Project Report on

LOGISTICS DELAY ANALYSIS

In partial fulfillment for the award

Of

Professional Certification in Data Analysis and Visualization

Year 2024-2025

Submitted By

Fathimath Saniya C

Tools & Technologies Used:

Python, R Programming, Tableau

Duration:

15/07/2025 – 30/07/2025

Submission Date:

31/07/2025

G - TEC CENTRE OF EXCELLENCE PERINTHALMANNA

ABSTRACT

This project, titled **"Logistics Delay Analysis"** focuses on identifying delivery inefficiencies in logistics operations using statistical and analytical techniques. Timely deliveries are vital in ensuring operational efficiency and customer satisfaction, especially in last-mile logistics. The segment times, source/destination locations, trip creation time, and delay durations.

Python was used for data preprocessing, exploratory data analysis (EDA), and hypothesis testing. Tableau was employed to create a dynamic dashboard, helping to visually interpret complex delivery metrics. Statistical tests such as the t-test, z-test, and F-test were applied to examine differences in delays between weekdays and weekends, variance in travel times, and other hypotheses.

Key findings indicate that delays are significantly higher on weekdays compared to weekends and that actual travel times vary greatly from estimated times. High-delay routes and peak operational times were also identified. The insights from this project offer recommendations for route optimization, better scheduling, and improved performance monitoring. This project demonstrates how integrating data analytics and visualizations can effectively support strategic decision-making in logistics.

INTRODUCTION

The logistics industry plays a critical role in ensuring the smooth movement of goods and services. However, one of the key challenges faced by logistics companies is the delay in deliveries, which can lead to increased operational costs, missed deadlines, and reduced customer satisfaction. In recent years, with the rise of e-commerce and last-mile delivery demands, the pressure to deliver shipments on time has grown substantially. Although companies use route estimation tools like OSRM (Open Source Routing Machine) to predict delivery times, these estimates often differ from actual outcomes due to unpredictable variables such as traffic congestion, infrastructure limitations, or inefficiencies in scheduling. This project, titled **"Logistics Delay Analysis"** aims to address these challenges by analyzing a large dataset containing over 140,000 trip records. The objective is to uncover patterns in delivery delays, identify key factors contributing to inefficiencies, and provide actionable insights through both statistical testing and interactive visualization. Using Python, statistical methods like t-tests, z-tests, and F-tests are applied to evaluate delay trends and verify whether significant differences exist across time frames and route types. A Tableau dashboard is also created to help visualize the most delayed destinations, peak delay days, and cumulative delays across trips. The scope of this analysis is limited to internal delivery data and does not incorporate external factors like weather or real-time traffic data. Nevertheless, the findings are expected to provide substantial business value by improving route planning, enhancing the accuracy of delivery time predictions, and ultimately supporting more efficient and reliable logistics operations.

LITERATURE REVIEW

1. Route Optimization and Emergency Logistics (2001)

The foundational work by Gendreau et al. introduced dynamic and real-time routing optimization through parallel tabu search techniques. Though originally developed for emergency services, its approach to dynamically adjusting routes to real-time conditions has direct implications for logistics delays. Their methodology laid the groundwork for adaptive logistics routing, showing that static delivery plans are less efficient under variable conditions.

2. Impact of Warehouse Design on Logistics Delays (2009)

Gue and Meller conducted a comprehensive study on warehouse layout and its impact on outbound logistics. The study found that inefficient warehouse design and disorganized order picking are significant contributors to outbound delays. The research highlighted how delays can originate even before goods are dispatched and proposed lean warehousing as a mitigating strategy.

3. Weather and Climate Effects on Transportation Systems (2009)

Koetse and Rietveld explored how weather variability—especially snow, rain, and fog—impacts transportation systems. Their analysis, although broad, strongly relates to road-based logistics where these conditions often lead to delay variability. They emphasized the need for forecasting weather impacts and integrating them into scheduling algorithms.

4. Urban Freight Congestion and Delivery Timing (2016)

Holguín-Veras et al. identified urban congestion as one of the most disruptive elements in last-mile delivery logistics. Their study emphasized that delivery schedules in cities are highly vulnerable to traffic fluctuations, and companies must incorporate flexible and responsive routing to minimize delays.

5. Classification and Solving the Vehicle Routing Problem (2016)

Braekers et al. reviewed various VRP-solving algorithms including time-dependent and real-time routing. Their work showed that smart routing not only reduces mileage but significantly lowers delay probabilities. The study supports the use of AI-based optimization models to address logistic inefficiencies.

6. Driver Behavior and Logistics Performance (2018)

Lin and Zhou examined how human factors such as fatigue, stress, and lack of technological support contribute to delivery delays. Their findings show that monitoring tools like telematics systems can reduce uncertainty and ensure more consistent delivery times through better driver management.

7. Machine Learning in Delay Prediction (2020)

Yu et al. provided one of the most recent studies focusing on machine learning to predict logistics delays. Using historical and real-time data, their models accurately forecast delay zones. The study confirmed that predictive analytics can reduce reactive delays and improve customer satisfaction in large-scale delivery systems.

8. The Role of Big Data Analytics in Supply Chain Management (2021)

Wang et al. explored how big data analytics can enhance visibility and reduce delays across complex logistics networks. Their study emphasized that integrating IoT devices with advanced data analytics enables real-time monitoring and predictive delay management, especially in last-mile delivery and inventory flow.

