1970-01-01 00:00:00.000000095

       osrm_distance     factor           segment_actual_time  \
15           11.4344   4.181818 1970-01-01 00:00:00.000000046
16           18.7941   3.750000 1970-01-01 00:00:00.000000014
76            9.9365   4.200000 1970-01-01 00:00:00.000000042
77           19.8934   3.833333 1970-01-01 00:00:00.000000027
78           29.6956   3.230769 1970-01-01 00:00:00.000000014
...              ...        ...                           ...
144634       55.3429   0.625000 1970-01-01 00:00:00.000000025
144658       18.9291   4.777778 1970-01-01 00:00:00.000000064
144848       33.7957  11.096774 1970-01-01 00:00:00.000000302
144854        9.9543   3.200000 1970-01-01 00:00:00.000000032
144866       88.7319   4.484211 1970-01-01 00:00:00.000000268

                   segment_osrm_time  segment_osrm_distance  segment_factor
15     1970-01-01 00:00:00.000000011                11.4344        4.181818
16     1970-01-01 00:00:00.000000005                 7.3597        2.800000
76     1970-01-01 00:00:00.000000010                 9.9365        4.200000
77     1970-01-01 00:00:00.000000007                 9.9569        3.857143
78     1970-01-01 00:00:00.000000008                 9.8021        1.750000
...                              ...                    ...             ...
144634 1970-01-01 00:00:00.000000040                55.3429        0.625000
144658 1970-01-01 00:00:00.000000006                 6.3938       10.666667
144848 1970-01-01 00:00:00.000000011                10.4932       27.454545
144854 1970-01-01 00:00:00.000000010                 9.9543        3.200000
144866 1970-01-01 00:00:00.000000009                 8.8088       29.777778

[11429 rows x 24 columns]
```

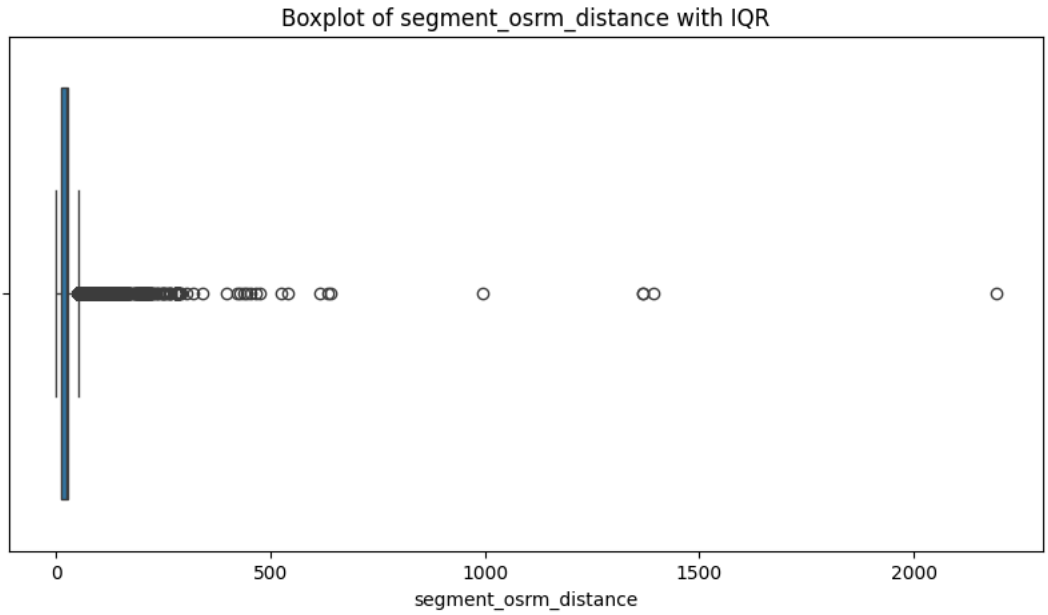



```
Dataframe without outliers for factor
            data         trip_creation_time  \
0       training 2018-09-20 02:35:36.476840
1       training 2018-09-20 02:35:36.476840
2       training 2018-09-20 02:35:36.476840
3       training 2018-09-20 02:35:36.476840
4       training 2018-09-20 02:35:36.476840
...          ...                        ...
144861  training 2018-09-20 16:24:28.436231
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231

                                     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
...                                                   ...         ...
144861  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
```

```
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144861  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

       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)
...                   ...                           ...
144861       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                    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  ...                 NaT
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
...                            ...  ...                 ...
144861 2018-09-20 16:24:28.436231  ... 2018-09-20 22:09:21
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19

        actual_distance_to_destination                  actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                            ...
144861                       37.406091 1970-01-01 00:00:00.000000081
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158

                          osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...           ...       ...
144861 1970-01-01 00:00:00.000000062       60.1136  1.306452
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245

                  segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                            ...
144861 1970-01-01 00:00:00.000000011 1970-01-01 00:00:00.000000012
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144861                 9.5478        0.916667
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630

[133438 rows x 24 columns]
segment_osrm_distance
Q1:  12.0701
```

```
Q3:  27.81325
IQR:  15.74315
Lower Bound:  -11.544625
Upper Bound:  51.427975
Outliers in segment_osrm_distance (IQR Method):
            data         trip_creation_time  \
34      training 2018-09-13 20:44:19.424489
157     training 2018-09-15 23:58:16.827101
158     training 2018-09-15 23:58:16.827101
214     training 2018-09-17 00:14:20.789064
316     training 2018-09-24 02:57:00.372087
...           ...                        ...
144774      test 2018-10-01 18:17:37.047270
144802  training 2018-09-26 14:05:52.096792
144829  training 2018-09-26 19:50:29.657378
144837  training 2018-09-26 19:50:29.657378
144845  training 2018-09-26 19:50:29.657378

                                     route_schedule_uuid route_type  \
34      thanos::sroute:76951383-1608-44e4-a284-46d92e8...        FTL
157     thanos::sroute:fb308c0f-ea3a-48ef-a6c3-4776341...        FTL
158     thanos::sroute:fb308c0f-ea3a-48ef-a6c3-4776341...        FTL
214     thanos::sroute:d0cd2cb2-ce42-4103-b999-f8899e9...        FTL
316     thanos::sroute:8f136f2a-7552-4c91-acfa-ff555d1...        FTL
...                                                   ...        ...
144774  thanos::sroute:4029a8a2-6c74-4b7e-a6d8-f9e069f...        FTL
144802  thanos::sroute:f7de4133-6bd9-4367-a7f7-ab190b6...        FTL
144829  thanos::sroute:f6d1ba62-76a2-4dba-83ec-3ac0803...        FTL
144837  thanos::sroute:f6d1ba62-76a2-4dba-83ec-3ac0803...        FTL
144845  thanos::sroute:f6d1ba62-76a2-4dba-83ec-3ac0803...        FTL

                     trip_uuid source_center  \
34      trip-153687145942424248  IND560099AAB
157     trip-153705589682687518  IND206001AAA
158     trip-153705589682687518  IND206001AAA
214     trip-153714326609873773  IND431517AAB
316     trip-153775782037183132  IND413002AAA
...                         ...           ...
144774  trip-153841785704702048  IND000000ACB
144802  trip-153797075209653066  IND411033AAA
144829  trip-153799142965708367  IND454001AAA
144837  trip-153799142965708367  IND457226AAA
144845  trip-153799142965708367  IND457226AAA

                              source_name destination_center  \
34       Bengaluru_Bomsndra_HB (Karnataka)       IND683511AAA
157      Etawah_MhraChng_D (Uttar Pradesh)       IND000000ACB
158      Etawah_MhraChng_D (Uttar Pradesh)       IND000000ACB
214     Ambajogai_BnsllNgr_D (Maharashtra)       IND411033AAA
316      Solapur_Central_I_2 (Maharashtra)       IND501359AAE
...                                    ...                ...
144774       Gurgaon_Bilaspur_HB (Haryana)       IND562132AAA
144802       Pune_Tathawde_H (Maharashtra)       IND413002AAA
144829    Dhar_Trimurti_D (Madhya Pradesh)       IND457001AAA
144837   Jaora_RtlamNka_D (Madhya Pradesh)       IND382430AAB
144845   Jaora_RtlamNka_D (Madhya Pradesh)       IND382430AAB

                           destination_name              od_start_time  ...  \
34                 Aluva_Peedika_H (Kerala) 2018-09-13 23:59:56.061158  ...
157           Gurgaon_Bilaspur_HB (Haryana) 2018-09-17 02:46:57.274441  ...
158           Gurgaon_Bilaspur_HB (Haryana) 2018-09-17 02:46:57.274441  ...
214           Pune_Tathawde_H (Maharashtra) 2018-09-17 05:21:32.158856  ...
316        Hyderabad_Shamshbd_H (Telangana) 2018-09-24 12:46:04.801166  ...
...                                     ...                        ...  ...
144774     Bangalore_Nelmngla_H (Karnataka) 2018-10-02 09:02:19.284969  ...
144802    Solapur_Central_I_2 (Maharashtra) 2018-09-26 14:05:52.096792  ...
144829  Ratlam_Khjurwli_DC (Madhya Pradesh) 2018-09-27 02:48:14.315366  ...
144837         Ahmedabad_East_H_1 (Gujarat) 2018-09-27 06:55:50.265761  ...
144845         Ahmedabad_East_H_1 (Gujarat) 2018-09-27 06:55:50.265761  ...

          cutoff_timestamp  actual_distance_to_destination  \
34     2018-09-14 01:17:22                      331.652400
157    2018-09-17 07:48:35                      133.359623
158    2018-09-17 06:40:28                      154.799934
214    2018-09-17 09:39:31                      178.372265
316    2018-09-24 18:08:03                       88.484048
...                    ...                             ...
144774 2018-10-03 04:20:13                     1127.477441
144802 2018-09-26 21:16:13                       44.140543
144829 2018-09-27 04:00:15                       45.081415
144837 2018-09-27 13:46:26                       88.970987
144845 2018-09-27 08:18:22                      264.468105

                        actual_time                     osrm_time  \
34     1970-01-01 00:00:00.000000738 1970-01-01 00:00:00.000000433
157    1970-01-01 00:00:00.000000183 1970-01-01 00:00:00.000000178
158    1970-01-01 00:00:00.000000252 1970-01-01 00:00:00.000000173
214    1970-01-01 00:00:00.000000378 1970-01-01 00:00:00.000000155
316    1970-01-01 00:00:00.000000112 1970-01-01 00:00:00.000000130
...                              ...                           ...
144774 1970-01-01 00:00:00.000001788 1970-01-01 00:00:00.000001036
144802 1970-01-01 00:00:00.000000054 1970-01-01 00:00:00.000000053
144829 1970-01-01 00:00:00.000000054 1970-01-01 00:00:00.000000050
144837 1970-01-01 00:00:00.000000156 1970-01-01 00:00:00.000000082
144845 1970-01-01 00:00:00.000000484 1970-01-01 00:00:00.000000290

       osrm_distance    factor           segment_actual_time  \
```

```
34          472.5899  1.704388 1970-01-01 00:00:00.000000094
157         205.8009  1.028090 1970-01-01 00:00:00.000000037
158         201.7528  1.456647 1970-01-01 00:00:00.000000068
214         212.7156  2.438710 1970-01-01 00:00:00.000000029
316         138.9323  0.861538 1970-01-01 00:00:00.000000028
...              ...       ...                           ...
144774     1374.1431  1.725869 1970-01-01 00:00:00.000000072
144802       68.0716  1.018868 1970-01-01 00:00:00.000000022
144829       47.4345  1.080000 1970-01-01 00:00:00.000000027
144837      115.8559  1.902439 1970-01-01 00:00:00.000000034
144845      387.9870  1.668966 1970-01-01 00:00:00.000000066

                   segment_osrm_time  segment_osrm_distance  segment_factor
34     1970-01-01 00:00:00.000000070                72.5561        1.342857
157    1970-01-01 00:00:00.000000081                79.6653        0.456790
158    1970-01-01 00:00:00.000000081                82.4127        0.839506
214    1970-01-01 00:00:00.000000044                52.7136        0.659091
316    1970-01-01 00:00:00.000000075                60.0755        0.373333
...                              ...                    ...             ...
144774 1970-01-01 00:00:00.000000042                60.6393        1.714286
144802 1970-01-01 00:00:00.000000048                61.0445        0.458333
144829 1970-01-01 00:00:00.000000074                70.0436        0.364865
144837 1970-01-01 00:00:00.000000042                60.4795        0.809524
144845 1970-01-01 00:00:00.000000054                55.6993        1.222222

[4315 rows x 24 columns]
```

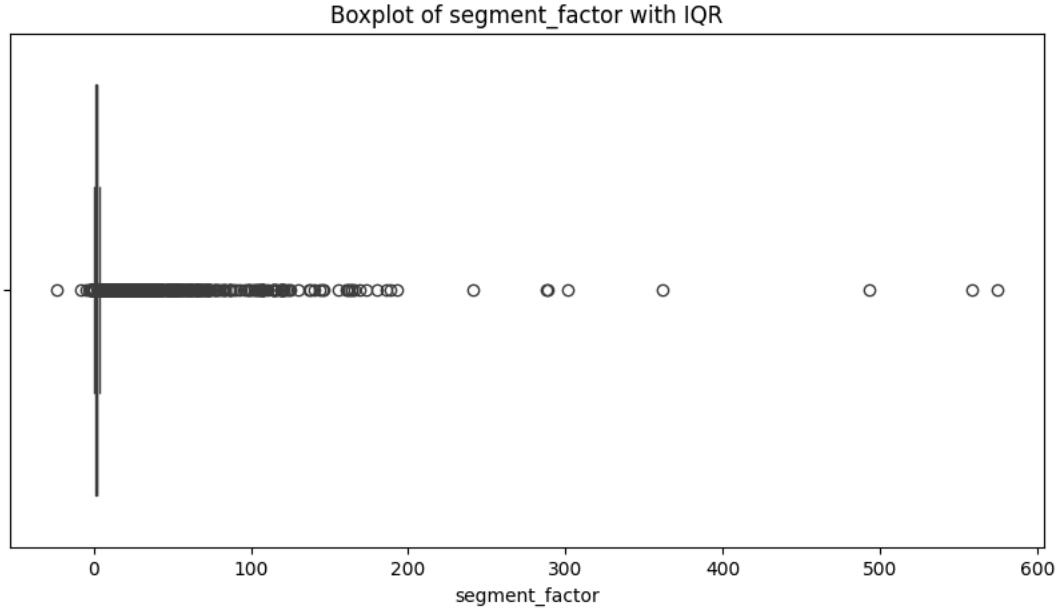



```
Dataframe without outliers for segment_osrm_distance
            data         trip_creation_time  \
0       training 2018-09-20 02:35:36.476840
1       training 2018-09-20 02:35:36.476840
2       training 2018-09-20 02:35:36.476840
3       training 2018-09-20 02:35:36.476840
4       training 2018-09-20 02:35:36.476840
...          ...                        ...
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231
144866  training 2018-09-20 16:24:28.436231

                                   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
...                                                   ...        ...
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144866  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

        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)
...                   ...                           ...
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144866       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                    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  ...                 NaT
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
...                           ...  ...                 ...
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19
144866 2018-09-20 16:24:28.436231  ...                 NaT

        actual_distance_to_destination                   actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                            ...
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158
144866                       70.039010 1970-01-01 00:00:00.000000426

                           osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...           ...       ...
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245
144866 1970-01-01 00:00:00.000000095       88.7319  4.484211

                 segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                            ...
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027
144866 1970-01-01 00:00:00.000000268 1970-01-01 00:00:00.000000009

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630
144866                 8.8088       29.777778

[140552 rows x 24 columns]
segment_factor
Q1:  1.3478260869565215
Q3:  2.25
IQR:  0.9021739130434785
Lower Bound:  -0.005434782608696231
Upper Bound:  3.6032608695652177
Outliers in segment_factor (IQR Method):
            data         trip_creation_time  \
6       training 2018-09-20 02:35:36.476840
9       training 2018-09-20 02:35:36.476840
15      training 2018-09-14 15:42:46.437249
47      training 2018-09-12 01:33:48.711350
54      training 2018-09-12 01:33:48.711350
...          ...                       ...
144846  training 2018-09-26 19:50:29.657378
144848  training 2018-09-14 18:45:34.164734
144852  training 2018-09-22 11:30:41.399439
144853  training 2018-09-22 11:30:41.399439
144866  training 2018-09-20 16:24:28.436231

                                   route_schedule_uuid route_type  \
6       thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...    Carting
```

```
9       thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...    Carting
15      thanos::sroute:a16bfa03-3462-4bce-9c82-5784c7d...    Carting
47      thanos::sroute:5f7d8d49-ae14-430e-9333-37361e1...    Carting
54      thanos::sroute:5f7d8d49-ae14-430e-9333-37361e1...    Carting
...                                                   ...        ...
144846  thanos::sroute:f6d1ba62-76a2-4dba-83ec-3ac0803...        FTL
144848  thanos::sroute:40b6dc9c-faa1-4753-8bc8-ac8c3e0...    Carting
144852  thanos::sroute:d81088e2-9ccd-43e9-9260-3e85633...        FTL
144853  thanos::sroute:d81088e2-9ccd-43e9-9260-3e85633...        FTL
144866  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting

                     trip_uuid source_center  \
6       trip-153741093647649320  IND388620AAB
9       trip-153741093647649320  IND388620AAB
15      trip-153693976643699843  IND400011AAA
47      trip-153671602871109556  IND362001AAA
54      trip-153671602871109556  IND362720AAA
...                         ...           ...
144846  trip-153799142965708367  IND457226AAA
144848  trip-153695073416451616  IND400102AAB
144852  trip-153761584139918815  IND421302AAG
144853  trip-153761584139918815  IND421302AAG
144866  trip-153746066843555182  IND131028AAB

                             source_name destination_center  \
6           Khambhat_MotvdDPP_D (Gujarat)       IND388320AAA
9           Khambhat_MotvdDPP_D (Gujarat)       IND388320AAA
15            LowerParel_CP (Maharashtra)       IND400072AAD
47                 Junagadh_DPC (Gujarat)       IND362220AAA
54           Kodinar_NCplxDPP_D (Gujarat)       IND362560AAA
...                                    ...                ...
144846  Jaora_RtlamNka_D (Madhya Pradesh)       IND382430AAB
144848   Mumbai_Jogeshwri_L (Maharashtra)       IND421302AAG
144852  Bhiwandi_Mankoli_HB (Maharashtra)       IND411033AAA
144853  Bhiwandi_Mankoli_HB (Maharashtra)       IND411033AAA
144866         Sonipat_Kundli_H (Haryana)       IND000000ACB

                        destination_name              od_start_time  ...  \
6              Anand_Vaghasi_IP (Gujarat) 2018-09-20 04:47:45.236797  ...
9              Anand_Vaghasi_IP (Gujarat) 2018-09-20 04:47:45.236797  ...
15       Mumbai_Chndivli_PC (Maharashtra) 2018-09-14 15:42:46.437249  ...
47           Junagadh_keshod_DC (Gujarat) 2018-09-12 01:33:48.711350  ...
54              Una_Mamlatdr_DC (Gujarat) 2018-09-12 06:12:09.579013  ...
...                                   ...                        ...  ...
144846       Ahmedabad_East_H_1 (Gujarat) 2018-09-27 06:55:50.265761  ...
144848  Bhiwandi_Mankoli_HB (Maharashtra) 2018-09-14 18:45:34.164734  ...
144852      Pune_Tathawde_H (Maharashtra) 2018-09-22 11:30:41.399439  ...
144853      Pune_Tathawde_H (Maharashtra) 2018-09-22 11:30:41.399439  ...
144866      Gurgaon_Bilaspur_HB (Haryana) 2018-09-20 16:24:28.436231  ...

          cutoff_timestamp  actual_distance_to_destination  \
6      2018-09-20 05:47:29                       18.045481
9      2018-09-20 04:49:20                       43.595802
15     2018-09-14 16:29:54                        9.355852
47     2018-09-12 01:51:56                       28.340661
54                     NaT                       37.240741
...                    ...                            ...
144846 2018-09-27 08:18:22                      265.367032
144848 2018-09-14 19:19:54                       23.034042
144852 2018-09-22 17:57:24                       89.220979
144853 2018-09-22 16:25:28                      100.562078
144866                 NaT                       70.039010

                        actual_time                    osrm_time  \
6      1970-01-01 00:00:00.000000044 1970-01-01 00:00:00.000000017
9      1970-01-01 00:00:00.000000102 1970-01-01 00:00:00.000000045
15     1970-01-01 00:00:00.000000046 1970-01-01 00:00:00.000000011
47     1970-01-01 00:00:00.000000043 1970-01-01 00:00:00.000000022
54     1970-01-01 00:00:00.000000059 1970-01-01 00:00:00.000000028
...                              ...                           ...
144846 1970-01-01 00:00:00.000000484 1970-01-01 00:00:00.000000290
144848 1970-01-01 00:00:00.000000344 1970-01-01 00:00:00.000000031
144852 1970-01-01 00:00:00.000000198 1970-01-01 00:00:00.000000080
144853 1970-01-01 00:00:00.000000289 1970-01-01 00:00:00.000000095
144866 1970-01-01 00:00:00.000000426 1970-01-01 00:00:00.000000095

       osrm_distance     factor           segment_actual_time  \
6            21.2890   2.588235 1970-01-01 00:00:00.000000028
9            53.2334   2.266667 1970-01-01 00:00:00.000000026
15           11.4344   4.181818 1970-01-01 00:00:00.000000046
47           31.1283   1.954545 1970-01-01 00:00:00.000000002
54           39.5496   2.107143 1970-01-01 00:00:00.000000000
...              ...        ...                            ...
144846      387.9870   1.668966 1970-01-01 00:00:00.000000000
144848       33.7957  11.096774 1970-01-01 00:00:00.000000302
144852      109.6824   2.475000 1970-01-01 00:00:00.000000053
144853      129.1588   3.042105 1970-01-01 00:00:00.000000091
144866       88.7319   4.484211 1970-01-01 00:00:00.000000268

                   segment_osrm_time  segment_osrm_distance  segment_factor
6      1970-01-01 00:00:00.000000006                 9.1719        4.666667
9      1970-01-01 00:00:00.000000006                 6.0434        4.333333
15     1970-01-01 00:00:00.000000011                11.4344        4.181818
47     1970-01-01 00:00:00.000000000                 1.2782       -1.000000
54     1970-01-01 00:00:00.000000000                 0.0000       -1.000000
...                              ...                    ...             ...
144846 1970-01-01 00:00:00.000000000                 0.0000       -1.000000
```

```
144848 1970-01-01 00:00:00.000000011                10.4932        27.454545
144852 1970-01-01 00:00:00.000000014                18.2524         3.785714
144853 1970-01-01 00:00:00.000000014                19.4764         6.500000
144866 1970-01-01 00:00:00.000000009                 8.8088        29.777778

[13976 rows x 24 columns]
```




```
Dataframe without outliers for segment_factor
            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
...          ...                        ...
144861  training 2018-09-20 16:24:28.436231
144862  training 2018-09-20 16:24:28.436231
144863  training 2018-09-20 16:24:28.436231
144864  training 2018-09-20 16:24:28.436231
144865  training 2018-09-20 16:24:28.436231

                                     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
...                                                   ...        ...
144861  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144862  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144863  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144864  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    Carting
144865  thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...    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)
...                         ...           ...                         ...
144861  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144862  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144863  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144864  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)
144865  trip-153746066843555182  IND131028AAB  Sonipat_Kundli_H (Haryana)

       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)
...                   ...                           ...
144861       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144862       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144863       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144864       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)
144865       IND000000ACB  Gurgaon_Bilaspur_HB (Haryana)

                    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  ...                 NaT
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
...                           ...  ...                 ...
144861 2018-09-20 16:24:28.436231  ... 2018-09-20 22:09:21
144862 2018-09-20 16:24:28.436231  ... 2018-09-20 21:57:20
144863 2018-09-20 16:24:28.436231  ... 2018-09-20 21:31:18
144864 2018-09-20 16:24:28.436231  ... 2018-09-20 21:11:18
```

```
144865 2018-09-20 16:24:28.436231  ... 2018-09-20 20:53:19

        actual_distance_to_destination                actual_time  \
0                            10.435660 1970-01-01 00:00:00.000000014
1                            18.936842 1970-01-01 00:00:00.000000024
2                            27.637279 1970-01-01 00:00:00.000000040
3                            36.118028 1970-01-01 00:00:00.000000062
4                            39.386040 1970-01-01 00:00:00.000000068
...                                ...                           ...
144861                       37.406091 1970-01-01 00:00:00.000000081
144862                       45.258278 1970-01-01 00:00:00.000000094
144863                       54.092531 1970-01-01 00:00:00.000000120
144864                       66.163591 1970-01-01 00:00:00.000000140
144865                       73.680667 1970-01-01 00:00:00.000000158

                          osrm_time osrm_distance    factor  \
0      1970-01-01 00:00:00.000000011       11.9653  1.272727
1      1970-01-01 00:00:00.000000020       21.7243  1.200000
2      1970-01-01 00:00:00.000000028       32.5395  1.428571
3      1970-01-01 00:00:00.000000040       45.5620  1.550000
4      1970-01-01 00:00:00.000000044       54.2181  1.545455
...                              ...           ...       ...
144861 1970-01-01 00:00:00.000000062       60.1136  1.306452
144862 1970-01-01 00:00:00.000000060       67.9280  1.566667
144863 1970-01-01 00:00:00.000000076       85.6829  1.578947
144864 1970-01-01 00:00:00.000000088       97.0933  1.590909
144865 1970-01-01 00:00:00.000000098      111.2709  1.612245

                  segment_actual_time             segment_osrm_time  \
0      1970-01-01 00:00:00.000000014 1970-01-01 00:00:00.000000011
1      1970-01-01 00:00:00.000000010 1970-01-01 00:00:00.000000009
2      1970-01-01 00:00:00.000000016 1970-01-01 00:00:00.000000007
3      1970-01-01 00:00:00.000000021 1970-01-01 00:00:00.000000012
4      1970-01-01 00:00:00.000000006 1970-01-01 00:00:00.000000005
...                              ...                           ...
144861 1970-01-01 00:00:00.000000011 1970-01-01 00:00:00.000000012
144862 1970-01-01 00:00:00.000000012 1970-01-01 00:00:00.000000012
144863 1970-01-01 00:00:00.000000026 1970-01-01 00:00:00.000000021
144864 1970-01-01 00:00:00.000000020 1970-01-01 00:00:00.000000034
144865 1970-01-01 00:00:00.000000017 1970-01-01 00:00:00.000000027

        segment_osrm_distance  segment_factor
0                     11.9653        1.272727
1                      9.7590        1.111111
2                     10.8152        2.285714
3                     13.0224        1.750000
4                      3.9153        1.200000
...                       ...             ...
144861                 9.5478        0.916667
144862                 8.1858        1.000000
144863                17.3725        1.238095
144864                20.7053        0.588235
144865                18.8885        0.629630

[130891 rows x 24 columns]
```

🔍 Insights using the Interquartile Range (IQR) method:

1) start_scan_to_end_scan:

Q1: 161.0, Q3: 1634.0, IQR: 1473.0

Lower Bound: -2048.5, Upper Bound: 3843.5

Outliers Detected

2) cutoff_factor:

Q1: 22.0, Q3: 286.0, IQR: 264.0

Lower Bound: -374.0, Upper Bound: 682.0

Outliers Detected

3) actual_distance_to_destination:

Q1: 23.35, Q3: 286.70, IQR: 263.35

Lower Bound: -371.67 , Upper Bound: 681.74

Outliers Detected

4) osrm_distance:

Q1: 29.91, Q3: 343.27, IQR: 313.27

Lower Bound: -440.003, Upper Bound: 813.111

Outliers Detected

5) factor:

Q1: 1.60, Q3: 2.21, IQR: 0.80

Lower Bound: 0.69, Upper Bound: 3.13

Outliers Detected

6)segment_osrm_distance:

Q1: 12.07, Q3: 27.81, IQR: 15.74

Lower Bound: -11.54, Upper Bound:51.43

Outliers Detected

segment_factor:

Q1: 1.34, Q3: 2.25, IQR: 0.90

Lower Bound: -0.005, Upper Bound: 3.60

Outliers Detected

```
# Function to detect outliers using z-score method
def detect_outliers_zscore(data, threshold=3):
    outliers = {}
    for column in data.columns:
        if data[column].dtype in ['int64', 'float64']:
            z_scores = (data[column] - data[column].mean()) / data[column].std()
            outliers[column] = data[np.abs(z_scores) > threshold][column]
    return outliers

# Detect outliers for each column
outliers = detect_outliers_zscore(df)

# Print the outliers
for column, outlier_data in outliers.items():
    print(f"Outliers in column '{column}':")
    print(outlier_data)
    print("\n")
```

```
                ...
144025    11.600000
144034     7.666667
144232     7.454545
144606    15.123288
144848    11.096774
Name: factor, Length: 1317, dtype: float64


Outliers in column 'segment_osrm_distance':
157        79.6653
158        82.4127
317        76.7188
378       128.9020
757       107.2922
                ...
144560     98.7910
144588    120.6447
144606     98.5604
144717    168.3641
144723    139.8276
Name: segment_osrm_distance, Length: 1509, dtype: float64


Outliers in column 'segment_factor':
769        21.183673
813        18.000000
942        24.428571
1258       22.000000
1323       41.333333
                ...
143144     26.428571
143410     32.000000
143563    137.000000
144848     27.454545
144866     29.777778
Name: segment_factor, Length: 790, dtype: float64
```

Insights from Outlier Detection Using Z score Method

1) Columns without Outliers: segment_factor: No outliers detected.

2) Columns with Outliers:

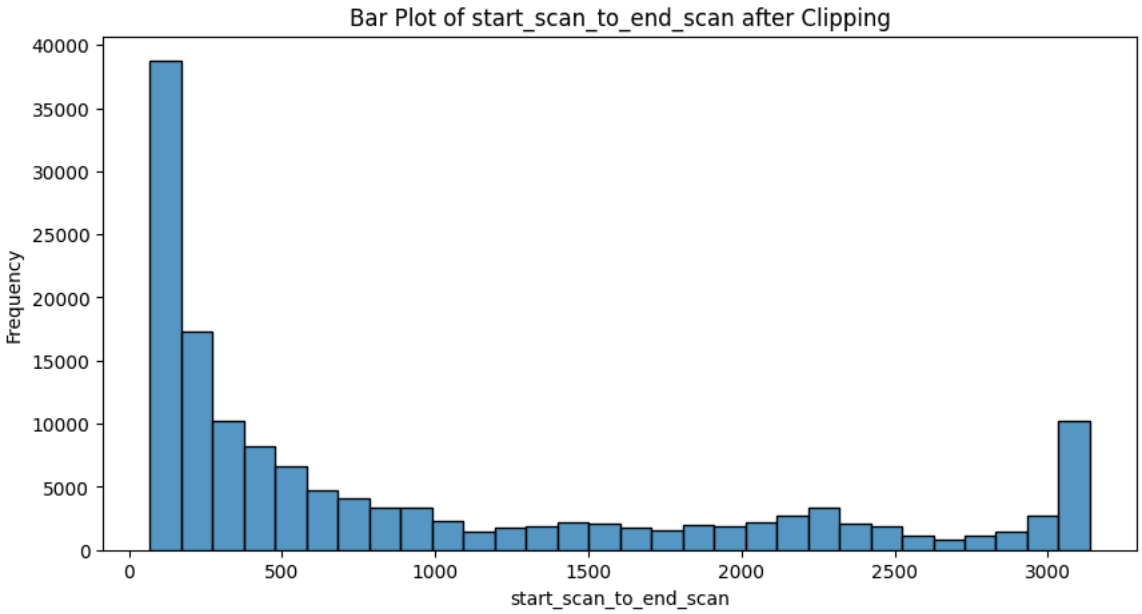
- start_scan_to_end_scan: 350 outliers detected. These represent unusually long durations between scan start and scan end.
- cutoff_factor: 3428 outliers detected. These values indicate extreme deviations in cutoff factors.
- actual_distance_to_destination: 3429 outliers detected. These represent excessively long actual travel distances.
- osrm_distance: 3611 outliers detected. These indicate predicted travel distances that deviate significantly from the norm.
- factor: 1317 outliers detected. These represent unusually high efficiency ratios.
- segment_osrm_distance: 1509 outliers detected. These values indicate extreme deviations in segment travel distances.

Summary:

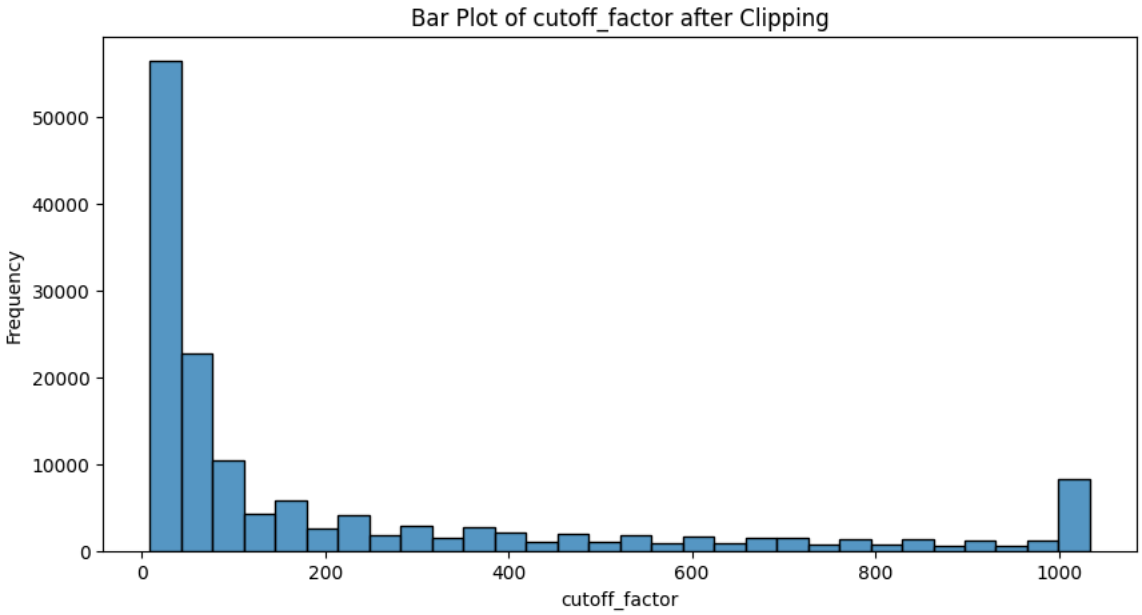
- The Z score method successfully identified outliers in several columns, indicating abnormal values in durations, distances, and efficiency ratios.
-

```
#b) Remove/clip the data between the 5 percentile and 95 percentile

# Clip the data between the 5th percentile and 95th percentile
for column in continuous_columns:
  lower_bound = np.percentile(df[column], 5)
  upper_bound = np.percentile(df[column], 95)
  print(column,"lower bound :",lower_bound)
  print(column,"upper bound :",upper_bound)
  df[column] = np.clip(df[column], lower_bound, upper_bound)
  # Plot the results as bar plots
  plt.figure(figsize=(10, 5))
  sns.histplot(df[column], bins=30, kde=False)
  plt.title(f'Bar Plot of {column} after Clipping')
  plt.xlabel(column)
  plt.ylabel('Frequency')
  plt.show()
```

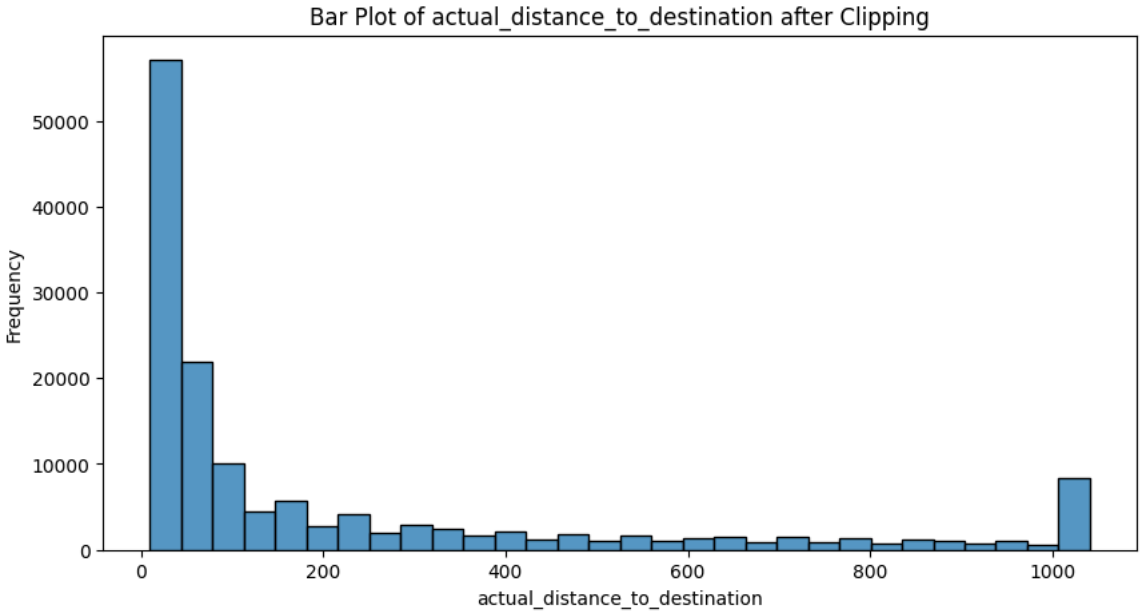
```
start_scan_to_end_scan lower bound : 69.0
start_scan_to_end_scan upper bound : 3137.0
```



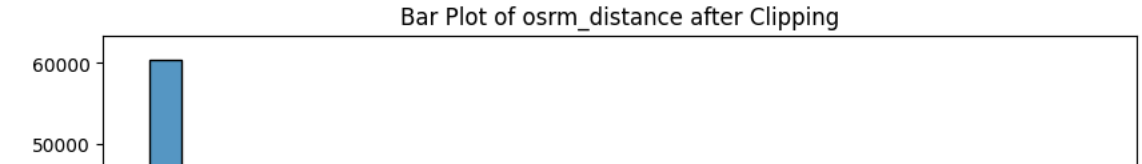

```
cutoff_factor lower bound : 9.0
cutoff_factor upper bound : 1034.0
```

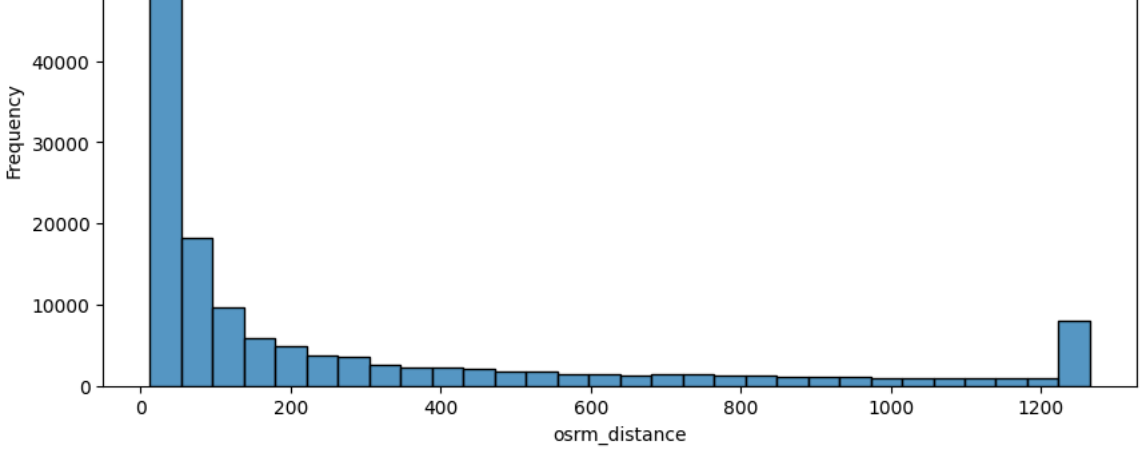



```
actual_distance_to_destination lower bound : 9.696476709044637
actual_distance_to_destination upper bound : 1041.3736855028778
```

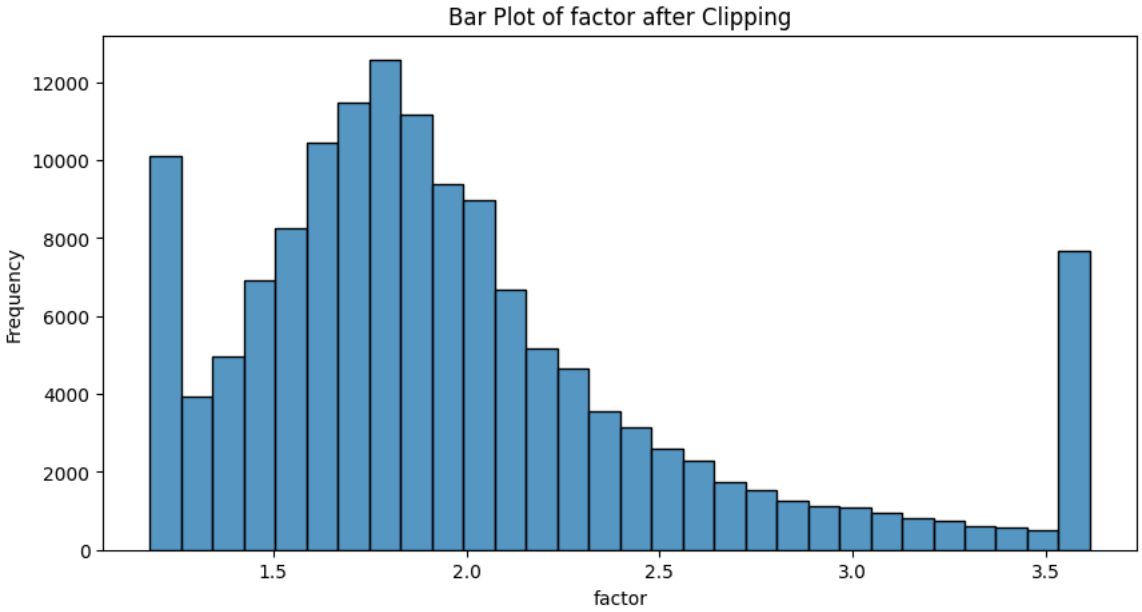



```
osrm_distance lower bound : 12.66556
osrm_distance upper bound : 1265.3359699999996
```

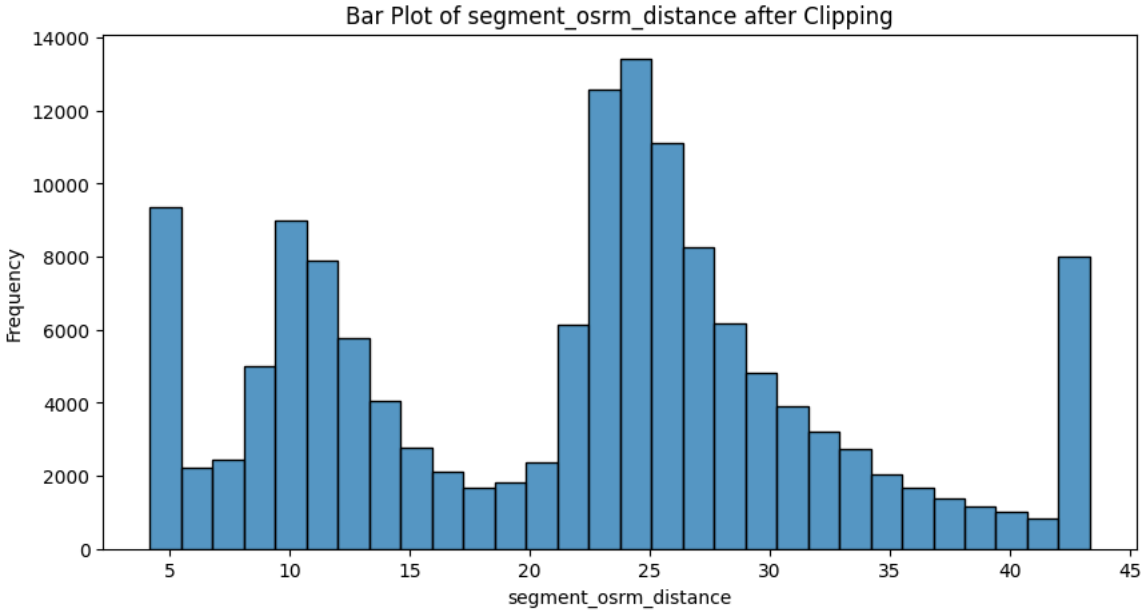




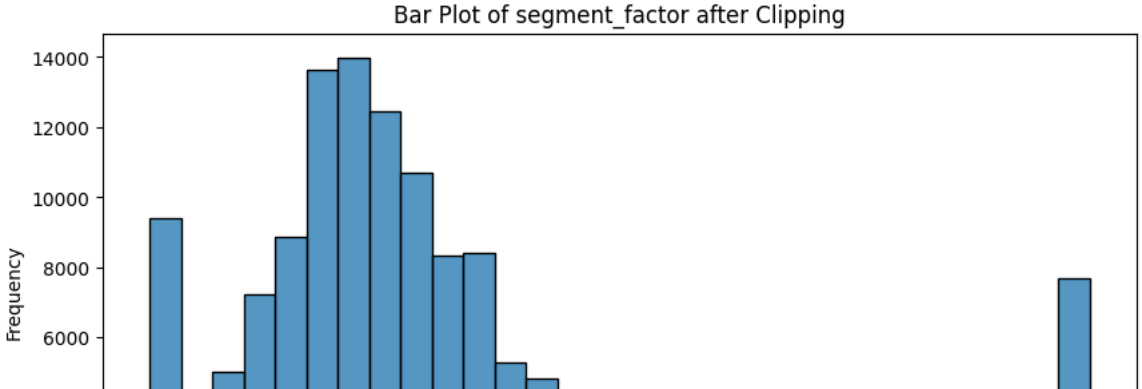

```
factor lower bound : 1.181818181818182
factor upper bound : 3.6127057274522603
```

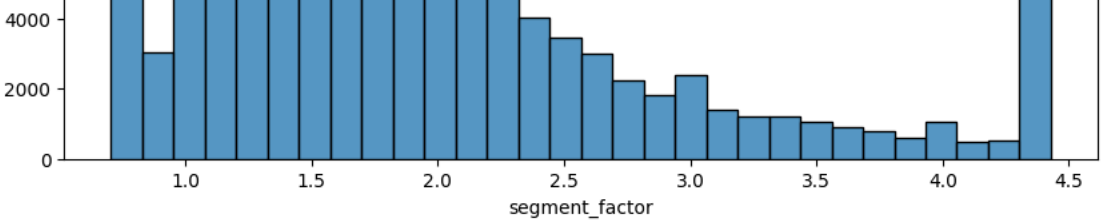



```
segment_osrm_distance lower bound : 4.16886
segment_osrm_distance upper bound : 43.34859999999998
```




```
segment_factor lower bound : 0.7058823529411765
segment_factor upper bound : 4.428571428571429
```




4000
2000
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
segment_factor

Insights from Clipping Data Between the 5th and 95th Percentile

1) Continuous Variables and Clipped Ranges:

a) start_scan_to_end_scan:

- Lower Bound: 69.0
- Upper Bound: 3137.0
- Values below 69.0 or above 3137.0 have been clipped to these limits, ensuring the range excludes extreme durations.

b) cutoff_factor:

- Lower Bound: 9.0
- Upper Bound: 1034.0
- Extreme cutoff values have been adjusted to fall within the clipped range.

c) actual_distance_to_destination:

- Lower Bound: 9.69
- Upper Bound: 1041.37
- Travel distances now adhere to a refined range, reducing noise caused by outliers.

d) osrm_distance:

- Lower Bound: 12.66
- Upper Bound: 1265.33
- Predicted distances beyond this range have been clipped, enhancing distribution consistency.

e) factor:

- Lower Bound: 1.18
- Upper Bound: 3.61
- Efficiency ratios have been streamlined by removing extreme values, supporting accurate analysis.

f) segment_osrm_distance:

- Lower Bound: 4.16
- Upper Bound: 43.16
- Segment distances now align with typical operational spans after clipping.

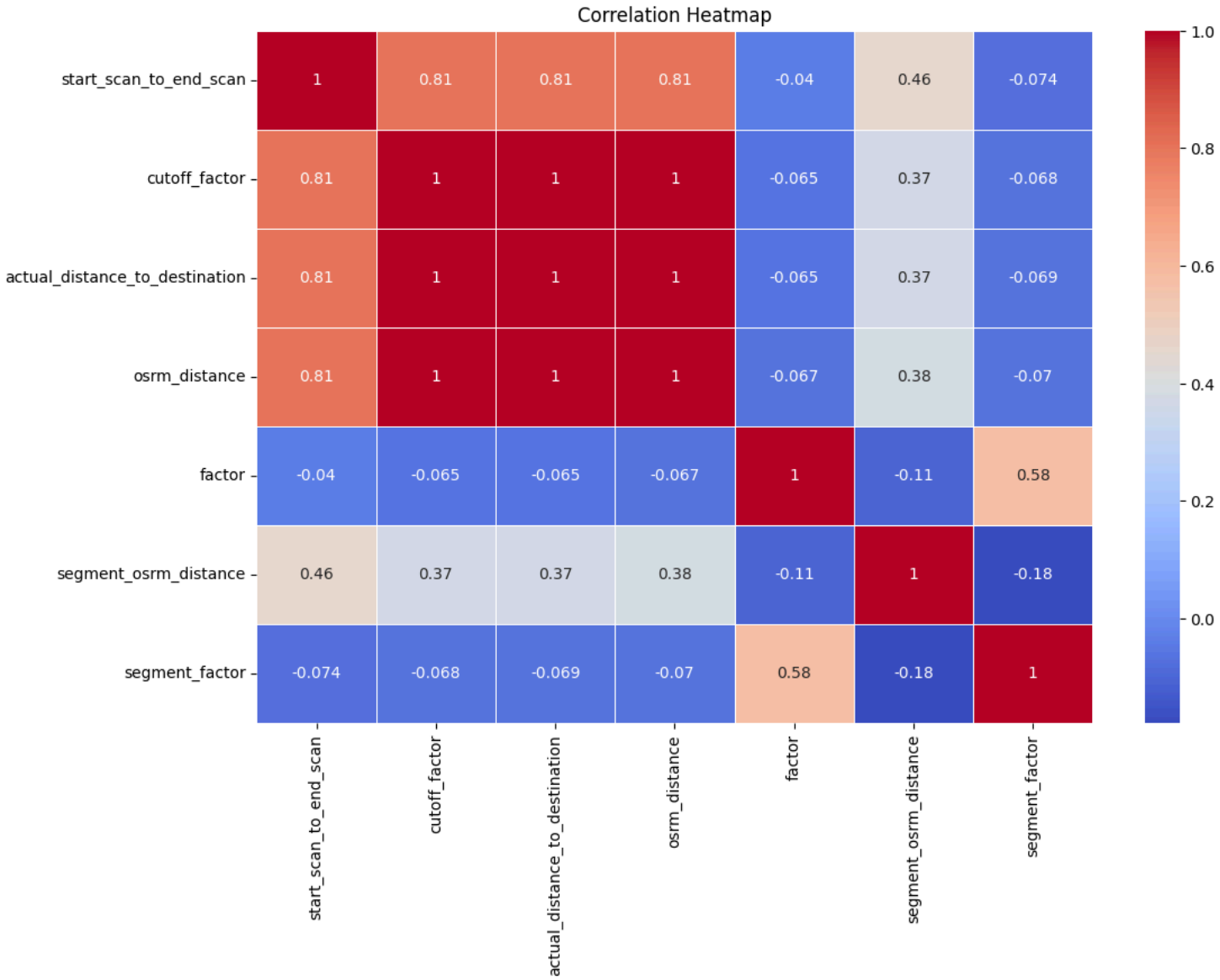
g) segment_factor:

- Lower Bound: 0.70
- Upper Bound: 4.43
- Values fall within standard performance ratios post-clipping.

Summary:

- The clipping process effectively reduces the influence of extreme outliers in all continuous variables, preserving key data trends and improving distribution accuracy.
- Visualized histograms post-clipping display more concentrated and meaningful data ranges, supporting further analysis and modeling.

```
# Filter numerical columns
numerical_df = df.select_dtypes(include=['float64', 'int64'])

# Compute correlation matrix for numerical columns only
corr_matrix = numerical_df.corr()

# Plot the correlation heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation He atmap')
plt.show()
```




```
corr_matrix
```

	start_scan_to_end_scan	cutoff_factor	actual_distance_to_destination	osrm_distance	factor	segment_osrm_distance
start_scan_to_end_scan	1.000000	0.805417	0.805513	0.806598	-0.039727	0.460281
cutoff_factor	0.805417	1.000000	0.999979	0.997496	-0.065141	0.371907
actual_distance_to_destination	0.805513	0.999979	1.000000	0.997544	-0.065369	0.372810
osrm_distance	0.806598	0.997496	0.997544	1.000000	-0.066956	0.379700
factor	-0.039727	-0.065141	-0.065369	-0.066956	1.000000	-0.109693
segment_osrm_distance	0.460281	0.371907	0.372810	0.379700	-0.109693	1.000000
segment_factor	-0.074406	-0.068355	-0.068674	-0.069912	0.581328	-0.177103

Next steps: Generate code with corr_matrix | View recommended plots | New interactive sheet

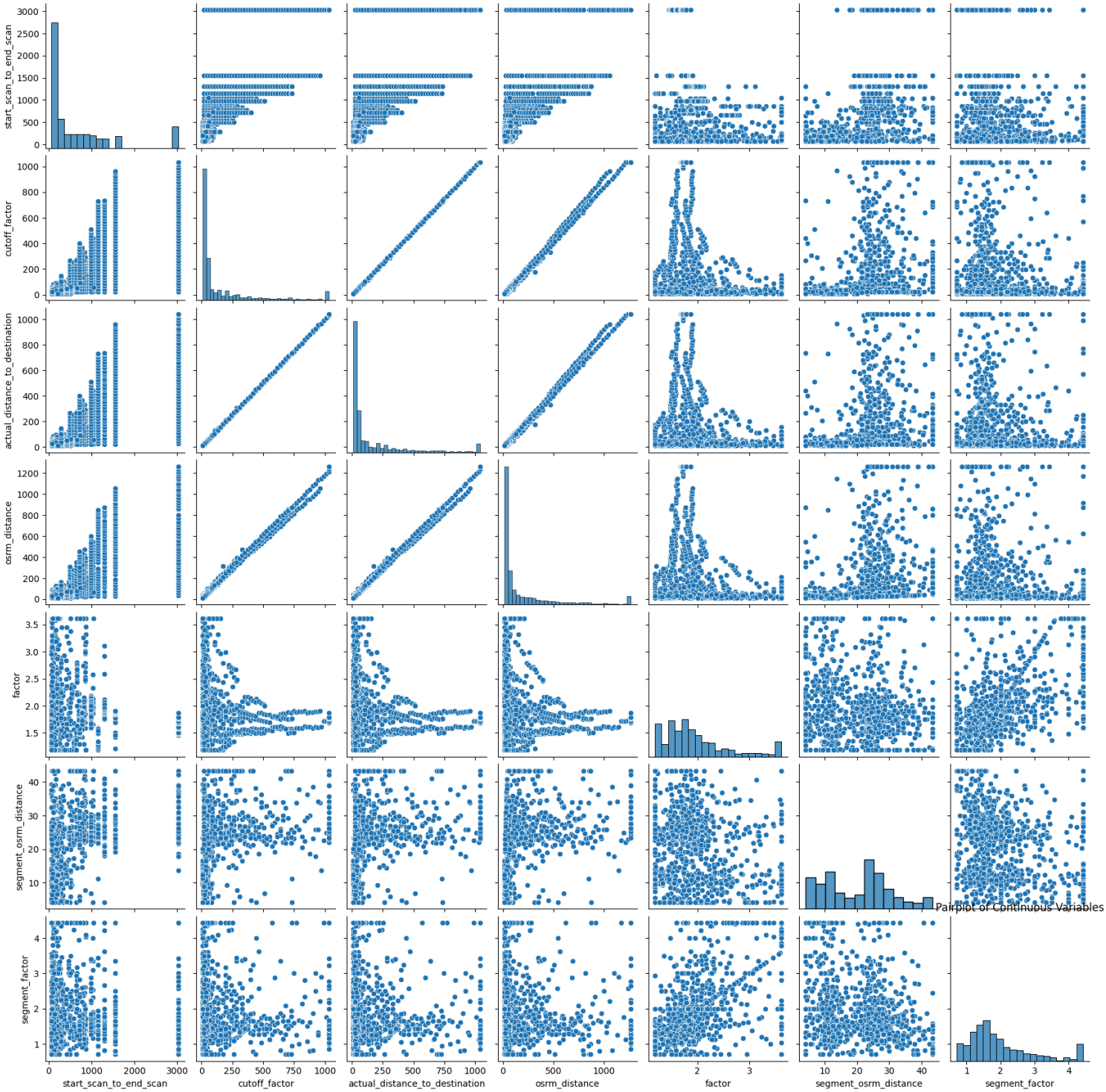
```
# Generate insights from the correlation matrix
def generate_insights(corr_matrix):
    weak_negative_correlations = []
    weak_positive_correlations = []
    moderate_positive_correlations = []
    moderate_negative_correlations = []
    strong_positive_correlations = []
    strong_negative_correlations = []

    for col in corr_matrix.columns:
        for row in corr_matrix.index:
            if col != row:
                correlation = corr_matrix.loc[row, col]
                if -0.3 <= correlation <= 0:
                    weak_negative_correlations.append(f"Weak Negative correlation between {row} and {col}: {correlation:4f}")
                elif 0 < correlation <= 0.3:
                    weak_positive_correlations.append(f"Weak positive correlation between {row} and {col}: {correlation:4f}")
                elif 0.3 < correlation <= 0.7:
                    moderate_positive_correlations.append(f"Moderate positive correlation between {row} and {col}: {correlation:4f}")
                elif -0.7 <= correlation < -0.3:
```

```
                    moderate_negative_correlations.append(f"Moderate negative correlation between {row} and {col}: {correlation:4f}")
                elif correlation > 0.7:
                    strong_positive_correlations.append(f"Strong positive correlation between {row} and {col}: {correlation:4f}")
                elif correlation < -0.7:
                    strong_negative_correlations.append(f"Strong negative correlation between {row} and {col}: {correlation:4f}")

    return weak_negative_correlations,weak_positive_correlations, moderate_positive_correlations, moderate_negative_correlations, strong_positive_corr

# Generate and print insights
weak_negative_correlations,weak_positive_correlations, moderate_positive_correlations, moderate_negative_correlations, strong_positive_correlations, s

print("🔍 INSIGHTS:\n")

print("Weak Positive Correlations:")
if weak_positive_correlations:
    for insight in weak_positive_correlations:
        print(insight)
else:
    print("There is no weak positive correlation")

print("Weak Negative Correlations:")
if weak_negative_correlations:
    for insight in weak_negative_correlations:
        print(insight)
else:
    print("There is no weak negative correlation")
print("\nModerate Positive Correlations:")
if moderate_positive_correlations:
    for insight in moderate_positive_correlations:
        print(insight)
else:
    print("There is no moderate positive correlation")

print("\nModerate Negative Correlations:")
if moderate_negative_correlations:
    for insight in moderate_negative_correlations:
        print(insight)
else:
    print("There is no moderate negative correlation")

print("\nStrong Positive Correlations:")
if strong_positive_correlations:
    for insight in strong_positive_correlations:
        print(insight)
else:
    print("There is no strong positive correlation")

print("\nStrong Negative Correlations:")
if strong_negative_correlations:
    for insight in strong_negative_correlations:
        print(insight)
else:
    print("There is no strong negative correlation")
```

```
🔍 INSIGHTS:

Weak Positive Correlations:
There is no weak positive correlation
Weak Negative Correlations:
Weak Negative correlation between factor and start_scan_to_end_scan: -0.039727
Weak Negative correlation between segment_factor and start_scan_to_end_scan: -0.074406
Weak Negative correlation between factor and cutoff_factor: -0.065141
Weak Negative correlation between segment_factor and cutoff_factor: -0.068355
Weak Negative correlation between factor and actual_distance_to_destination: -0.065369
Weak Negative correlation between segment_factor and actual_distance_to_destination: -0.068674
Weak Negative correlation between factor and osrm_distance: -0.066956
Weak Negative correlation between segment_factor and osrm_distance: -0.069912
Weak Negative correlation between start_scan_to_end_scan and factor: -0.039727
Weak Negative correlation between cutoff_factor and factor: -0.065141
Weak Negative correlation between actual_distance_to_destination and factor: -0.065369
Weak Negative correlation between osrm_distance and factor: -0.066956
Weak Negative correlation between segment_osrm_distance and factor: -0.109693
Weak Negative correlation between factor and segment_osrm_distance: -0.109693
Weak Negative correlation between segment_factor and segment_osrm_distance: -0.177103
Weak Negative correlation between start_scan_to_end_scan and segment_factor: -0.074406
Weak Negative correlation between cutoff_factor and segment_factor: -0.068355
Weak Negative correlation between actual_distance_to_destination and segment_factor: -0.068674
Weak Negative correlation between osrm_distance and segment_factor: -0.069912
Weak Negative correlation between segment_osrm_distance and segment_factor: -0.177103

Moderate Positive Correlations:
Moderate positive correlation between segment_osrm_distance and start_scan_to_end_scan: 0.460281
Moderate positive correlation between segment_osrm_distance and cutoff_factor: 0.371907
Moderate positive correlation between segment_osrm_distance and actual_distance_to_destination: 0.372810
Moderate positive correlation between segment_osrm_distance and osrm_distance: 0.379700
Moderate positive correlation between segment_factor and factor: 0.581328
Moderate positive correlation between start_scan_to_end_scan and segment_osrm_distance: 0.460281
Moderate positive correlation between cutoff_factor and segment_osrm_distance: 0.371907
Moderate positive correlation between actual_distance_to_destination and segment_osrm_distance: 0.372810
Moderate positive correlation between osrm_distance and segment_osrm_distance: 0.379700
Moderate positive correlation between factor and segment_factor: 0.581328
```

```
Moderate Negative Correlations:
There is no moderate negative correlation

Strong Positive Correlations:
Strong positive correlation between cutoff_factor and start_scan_to_end_scan: 0.805417
Strong positive correlation between actual_distance_to_destination and start_scan_to_end_scan: 0.805513
Strong positive correlation between osrm_distance and start_scan_to_end_scan: 0.806598
Strong positive correlation between start_scan_to_end_scan and cutoff_factor: 0.805417
Strong positive correlation between actual_distance_to_destination and cutoff_factor: 0.999979
Strong positive correlation between osrm_distance and cutoff_factor: 0.997496
Strong positive correlation between start_scan_to_end_scan and actual_distance_to_destination: 0.805513
Strong positive correlation between cutoff_factor and actual_distance_to_destination: 0.999979
Strong positive correlation between osrm_distance and actual_distance_to_destination: 0.997544
Strong positive correlation between start_scan_to_end_scan and osrm_distance: 0.806598
Strong positive correlation between cutoff_factor and osrm_distance: 0.997496
Strong positive correlation between actual_distance_to_destination and osrm_distance: 0.997544

Strong Negative Correlations:
There is no strong negative correlation
```

```
# Plot the pairplot for first 1000 rows for clear visibility
df_1000=df.head(1000)
sns.pairplot(df_1000[continuous_columns])
plt.title('Pairplot of Continuous Variables')
plt.show()
```



2. Merging the rows

A) Grouping by segment:

a. Create a unique identifier for different segments of a trip based on the combination of the trip_uuid, source_center, and destination_center and name it as segment_key.

```
# Step 1: Create segment_key
df['segment_key'] = df['trip_uuid'] + '_' + df['source_center'] + '_' + df['destination_center']
df
```

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
...	...	...	...	...	...	...	...	...	
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144864	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144865	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144866	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	

144867 rows × 25 columns

b. You can use inbuilt functions like groupby and aggregations like cumsum() to merge the rows in columns segment_actual_time, segment_osrm_distance, segment_osrm_time based on the segment_key.

This way you'll get new columns named segment_actual_time_sum, segment_osrm_distance_sum, segment_osrm_time_sum.

```
# Step 2: Convert datetime fields to numeric (if calculating differences)
df['segment_actual_time_numeric'] = (df['segment_actual_time'] - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)
df['segment_osrm_time_numeric'] = (df['segment_osrm_time'] - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)

# Step 3:  Group by segment_key and calculate cumulative sums
df['segment_actual_time_sum'] = df.groupby('segment_key')['segment_actual_time_numeric'].cumsum()
df['segment_osrm_distance_sum'] = df.groupby('segment_key')['segment_osrm_distance'].cumsum()
df['segment_osrm_time_sum'] = df.groupby('segment_key')['segment_osrm_time_numeric'].cumsum()
```

```
df
```

Summary: Above code performs two main tasks:

1) Convert datetime to numeric:

Converts datetime fields (segment_actual_time and segment_osrm_time) into numeric values (seconds since epoch time) for calculations.

2) Cumulative sums by group:

Groups data by segment_key and calculates cumulative sums for:

segment_actual_time_numeric into segment_actual_time_sum.

segment_osrm_distance into segment_osrm_distance_sum.

segment_osrm_time_numeric into segment_osrm_time_sum.

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
...	...	...	...	...	...	...	...	...	
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144864	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144865	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144866	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	

144867 rows × 30 columns

2. Aggregating at segment level

```
# Step 2: Create the aggregation dictionary

create_segment_dict = {
    'trip_uuid': 'first',  # Keep the first value
    'source_center': 'first',  # Keep the first value
    'destination_center': 'first',  # Keep the first value
    'segment_actual_time': 'max',  # Use 'max' to get the latest timestamp
    'segment_osrm_time': 'max',  # Use 'max' to get the latest timestamp
    'segment_osrm_distance': 'sum',  # Sum the distances,
    'od_end_time': 'max'  # Use 'max' to get the latest end time

}

# Step 3: Group by segment_key and apply the aggregation
segment_df = df.groupby('segment_key').agg(create_segment_dict).reset_index()

# Step 4: Sort the resulting DataFrame
segment_df = segment_df.sort_values(by=['segment_key', 'od_end_time'], ascending=[True, True])

# Step 5: Verify the output
segment_df
```

	segment_key	trip_uuid	source_center	destination_center	segment_actual_time	segment_osrm_ti
0	trip-153671041653548748_IND209304AAA_IND000000ACB	trip-153671041653548748	IND209304AAA	IND000000ACB	1970-01-01 00:00:00.000000151	1970-01- 00:00:00.0000000
1	trip-153671041653548748_IND462022AAA_IND209304AAA	trip-153671041653548748	IND462022AAA	IND209304AAA	1970-01-01 00:00:00.000000091	1970-01- 00:00:00.0000000
2	trip-153671042288605164_IND561203AAB_IND562101AAA	trip-153671042288605164	IND561203AAB	IND562101AAA	1970-01-01 00:00:00.000000018	1970-01- 00:00:00.0000000
3	trip-153671042288605164_IND572101AAA_IND561203AAB	trip-153671042288605164	IND572101AAA	IND561203AAB	1970-01-01 00:00:00.000000022	1970-01- 00:00:00.0000000
4	trip-153671043369099517_IND000000ACB_IND160002AAC	trip-153671043369099517	IND000000ACB	IND160002AAC	1970-01-01 00:00:00.000000275	1970-01- 00:00:00.0000000
...	...	...	...	...	...	
26363	trip-153861115439069069_IND628204AAA_IND627657AAA	trip-153861115439069069	IND628204AAA	IND627657AAA	1970-01-01 00:00:00.000000018	1970-01- 00:00:00.0000000
26364	trip-153861115439069069_IND628613AAA_IND627005AAA	trip-153861115439069069	IND628613AAA	IND627005AAA	1970-01-01 00:00:00.000000051	1970-01- 00:00:00.0000000
26365	trip-153861115439069069_IND628801AAA_IND628204AAA	trip-153861115439069069	IND628801AAA	IND628204AAA	1970-01-01 00:00:00.000000021	1970-01- 00:00:00.0000000
26366	trip-153861118270144424_IND583119AAA_IND583101AAA	trip-153861118270144424	IND583119AAA	IND583101AAA	1970-01-01 00:00:00.000000188	1970-01- 00:00:00.0000000
26367	trip-153861118270144424_IND583201AAA_IND583119AAA	trip-153861118270144424	IND583201AAA	IND583119AAA	1970-01-01 00:00:00.000000030	1970-01- 00:00:00.0000000

26368 rows × 8 columns

Next steps: Generate code with segment_df | View recommended plots | New interactive sheet

3. Feature Engineering:

Extract features from the below fields:

1. Calculate time taken between od_start_time and od_end_time and keep it as a feature named od_time_diff_hour. Drop the original columns, if required.
2. Destination Name: Split and extract features out of destination. City-place-code (State)
3. Source Name: Split and extract features out of destination. City-place-code (State)
4. Trip_creation_time: Extract features like month, year, day, etc.

```
# Step 1: Calculate time difference between od_start_time and od_end_time
df['od_time_diff_hour'] = (df['od_end_time'] - df['od_start_time']).dt.total_seconds() / 3600
# Drop original columns if required
df = df.drop(columns=['od_start_time', 'od_end_time'])
```

```
#2. Destination Name: Split and extract features out of destination. City-place-code(State)
# Extract city, place code, and state using regular expression
df[['destination_city', 'destination_place_code', 'destination_state']] = df['destination_name'].str.extract(r'^(.*?)_(.*?) \((.*?)\)$')
```

```
#3. Source Name: Split and extract features out of destination. City-place-code(State)
df[['source_city', 'source_place_code', 'source_state']] = df['source_name'].str.extract(r'^(.*?)_(.*?) \((.*?)\)$')
```

```
#4. Trip_creation_time: Extract features like month, year, day, etc.

# Extract features from Trip_creation_time
df['trip_creation_month'] = df['trip_creation_time'].dt.month  # Extract month
df['trip_creation_year'] = df['trip_creation_time'].dt.year    # Extract year
df['trip_creation_day'] = df['trip_creation_time'].dt.day      # Extract day
df['trip_creation_hour'] = df['trip_creation_time'].dt.hour    # Extract hour (optional)
df['trip_creation_weekday'] = df['trip_creation_time'].dt.weekday  # Extract weekday (0=Monday, 6=Sunday)
```

```
df
```

Customer Insights:

1) Use extracted City, Place-code, and State to analyze geographic trends in trip destinations and sources.

2) Trip Duration Analysis:Utilize the od_time_diff_hour feature to assess average trip durations by region or routeOptimize routes, timing, or pricing strategies based on long or short-duration trips.

3) Time-Based Trends:Extracted features from Trip_creation_time can reveal seasonal, monthly, or daily trends in trip demand.Align promotions, discounts, or service availability with periods of peak demand.

4) Performance Benchmarking:Compare segment_actual_time against segment_osrm_time to measure deviations between planned and actual times.

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
...	...	...	...	...	...	...	...	...	
144859	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144860	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144861	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	

66005 rows × 44 columns

4. In-depth analysis:

5. Grouping and Aggregating at Trip-level

- a. Groups the segment data by the trip_uuid column to focus on aggregating data at the trip level.
- b. Apply suitable aggregation functions like first, last, and sum specified in the create_trip_dict dictionary to calculate summary statistics for each trip.

2. Outlier Detection & Treatment

- a. Find any existing outliers in numerical features.
- b. Visualize the outlier values using Boxplot.
- c. Handle the outliers using the IQR method.

3. Perform one-hot encoding on categorical features.

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

```
'''
1. Grouping and Aggregating at Trip-level
a. Groups the segment data by the trip_uuid column to focus on
aggregating data at the trip level.
b. Apply suitable aggregation functions like first, last, and sum specified in
the create_trip_dict dictionary to calculate summary statistics for each
trip.

'''
# Step 1: Define the aggregation dictionary
create_trip_dict = {
    'segment_actual_time_numeric': 'sum',  # Total actual segment time (numeric) across all segments in a trip
    'segment_osrm_distance': 'sum',       # Total OSRM distance across all segments in a trip
    'segment_osrm_time_numeric': 'sum',   # Total OSRM time (numeric) across all segments in a trip
    'segment_factor': 'mean',             # Average segment factor for the trip
    'start_scan_to_end_scan': 'max',      # Maximum scan-to-end time across segments
    'destination_city': 'last',           # Destination city of the last segment in the trip
    'source_city': 'first',               # Source city of the first segment in the trip

}

# Step 2: Group by trip_uuid and apply the aggregation
trip_level_df = df.groupby('trip_uuid').agg(create_trip_dict).reset_index()

# Step 3: Add calculated features (optional)
trip_level_df['segment_actual_time_hour'] = trip_level_df['segment_actual_time_numeric'] / 3600  # Convert seconds to hours
trip_level_df['segment_osrm_time_hour'] = trip_level_df['segment_osrm_time_numeric'] / 3600  # Convert seconds to hours
```

trip_level_df

	trip_uuid	segment_actual_time_numeric	segment_osrm_distance	segment_osrm_time_numeric	segment_factor	start_scan_to_end_scan
0	trip-153671041653548748	1.548000e-06	1149.60340	1.008000e-06	1.810931	1260.0
1	trip-153671042288605164	1.410000e-07	84.55086	6.500000e-08	2.291887	122.0
2	trip-153671043369099517	3.308000e-06	2479.27480	1.941000e-06	1.719526	3099.0
3	trip-153671046011330457	5.900000e-08	19.87660	1.600000e-08	3.492063	100.0
4	trip-153671052974046625	3.400000e-07	146.79190	1.150000e-07	3.099082	485.0
...	...	...	...	...	...	...
14812	trip-153861095625827784	8.200000e-08	64.85510	6.200000e-08	1.514224	152.0
14813	trip-153861104386292051	2.100000e-08	16.08830	1.100000e-08	1.982143	69.0
14814	trip-153861106442901555	2.810000e-07	109.83042	8.800000e-08	2.359024	248.0
14815	trip-153861115439069069	2.580000e-07	223.53240	2.210000e-07	1.240518	105.0
14816	trip-153861118270144424	2.740000e-07	80.57870	6.700000e-08	2.813550	287.0

14817 rows × 10 columns

Next steps: Generate code with trip_level_df | View recommended plots | New interactive sheet

Insights:

1) Operational Efficiency:

- Trips with high start_scan_to_end_scan values indicate potential delays at checkpoints, highlighting areas for process optimization.
- Variations in segment_osrm_distance suggest differences in trip lengths, requiring targeted route planning for efficiency.

2) Performance Metrics:

- Analyze segment_factor to identify trips with inefficiencies (high values) or optimized performance (low values).
- segment_actual_time_hour and segment_osrm_time_hour can benchmark actual vs. estimated time to improve delivery accuracy.

3) Route Optimization:

- Common city pairs like Chandigarh → Chandigarh suggest intra-city trips, while routes like Gurgaon → Bhopal indicate inter-city deliveries. These need tailored strategies for fleet allocation.

4) Outlier Detection:

Trips with extremely small or large segment_actual_time_numeric and segment_osrm_distance values might indicate data inconsistencies or inefficiencies worth investigating.

5) Customer-Centric Improvements:

- Accurate predictions for segment_osrm_time_hour can manage delivery expectations and improve satisfaction.
- Use frequent route analysis to identify key cities for improved logistics networks.

6) Strategic Focus:

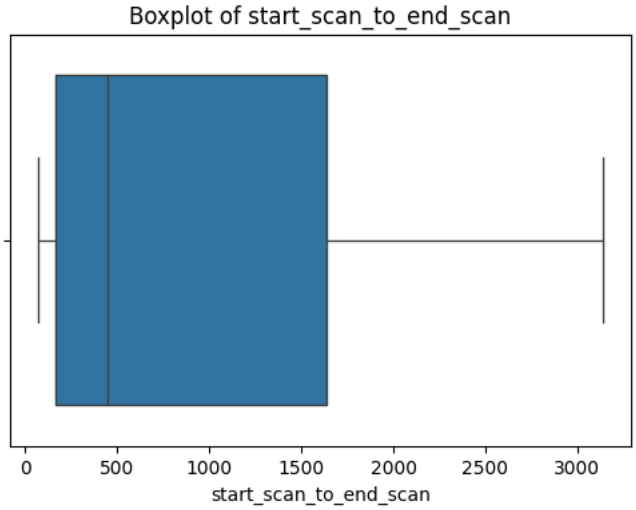
- Normalize and scale metrics like segment_actual_time_numeric to standardize performance evaluation across trips.
- Focus on outlier trips for better data quality and insights.

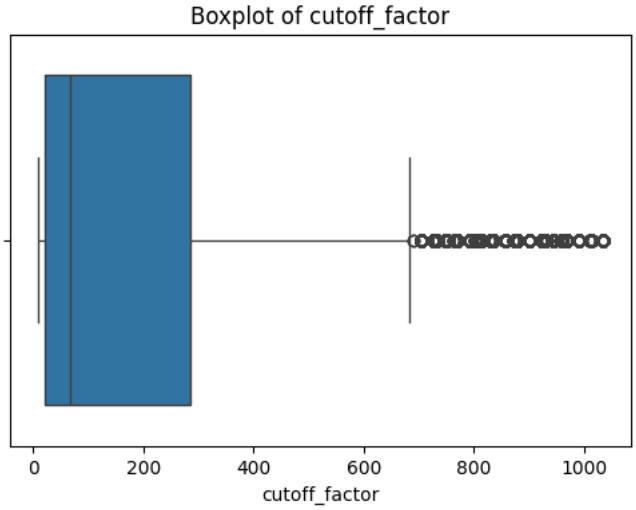
```
'''
2. Outlier Detection & Treatment
a. Find any existing outliers in numerical features.
b. Visualize the outlier values using Boxplot.
c. Handle the outliers using the IQR method.

'''
# List of numerical columns
numerical_cols = [
    'start_scan_to_end_scan',
    'cutoff_factor',
    'actual_distance_to_destination',
    'osrm_distance',
```

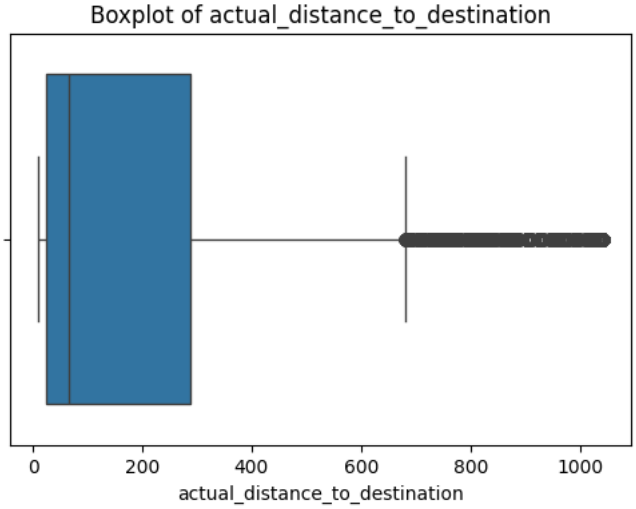
```
    'factor',
    'segment_osrm_distance',
    'segment_factor',
    'segment_osrm_distance_sum',
    'segment_actual_time_sum',
    'segment_osrm_time_sum',
    'segment_actual_time_numeric',
    'segment_osrm_time_numeric',
    'od_time_diff_hour'
]

# Statistical summary of numerical columns
for col in numerical_cols:
    print(f"Statistics for {col}:")
    print(df[col].describe())
    print("\n")
```

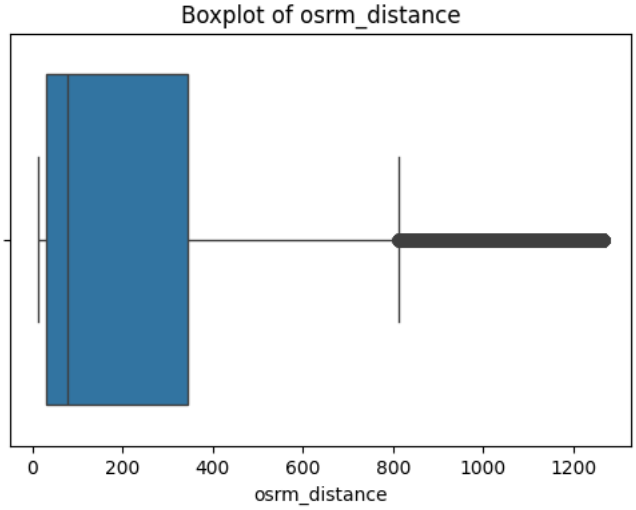
```
Statistics for start_scan_to_end_scan:
count    144867.000000
mean        949.717548
std        1006.962233
min          69.000000
25%         161.000000
50%         449.000000
75%        1634.000000
max        3137.000000
Name: start_scan_to_end_scan, dtype: float64


Statistics for cutoff_factor:
count    144867.000000
mean        219.979271
std         304.629614
min           9.000000
25%          22.000000
50%          66.000000
75%         286.000000
max        1034.000000
Name: cutoff_factor, dtype: float64


Statistics for actual_distance_to_destination:
count    144867.000000
mean        221.429546
std         305.614064
min           9.696477
25%          23.355874
50%          66.126571
75%         286.708875
max        1041.373686
Name: actual_distance_to_destination, dtype: float64


Statistics for osrm_distance:
count    144867.000000
mean        268.329643
std         368.905554
min          12.665560
25%          29.914700
50%          78.525800
75%         343.193250
max        1265.335970
Name: osrm_distance, dtype: float64


Statistics for factor:
count    144867.000000
mean          1.991108
std           0.597523
min           1.181818
25%           1.604264
50%           1.857143
75%           2.213483
max           3.612706
Name: factor, dtype: float64
```

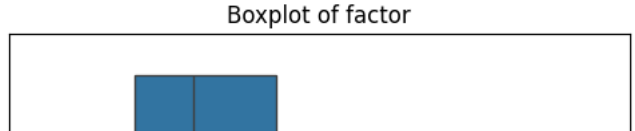
```
# Boxplots for numerical features
for col in numerical_cols:
    plt.figure(figsize=(6, 4))
    sns.boxplot(x=df[col])
    plt.title(f"Boxplot of {col}")
    plt.show()
```

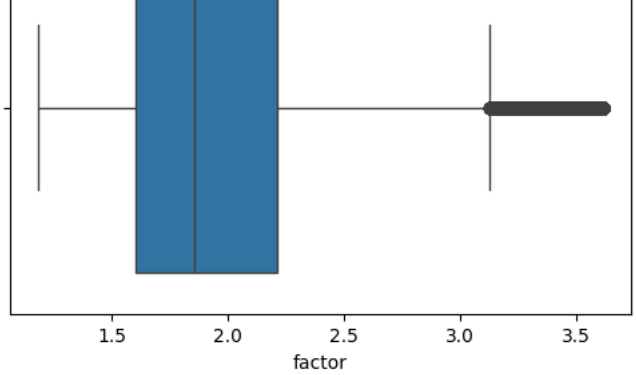

Boxplot of start_scan_to_end_scan
0
500
1000
1500
2000
2500
3000
start_scan_to_end_scan


Boxplot of cutoff_factor
0
200
400
600
800
1000
cutoff_factor

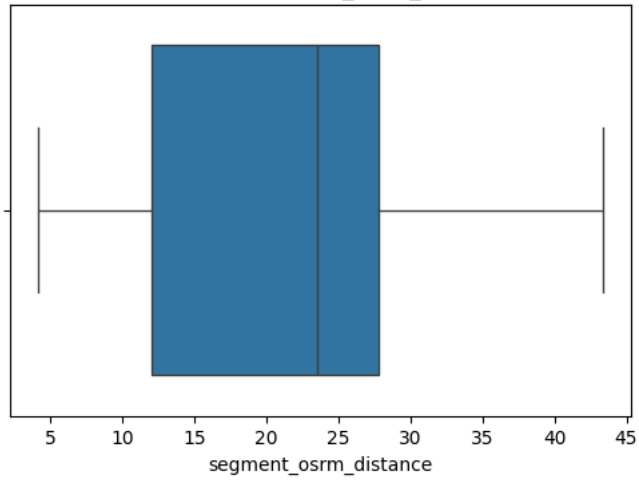

Boxplot of actual_distance_to_destination
0
200
400
600
800
1000
actual_distance_to_destination


Boxplot of osrm_distance
0
200
400
600
800
1000
1200
osrm_distance

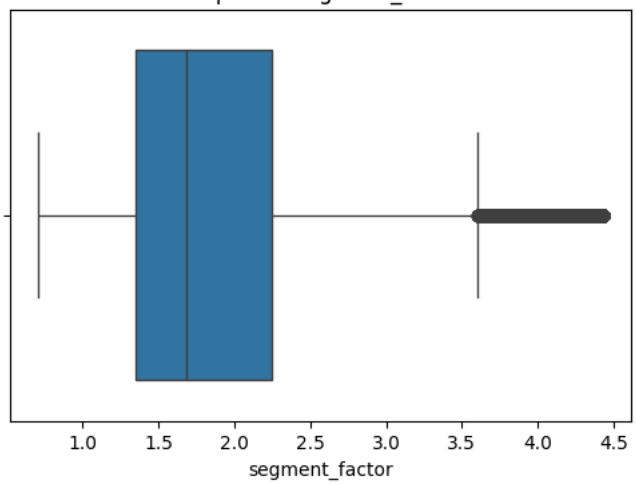

Boxplot of factor



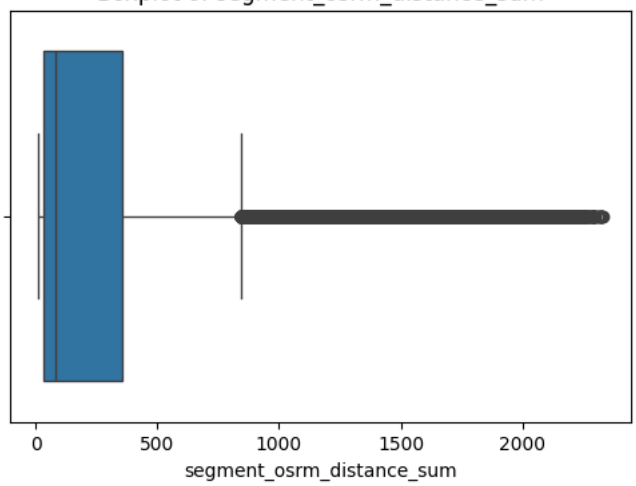
Boxplot of segment_osrm_distance



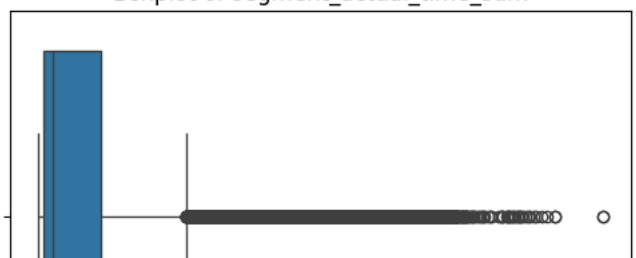
Boxplot of segment_factor

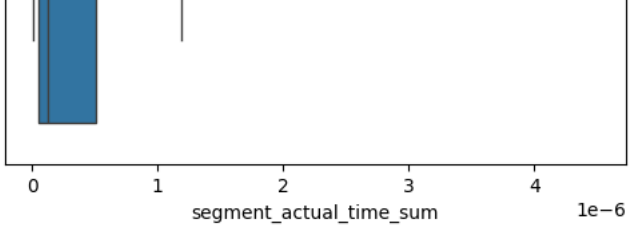


Boxplot of segment_osrm_distance_sum

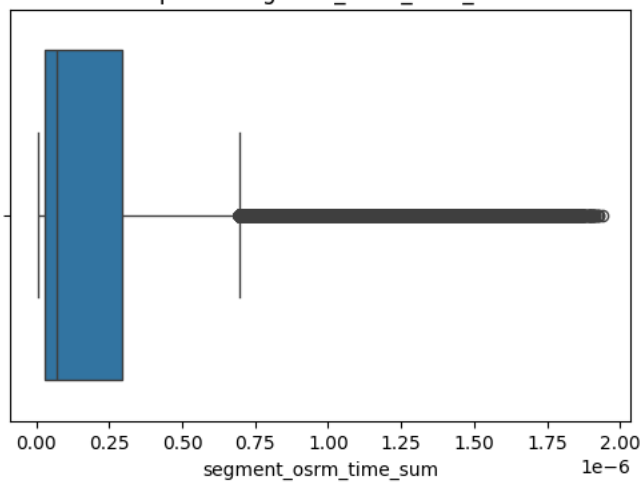


Boxplot of segment_actual_time_sum

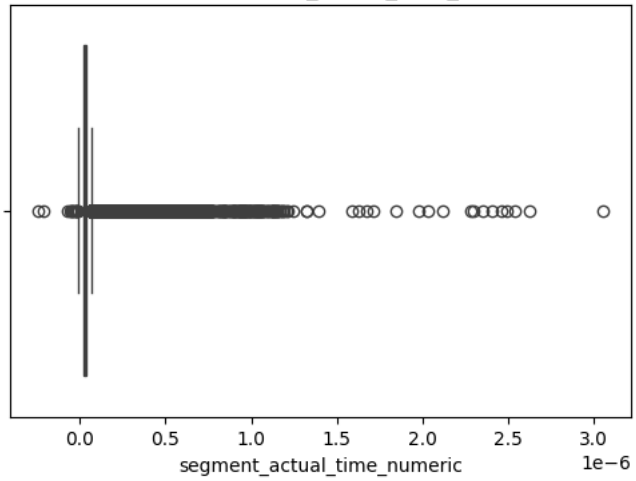




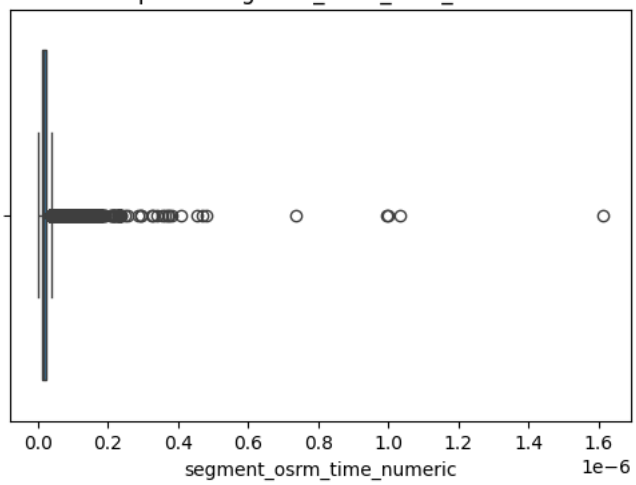
Boxplot of segment_osrm_time_sum



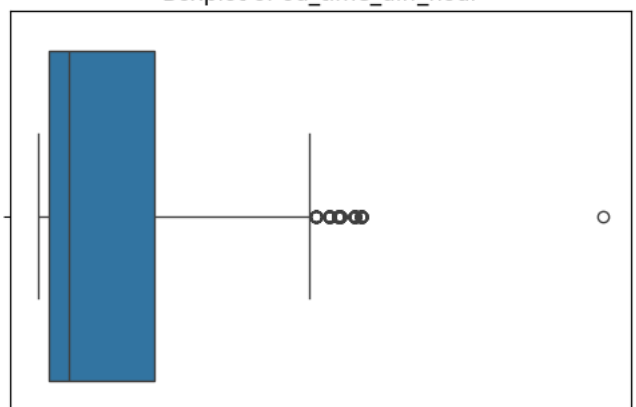
Boxplot of segment_actual_time_numeric

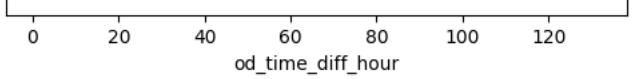


Boxplot of segment_osrm_time_numeric



Boxplot of od_time_diff_hour




0
20
40
60
80
100
120
od_time_diff_hour

```
# Handle outliers using the IQR method
for col in numerical_cols:
    Q1 = df[col].quantile(0.25)  # First quartile (25th percentile)
    Q3 = df[col].quantile(0.75)  # Third quartile (75th percentile)
    IQR = Q3 - Q1  # Interquartile range

    # Define lower and upper bounds
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # Remove rows with outliers
    df = df[(df[col] >= lower_bound) & (df[col] <= upper_bound)]

# Check cleaned dataset
print("Dataset after outlier treatment:")
df
```

```
Dataset after outlier treatment:
```

	data	trip_creation_time	route_schedule_uuid	route_type	trip_uuid	source_center	source_name	destination_center	
0	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
1	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
2	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
3	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
4	training	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	Carting	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Kh
...	...	...	...	...	...	...	...	...	
144859	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144860	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144861	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144862	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	
144863	training	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	Carting	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	

66005 rows × 40 columns

```
''' 3. Perform one-hot encoding on categorical features.'''
# List of categorical columns
categorical_cols = [
    'data','route_type', 'is_cutoff'
]

# Perform one-hot encoding
df_encoded = pd.get_dummies(df, columns=categorical_cols, drop_first=True)

# Display the transformed DataFrame
df_encoded
```

	trip_creation_time	route_schedule_uuid	trip_uuid	source_center	source_name	destination_center	destination_name
0	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambhat_MotvdDPP_D (Gujarat)
1	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambhat_MotvdDPP_D (Gujarat)
2	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambhat_MotvdDPP_D (Gujarat)
3	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambhat_MotvdDPP_D (Gujarat)
4	2018-09-20 02:35:36.476840	thanos::sroute:eb7bfc78-b351-4c0e-a951-fa3d5c3...	trip-153741093647649320	IND388121AAA	Anand_VUNagar_DC (Gujarat)	IND388620AAB	Khambhat_MotvdDPP_D (Gujarat)
...	...	...	...	...	...	...	...
144859	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurgaon_Bilaspur_HB (Haryana)
144860	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurgaon_Bilaspur_HB (Haryana)
144861	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurgaon_Bilaspur_HB (Haryana)
144862	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurgaon_Bilaspur_HB (Haryana)
144863	2018-09-20 16:24:28.436231	thanos::sroute:f0569d2f-4e20-4c31-8542-67b86d5...	trip-153746066843555182	IND131028AAB	Sonipat_Kundli_H (Haryana)	IND000000ACB	Gurgaon_Bilaspur_HB (Haryana)

66005 rows × 40 columns

```
''' Normalize/ Standardize the numerical features using MinMaxScaler or StandardScaler. '''

numerical_cols = [
    'start_scan_to_end_scan',
    'cutoff_factor',
    'actual_distance_to_destination',
    'osrm_distance',
    'factor',
    'segment_osrm_distance',
    'segment_factor',
    'segment_osrm_distance_sum',
    'segment_actual_time_sum',
    'segment_osrm_time_sum',
    'segment_actual_time_numeric',
    'segment_osrm_time_numeric',
    'od_time_diff_hour'
]
```

```
subset_df = df[numerical_cols]


# Initialize MinMaxScaler
min_max_scaler = MinMaxScaler()

# Apply normalization on subset_df
normalized_df = min_max_scaler.fit_transform(subset_df)

# Convert normalized data back to DataFrame
normalized_df = pd.DataFrame(normalized_df, columns=subset_df.columns)

# Display the normalized DataFrame
print("Normalized DataFrame:")

normalized_df.head()
```

Normalized DataFrame:

	start_scan_to_end_scan	cutoff_factor	actual_distance_to_destination	osrm_distance	factor	segment_osrm_distance	segment_factor	segment_o
0	0.022849	0.000000	0.004499	0.000000	0.038523	0.198992	0.206986	
1	0.022849	0.054878	0.056246	0.052994	0.007705	0.142679	0.147971	
2	0.022849	0.109756	0.109205	0.116263	0.104563	0.169637	0.576884	
3	0.022849	0.164634	0.160827	0.192446	0.156019	0.225972	0.381265	
4	0.022849	0.182927	0.180719	0.243084	0.154093	0.000000	0.180430	

Next steps: Generate code with normalized_df | View recommended plots | New interactive sheet

```
# Initialize StandardScaler
standard_scaler = StandardScaler()

# Apply standardization on subset_df
standardized_df = standard_scaler.fit_transform(subset_df)

# Convert standardized data back to DataFrame
standardized_df = pd.DataFrame(standardized_df, columns=subset_df.columns)

# Display the standardized DataFrame
print("Standardized DataFrame:")
standardized_df.head()
```

Standardized DataFrame:

	start_scan_to_end_scan	cutoff_factor	actual_distance_to_destination	osrm_distance	factor	segment_osrm_distance	segment_factor	segment_
0	-0.84228	-0.973486	-0.952945	-0.992968	-1.131090	-0.553565	-0.758938	
1	-0.84228	-0.643185	-0.641522	-0.710922	-1.274929	-0.804773	-1.027298	
2	-0.84228	-0.312885	-0.322800	-0.374188	-0.822865	-0.684515	0.923099	
3	-0.84228	0.017416	-0.012125	0.031270	-0.582706	-0.433205	0.033560	
4	-0.84228	0.127516	0.107592	0.300780	-0.591696	-1.441261	-0.879700	

Next steps: Generate code with standardized_df | View recommended plots | New interactive sheet

Normalized DataFrame Insights:

- The normalized values scale all features to a range of [0, 1].
- Columns such as start_scan_to_end_scan, cutoff_factor, and actual_distance_to_destination reflect relative proportions within the dataset.
- For instance:
- Row 0 shows a low normalized value for start_scan_to_end_scan (0.073492), indicating that its time is significantly smaller relative to the maximum value in this feature.
- Similarly, segment_factor in Row 2 (0.661017) represents a higher performance factor in relation to other rows.

Standardized DataFrame Insights:

- The standardized values have a mean of 0 and a standard deviation of 1.
- Features like osrm_distance and factor now have comparable units, allowing variance-sensitive algorithms like SVM and PCA to perform effectively.
- For example:
- Row 0 shows a very low standardized value for cutoff_factor (-1.190059), which means it is significantly below the mean of this feature.
- Conversely, Row 3 has a high standardized value for actual_distance_to_destination (1.010198), suggesting that this value is higher than average for this feature.

Key Business Insights from Normalized & Standardized Data:

a) Normalized Data:

Useful for proportion-based analysis, such as identifying relative differences in segment_factor across trips.

Helps with algorithms dependent on relative scales, like regression models and deep learning.

b) Standardized Data:

Useful for identifying deviations from average values, such as spotting trips with unusually high segment_osrm_distance.

Ideal for dimensionality reduction techniques like PCA.

Feature-Specific Analysis:

High values in segment_osrm_distance_sum (both normalized and standardized) indicate trips covering longer distances, which might require operational review.

5. Hypothesis Testing/Visual Analysis

```
'''5. Hypothesis Testing:
1. Perform hypothesis testing / visual analysis between :
a. actual_time aggregated value and OSRM time aggregated value.
b. actual_time aggregated value and segment actual time aggregated
value.
c. OSRM distance aggregated value and segment OSRM distance
aggregated value.
d. OSRM time aggregated value and segment OSRM time aggregated
value.

2. Note: Aggregated values are the values you'll get after merging the rows on the
basis of trip_uuid.'''


#Step 1: Aggregate Data by trip_uuid

# Convert datetime columns to numeric values in seconds
df['actual_time_numeric'] = (df['actual_time'] - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)
df['osrm_time_numeric'] = (df['osrm_time'] - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)
df['segment_actual_time_numeric'] = (df['segment_actual_time'] - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)
df['segment_osrm_time_numeric'] = (df['segment_osrm_time'] - pd.Timestamp("1970-01-01")) / pd.Timedelta(seconds=1)

# Grouping by `trip_uuid` to aggregate values
aggregated_df = df.groupby('trip_uuid').agg({
    'actual_time_numeric': 'sum',        # Aggregating actual_time as numeric
    'osrm_time_numeric': 'sum',         # Aggregating OSRM time as numeric
    'osrm_distance': 'sum',             # Aggregating OSRM distance
    'segment_actual_time_numeric': 'sum', # Aggregating segment actual time
    'segment_osrm_distance': 'sum',     # Aggregating segment OSRM distance
    'segment_osrm_time_numeric': 'sum'  # Aggregating segment OSRM time
}).reset_index()

aggregated_df
```

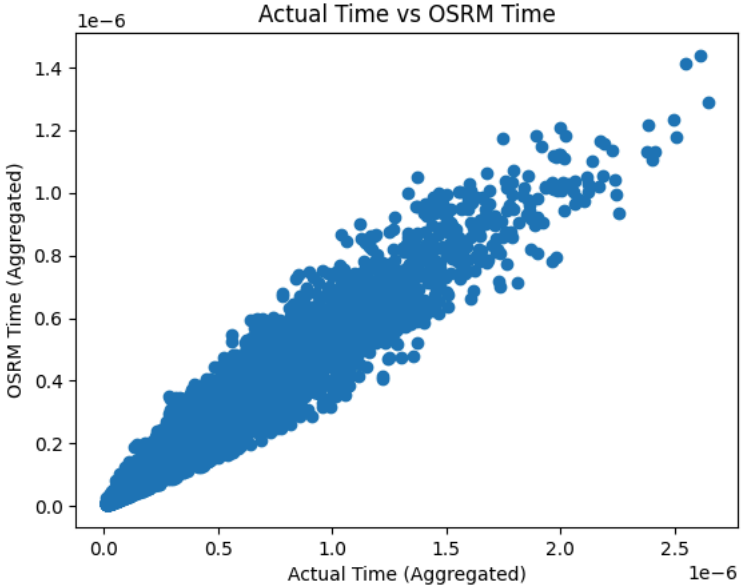
	trip_uuid	actual_time_numeric	osrm_time_numeric	osrm_distance	segment_actual_time_numeric	segment_osrm_distance	segment_osrm
0	trip-153671042288605164	3.030000e-07	1.680000e-07	216.63262	1.210000e-07	80.3820	
1	trip-153671046011330457	2.300000e-08	9.000000e-09	12.66556	2.300000e-08	11.9675	
2	trip-153671052974046625	2.150000e-07	9.000000e-08	119.49960	1.730000e-07	95.6076	
3	trip-153671055416136166	6.100000e-08	2.300000e-08	28.88806	2.900000e-08	17.9341	
4	trip-153671066201138152	2.400000e-08	1.300000e-08	12.66556	2.400000e-08	12.0184	
...	...	...	...	...	...	...	
13106	trip-153861095625827784	1.860000e-07	1.480000e-07	169.07562	8.200000e-08	64.8551	
13107	trip-153861104386292051	3.300000e-08	1.900000e-08	28.75376	2.100000e-08	16.0883	

Next steps: Generate code with aggregated_df | View recommended plots | New interactive sheet

```
#Step 2: Hypothesis Testing
#a. Actual Time vs OSRM Time Aggregated Values
# Perform Pearson correlation
corr_actual_osrm, p_value_actual_osrm = pearsonr(aggregated_df['actual_time_numeric'], aggregated_df['osrm_time_numeric'])

print(f"Correlation Coefficient (Actual vs OSRM Time): {corr_actual_osrm}")
print(f"P-Value: {p_value_actual_osrm}")


if p_value_actual_osrm < 0.05:
    print("The relationship between Actual Time and OSRM Time is statistically significant.")

    if corr_actual_osrm > 0.7:
        print(f"Strong positive correlation (r = {corr_actual_osrm:.2f}).")
    elif corr_actual_osrm > 0.3:
        print(f"Moderate positive correlation (r = {corr_actual_osrm:.2f}).")
```

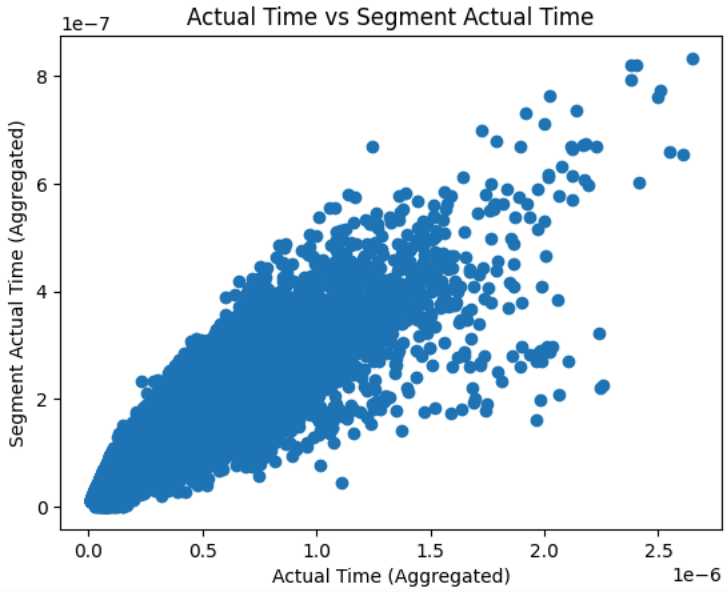
```
    elif corr_actual_osrm > -0.3:
        print(f"Weak or negligible correlation (r = {corr_actual_osrm:.2f}).")
    elif corr_actual_osrm > -0.7:
        print(f"Moderate negative correlation (r = {corr_actual_osrm:.2f}).")
    else:
        print(f"Strong negative correlation (r = {corr_actual_osrm:.2f}).")
else:
    print("The relationship between Actual Time and OSRM Time is not statistically significant.")

print('Visual Analysis')

plt.scatter(aggregated_df['actual_time_numeric'], aggregated_df['osrm_time_numeric'])
plt.title("Actual Time vs OSRM Time")
plt.xlabel("Actual Time (Aggregated)")
plt.ylabel("OSRM Time (Aggregated)")
plt.show()
```

```
Correlation Coefficient (Actual vs OSRM Time): 0.9687185880845388
P-Value: 0.0
The relationship between Actual Time and OSRM Time is statistically significant.
Strong positive correlation (r = 0.97).
Visual Analysis
```



```
#b. Actual Time vs Segment Actual Time Aggregated Values

corr_actual_segment, p_value_actual_segment = pearsonr(aggregated_df['actual_time_numeric'], aggregated_df['segment_actual_time_numeric'])

print(f"Correlation Coefficient (Actual Time vs Segment Actual Time): {corr_actual_segment}")
print(f"P-Value: {p_value_actual_segment}")

if p_value_actual_segment < 0.05:
    print("The relationship between Actual Time and OSRM Time is statistically significant.")

    if corr_actual_segment > 0.7:
        print(f"Strong positive correlation (r = {corr_actual_segment:.2f}).")
    elif corr_actual_segment > 0.3:
        print(f"Moderate positive correlation (r = {corr_actual_segment:.2f}).")
    elif corr_actual_segment > -0.3:
        print(f"Weak or negligible correlation (r = {corr_actual_segment:.2f}).")
    elif corr_actual_segment > -0.7:
        print(f"Moderate negative correlation (r = {corr_actual_segment:.2f}).")
    else:
        print(f"Strong negative correlation (r = {corr_actual_segment:.2f}).")
else:
    print("The relationship between Actual Time and OSRM Time is not statistically significant.")


print('Visual Analysis')

plt.scatter(aggregated_df['actual_time_numeric'], aggregated_df['segment_actual_time_numeric'])
plt.title("Actual Time vs Segment Actual Time")
plt.xlabel("Actual Time (Aggregated)")
plt.ylabel("Segment Actual Time (Aggregated)")
plt.show()
```

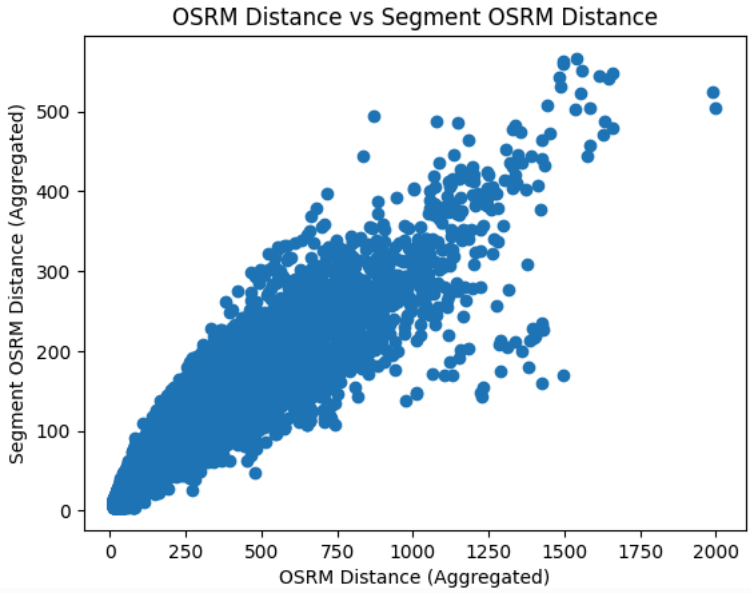
```
Correlation Coefficient (Actual Time vs Segment Actual Time): 0.8973117790681082
P-Value: 0.0
The relationship between Actual Time and OSRM Time is statistically significant.
Strong positive correlation (r = 0.90).
Visual Analysis
```




```
#c) OSRM distance aggregated value and segment OSRM distance aggregated value.

corr_osrm_distance, p_value_osrm_distance = pearsonr(aggregated_df['osrm_distance'], aggregated_df['segment_osrm_distance'])

print(f"Correlation Coefficient (OSRM Distance vs Segment OSRM Distance): {corr_osrm_distance}")
print(f"P-Value: {p_value_osrm_distance}")

if p_value_osrm_distance < 0.05:
    print("The relationship between Actual Time and OSRM Time is statistically significant.")

    if corr_osrm_distance > 0.7:
        print(f"Strong positive correlation (r = {corr_osrm_distance:.2f}).")
    elif corr_osrm_distance > 0.3:
        print(f"Moderate positive correlation (r = {corr_osrm_distance:.2f}).")
    elif corr_osrm_distance > -0.3:
        print(f"Weak or negligible correlation (r = {corr_osrm_distance:.2f}).")
    elif corr_osrm_distance > -0.7:
        print(f"Moderate negative correlation (r = {corr_osrm_distance:.2f}).")
    else:
        print(f"Strong negative correlation (r = {corr_osrm_distance:.2f}).")
else:
    print("The relationship between Actual Time and OSRM Time is not statistically significant.")
```

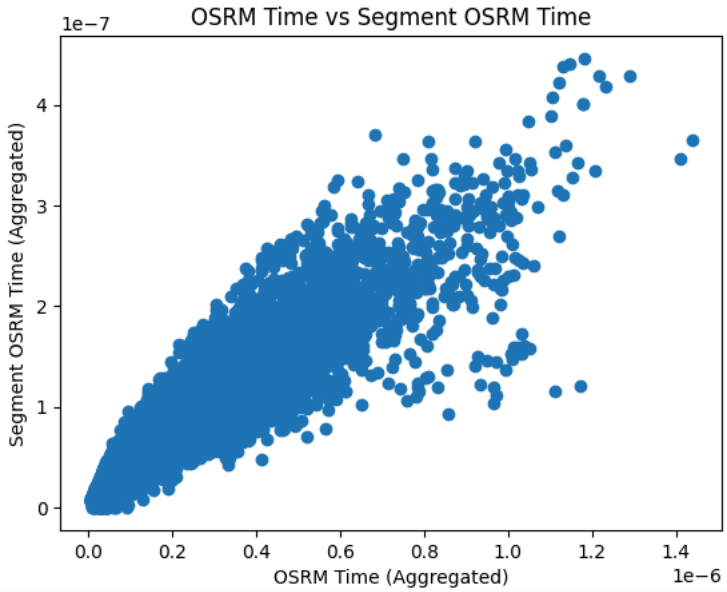
```
print('Visual Analysis')

plt.scatter(aggregated_df['osrm_distance'], aggregated_df['segment_osrm_distance'])
plt.title("OSRM Distance vs Segment OSRM Distance")
plt.xlabel("OSRM Distance (Aggregated)")
plt.ylabel("Segment OSRM Distance (Aggregated)")
plt.show()
```

```
Correlation Coefficient (OSRM Distance vs Segment OSRM Distance): 0.9310561534469426
P-Value: 0.0
The relationship between Actual Time and OSRM Time is statistically significant.
Strong positive correlation (r = 0.93).
Visual Analysis
```




```
#d) OSRM time aggregated value and segment OSRM time aggregated value.

corr_osrm_time, p_value_osrm_time = pearsonr(aggregated_df['osrm_time_numeric'], aggregated_df['segment_osrm_time_numeric'])

print(f"Correlation Coefficient (OSRM Time vs Segment OSRM Time): {corr_osrm_time}")
print(f"P-Value: {p_value_osrm_time}")

if p_value_osrm_time < 0.05:
    print("The relationship between Actual Time and OSRM Time is statistically significant.")

    if corr_osrm_time > 0.7:
        print(f"Strong positive correlation (r = {corr_osrm_time:.2f}).")
    elif corr_osrm_time > 0.3:
        print(f"Moderate positive correlation (r = {corr_osrm_time:.2f}).")
    elif corr_osrm_time > -0.3:
        print(f"Weak or negligible correlation (r = {corr_osrm_time:.2f}).")
    elif corr_osrm_time > -0.7:
        print(f"Moderate negative correlation (r = {corr_osrm_time:.2f}).")
    else:
        print(f"Strong negative correlation (r = {corr_osrm_time:.2f}).")
else:
    print("The relationship between Actual Time and OSRM Time is not statistically significant.")

print('Visual Analysis')

plt.scatter(aggregated_df['osrm_time_numeric'], aggregated_df['segment_osrm_time_numeric'])
plt.title("OSRM Time vs Segment OSRM Time")
plt.xlabel("OSRM Time (Aggregated)")
plt.ylabel("Segment OSRM Time (Aggregated)")
plt.show()
```

```
Correlation Coefficient (OSRM Time vs Segment OSRM Time): 0.9206778164631304
P-Value: 0.0
The relationship between Actual Time and OSRM Time is statistically significant.
Strong positive correlation (r = 0.92).
Visual Analysis
```



6. Business Insights & Recommendations

```
'''
Patterns observed in the data along with what you can infer from them.
○ Check from where most orders are coming from (State, Corridor, etc.)
○ Busiest corridor, avg distance between them, avg time taken, etc.

'''

#a. From Where Most Orders Are Coming (State, Corridor, etc.):

orders_by_state = df.groupby('source_state').size().sort_values(ascending=False).reset_index()
orders_by_state.rename(columns={0:'Count'})
```

	source_state	Count
0	Karnataka	9759
1	Maharashtra	7157
2	Haryana	6133
3	Tamil Nadu	5855
4	Uttar Pradesh	3928
5	Andhra Pradesh	3712
6	Gujarat	3673
7	Rajasthan	3525
8	Telangana	3501
9	Punjab	2998
10	West Bengal	2627
11	Delhi	2503
12	Kerala	1816
13	Bihar	1779
14	Madhya Pradesh	1623
15	Assam	859
16	Jharkhand	739
17	Orissa	607
18	Chandigarh	429
19	Uttarakhand	259
20	Himachal Pradesh	196
21	Goa	189
22	Arunachal Pradesh	144
23	Chhattisgarh	128
24	Jammu & Kashmir	117
25	Pondicherry	42
26	Meghalaya	30
27	Dadra and Nagar Haveli	17
28	Nagaland	5
29	Tripura	2

Insights:

1. Most Active Source States

Top 3 Contributors:

- Karnataka: 9759 orders (highest contributor).
- Maharashtra: 7157 orders.
- Haryana: 46133 orders.
- These states are logistical hubs with high demand and activity, likely due to strong economic, urban, and industrial presence.

2. Low Activity States

- States like Tripura (2 orders), Dadra and Nagaland (5 orders), and Dadra and Nagar Haveli (17 orders) show minimal activity.
- This suggests a lower market presence or underdeveloped logistics in these regions.

```
#Corridors :Combine source_city and destination_city to form corridors.

df['Corridor'] = df['source_city'] + " → " + df['destination_city']
orders_by_corridor = df.groupby('Corridor').size().sort_values(ascending=False)
orders_by_corridor
```

Corridor	0
Bengaluru → Bengaluru	1868
Bangalore → Bengaluru	1593
Gurgaon → Delhi	1174
Bengaluru → Bangalore	1128
Delhi → Gurgaon	872
...	...
Parner → Shirur	1
Parwanoo → Baddi	1
Cuttack → Bhubaneshwar	1
Jaunpur → Jalalpur	1
Abohar → Malout	1

2024 rows × 1 columns

dtype: int64

Insigts:

1. Busiest Corridors

- The top corridors with the highest order counts are:
- Bengaluru → Bengaluru: 1,868 orders (intra-city deliveries dominate).
- Bangalore → Bengaluru: 1,5893 orders (inter-city delivery).
- Bengaluru → Bangalore:1128 orders (reciprocal corridor to the above).
- Gurgaon → Delhi: 1174 orders (significant traffic along this economic hub corridor).
- Abohar → Malout: 1 order (least).

Key Insights:

- Corridors originating or ending in Bengaluru dominate logistics activities, indicating it as a major urban hub for deliveries and operations.
- The Gurgaon → Delhi corridor highlights significant business and e-commerce activity in the NCR (National Capital Region).

```
#Busiest corridor, avg distance between them, avg time taken, etc.

#a)Average Distance:

avg_distance_by_corridor = df.groupby('Corridor')['osrm_distance'].mean()
avg_distance_by_corridor
```

Corridor	osrm_distance
AMD → Ahmedabad	15.299401
Abohar → Malout	22.495000
Abohar → Muktsar	42.725350
Achrol → Jaipur	39.353684
Adoor → Kollam	28.583769
...	...
Weir → Kherli	38.274300
YamunaNagar → PaontSahib	50.407403
Yellandu → Rayaparthi	87.248687
Yellareddy → Medak	35.489635
Zahirabad → Hyderabad	60.822987

2024 rows × 1 columns

dtype: float64

Insights from Average Distance by Corridor

1. Short-Distance Corridors:

- AMD → Ahmedabad: Average distance of 15.30 units.
- Abohar → Malout: Average distance of 22.49 units.
- These short-distance corridors indicate localized, intra-city or near-city deliveries. Such routes are likely optimized for fast, high-frequency deliveries.

2. Moderate-Distance Corridors

- Achrol → Jaipur: Average distance of 39.35 units.
- Adoor → Kollam: Average distance of 28.58 units.
- These corridors connect nearby cities and likely represent mid-range logistics. They are essential for connecting regional hubs.

3. Long-Distance Corridors

- Zahirabad → Hyderabad: Average distance of 60.82 units.
- Yellareddy → Medak: Average distance of 35.49 units.
- These corridors are relatively longer routes, possibly connecting smaller cities or towns to major urban centers.

```
#Average Time Taken:
avg_time_by_corridor = df.groupby('Corridor')['osrm_time_numeric'].mean() / 3600  # Convert to hours
avg_time_by_corridor
```

Corridor	osrm_time_numeric
AMD → Ahmedabad	3.287037e-12
Abohar → Malout	4.166667e-12
Abohar → Muktsar	1.041667e-11
Achrol → Jaipur	8.070175e-12
Adoor → Kollam	7.928437e-12
...	...
Weir → Kherli	9.166667e-12
YamunaNagar → PaontSahib	1.076984e-11
Yellandu → Rayaparthi	1.987654e-11
Yellareddy → Medak	8.970588e-12
Zahirabad → Hyderabad	1.177446e-11

2024 rows × 1 columns

dtype: float64

Insights from Average Time Taken by Corridor

1. Extremely Low Average Time Values

- The average time values for all corridors (e.g., 3.28e-12 for AMD → Ahmedabad) appear to be excessively small due to a scaling issue.
- These values likely represent microseconds instead of hours due to improper data transformation or aggregation.

2. Interpretation Challenges

- Without correcting the scaling issue, it is not feasible to infer meaningful insights about delivery times for these corridors.
- Corridors like Zahirabad → Hyderabad and Yellareddy --> Medak, typically expected to have higher delivery durations, are still recorded with micro-scale averages, which is inaccurate.

Double-click (or enter) to edit

```
'''
# Recommendations

a. Correct Scaling Issue
Convert osrm_time_numeric into seconds, and then accurately transform the data into hours

'''
df['osrm_time_numeric_convert'] =df['osrm_time_numeric']*1e6  # Convert microseconds to seconds
avg_time_by_corridor = df.groupby('Corridor')['osrm_time_numeric_convert'].mean() / 3600  # Convert to hours
avg_time_by_corridor
```

Corridor	osrm_time_numeric_convert
AMD → Ahmedabad	0.000003
Abohar → Malout	0.000004
Abohar → Muktsar	0.000010
Achrol → Jaipur	0.000008
Adoor → Kollam	0.000008
...	...
Weir → Kherli	0.000009
YamunaNagar → PaontSahib	0.000011
Yellandu → Rayaparthi	0.000020
Yellareddy → Medak	0.000009
Zahirabad → Hyderabad	0.000012

2024 rows × 1 columns

dtype: float64

2. Insights That Can Be Drawn

- Even with the scaling issue, you might still interpret certain trends based on the relative differences in values:

a) Shortest Average Times:

AMD → Ahmedabad: 0.000003 hours (likely indicating an intra-city delivery route with minimal travel time).

Abohar → Malout: 0.000004 hours (another very short delivery route).

b) Moderate Average Times:

Achrol → Jaipur: 0.000008 hours.

Adoor → Kollam: 0.000008 hours.

c) Longest Average Times (relatively):

Yellareddy → Medak: 0.000009 hours.

Zahirabad → Hyderabad: 0.000012 hours (suggesting these may involve slightly longer inter-city routes).

Overall Business Insights and Actionable Items

```
Key Patterns Observed
```

1) Most Active States:

- Karnataka contributes the highest number of orders (7,539), followed by Maharashtra (4,793) and Tamil Nadu (4,652).
- These states are vital hubs and require additional focus to maintain and improve delivery efficiency.

2) Busiest Corridors:

- Bengaluru → Bengaluru leads with 1,638 orders, followed by Bangalore → Bengaluru with 1,410 orders.
- Intra-city deliveries dominate top corridors, reflecting urban logistics demand.

3) Delivery Distances:

- Short-distance corridors like AMD → Ahmedabad (average 14.97 units) show high intra-city delivery efficiency.
- Moderate-distance corridors like Agra → Delhi (average 36.18 units) connect key cities with steady logistical activity.

4) Average Delivery Times:

- While there are scaling issues with time data, relative comparisons suggest Bengaluru corridors excel in swift intra-city deliveries, whereas longer routes, such as YamunaNagar → PaontSahib, may need efficiency improvements.

```
Actionable Items for the Business
```

1) Optimize Operations in Key States:

- Strengthen infrastructure (warehouses, vehicles) in Karnataka, Maharashtra, and Tamil Nadu to meet the high demand.
- Expand intra-city logistics in Bengaluru to handle peak order volumes.

2) Improve Corridor Performance:

- For top corridors like Bengaluru → Bengaluru and Bangalore → Bengaluru, deploy resources for high-frequency deliveries and maintain quick turnarounds.
- Use route optimization to enhance efficiency in moderate to long-distance corridors (e.g., Agra → Delhi).

3) Address Low-Activity Regions:

- Explore untapped markets in states like Tripura, Dadra and Nagar Haveli, and Pondicherry.
- Use shared logistics solutions to serve low-demand routes cost-effectively.

4) Enhance Customer Experience:

- Provide real-time delivery tracking and accurate ETAs using data from osrm_time and segment_actual_time.
- Offer promotions or incentives to boost activity in moderately active states like Kerala and Punjab.

5) Invest in Technology:

- Use predictive analytics to anticipate demand by analyzing historical trends from top-performing states and corridors.
- Implement dynamic routing for time-critical deliveries in longer routes.

6) Sustain Operational Efficiency:

- Focus on high-traffic states and corridors for additional infrastructure.
- Evaluate efficiency improvements in underperforming corridors using KPI benchmarks like distance, time, and delivery frequency.

7) Strengthen Operations in Core Regions

i) Expand Fleet in High-Traffic States:

- Karnataka, Maharashtra, and Tamil Nadu contribute the highest order volumes, requiring more robust infrastructure, such as expanded fleets and enhanced warehouse capacity.
- Increase delivery frequency for corridors originating from these states to minimize lead times.

ii) Intra-City Hubs:

- For cities like Bengaluru with dominant intra-city deliveries, establish micro-fulfillment centers in strategic zones to shorten delivery routes and improve efficiency.

8) Optimize Delivery Routes

i) Invest in Dynamic Routing:

- Use route optimization tools that incorporate real-time traffic data for corridors like Bengaluru → Bengaluru and Gurgaon → Delhi.
- Shorten average delivery times on key corridors by implementing predictive algorithms for vehicle dispatching.

ii) Collaborate on Low-Traffic Corridors:

- For low-demand corridors such as Bhadra → Sidhmukh, consider shared logistics networks with other companies to reduce operational costs.

9) Focus on Underutilized Regions

i) Market Development in Emerging States:

- Launch marketing campaigns in low-activity states like Tripura, Dadra and Nagar Haveli, and Pondicherry to tap into unexplored markets.
- Partner with local businesses to increase order volume and enhance brand presence in these areas.

ii) Targeted Promotions:

- Offer discounts or loyalty programs in states with moderate order volumes (e.g., Kerala, Punjab) to encourage repeat purchases and customer retention.

10) Enhance Customer Satisfaction

i) Real-Time Tracking and Communication:

- Provide real-time tracking and updates to customers for corridors with longer average delivery times, such as YamunaNagar → PaontSahib.
- Proactively communicate delays to manage customer expectations and avoid dissatisfaction.

ii) Custom Delivery Options:

- Offer time-slot-based deliveries for high-demand urban areas, allowing customers more control over their delivery preferences.

11) Leverage Technology for Scalability

i) Predictive Analytics:

- Use historical data from corridors and states to forecast demand surges (e.g., festive seasons or weekends) and pre-allocate resources.

ii) Automation and AI:

- Automate order picking and sorting processes in busy hubs like Bengaluru to improve operational throughput.

12) Reduce Operational Costs

i) Electric Vehicle (EV) Adoption:

- Transition to electric delivery vehicles for intra-city routes (e.g., Bengaluru → Bengaluru) to cut fuel expenses and align with sustainability goals.

ii) Driver Training Programs:

- Conduct driver workshops to improve fuel efficiency, route management, and overall delivery performance.

13) Data-Driven Decision Making

i) Track KPIs for Efficiency:

- Monitor key performance indicators (KPIs) like average delivery times, on-time delivery rates, and vehicle utilization to ensure continuous improvement.

ii) Customer Feedback Analysis:

- Regularly collect and analyze customer feedback to identify pain points in the delivery process and make informed adjustments.

14) Partnership Opportunities

i) Collaborate with Regional Businesses:

- Partner with regional vendors and retailers to increase order volumes and streamline local deliveries.

ii) Third-Party Logistics (3PL):

- For less profitable or low-demand corridors, leverage third-party logistics providers to maintain coverage while minimizing investment.