RESEARCH GAP

While logistics delay analysis has seen considerable attention, much of the existing research focuses on isolated variables like traffic, weather, or infrastructure without integrating diverse data-driven tools. Most studies fail to combine Python for data exploration, R for statistical testing, and Tableau for visualization into a unified workflow. Moreover, limited work has been done using Indian logistics datasets, which present unique challenges such as inconsistent infrastructure, regional delivery bottlenecks, and variable route types. There’s also a lack of studies validating observed patterns using formal hypothesis testing. This project bridges these gaps by applying a cross-platform, statistically validated approach to uncover actionable insights from real-world delivery data.

DATA COLLECTION & PREPROCESSING

Data Source and Collection Methods

The dataset used in this project is titled delhivery.csv. It contains delivery trip records collected from a logistics company's internal operations database. Each record captures key trip information such as trip creation time, source and destination names, estimated and actual segment times, delay in hours, route type, and cutoff time. The dataset was designed to analyze delivery performance and delay patterns across different conditions

Data Quality Assessment and Cleaning Procedures

The dataset was cleaned using Python. The following preprocessing steps were carried out:

- **Missing values**: All missing values in categorical columns were replaced with "Not Specified" to avoid null-related errors during analysis.
- **Duplicate entries**: Checked for and removed duplicate records to maintain consistency.
- **Data types**: Ensured proper data types for each column. For instance, trip_creation_time was converted into datetime format for time-based analysis.
- **Unit conversions**: Columns such as segment_actual_time and trip_duration were originally in minutes and were converted into **hours** to allow for consistent statistical comparison and reporting.
- **Rename column:** A new column trip_duration was created by duplicating the existing start_scan_to_end_scan column to enhance clarity and simplify further analysis.

Feature Engineering and Selection Techniques

To enhance the dataset for deeper analysis and visualization, several new features were created:

- **day_type**: This column was derived from the trip_creation_time and classified each trip as either Weekday or Weekend. It was essential for hypothesis testing (e.g., comparing delays between day types).
- **delay_in_minutes**: This new column was calculated by converting the delay_in_hours to minutes, making it more readable and suitable for Tableau dashboards.

- **Filtered subsets**: The dataset was filtered by certain fields, such as high-delay trips or specific route types, to support focused statistical analysis and visual storytelling.
- **Categorical parsing**: Fields like source_name and destination_name, which were initially in combined formats, were parsed and cleaned to improve usability in charts and route-based analysis.

Columns Selected for Analysis

The following columns were selected for core analysis and visualization: trip_creation_time, day_type, cutoff_time, source_name, destination_name, route_type, segment_actual_time, segment_osrm_time, trip_duration, delay_in_hours, delay_in_minutes

METHODOLOGY

This project follows a multi-tool analytical approach combining Python, R programming, and Tableau to analyze logistics delivery delays. The methodology is designed to explore and validate delay patterns using statistical tests and present actionable insights through visual dashboards for stakeholders.

Tools and Technologies Used

- **Python** was used for data cleaning, transformation, and exploratory data analysis. Libraries such as pandas, matplotlib, and seaborn were used for handling data and creating visual summaries.
- **R programming** was used to perform statistical hypothesis testing. Functions such as t.test(), z.test(), aov(), and chisq.test() were applied to assess delay variations across categories.
- **Tableau** was used to build an interactive dashboard, making the analytical insights accessible and visually engaging for operational decision-makers.

Exploratory Data Analysis – Python

After cleaning the dataset and creating new features like day_type and delay_in_minutes, exploratory analysis was carried out using Python to identify patterns in delivery behavior.

Visualizations:

- **Average Trip Duration by Route Type (Bar Chart):**

Shows which route types have the longest average delivery times.

- **Trip Duration Distribution (Histogram):**

Displays how trip durations are spread, helping identify common delivery times and outliers.

- **On-Time vs Late Deliveries (Pie Chart):**

Highlights the percentage of deliveries completed on time versus those delayed.

- **Time per KM (Box Plot):**

Shows how much time is spent per kilometer, helping detect route inefficiencies.

Groupings were made using the groupby() function to calculate mean delays across different features

Statistical Testing – R Programming

The statistical testing component was performed in R to validate the insights gained from exploratory analysis. The following tests were applied:

1. **T-test** to compare trip durations between Weekday and Weekend trips using the day_type column.
2. **Z-test** to check if the average trip duration significantly exceeds 5 hours.
3. **F-test** to compare variance between segment_actual_time and segment_osrm_time to test consistency in travel time estimates.
4. **ANOVA** to test if trip duration varies across different route_type categories.
5. **Chi-Square Test** to assess whether pickup zones (source_name) influence the route_type.

Data Visualization and Dashboarding – Tableau

After the data was cleaned and analyzed in Python, Tableau was used to create an interactive dashboard for visualizing delivery delays and performance. Tableau was chosen for its ease of use and powerful ability to convert raw data into visual insights, making it accessible for both technical and non-technical stakeholders.

- **Interactive Filters:**

  Filters such as *Day Type* and *Source Name* were added, enabling users to focus on specific days or locations to understand localized delay patterns.

- **Simple Layout and Clear Visuals:**

  The dashboard uses a clean and minimal design with clear charts, and labels. This helps users quickly understand delivery volumes, delay trends, and destination performance.

- **Visual Elements Included:**

  Summary table, bar charts, donut and lollipop charts, and line graphs were used to highlight key patterns like average delay by destination, trip distribution by day, and overall delay trends.

- **Navigation Button to Story:**

A button was added to allow users to navigate to a Tableau Story, providing a guided flow through the most important findings and visual summaries.

RESULTS AND ANALYSIS

Python Based Result

- Full Truck Load routes had much higher average OSRM time errors (287.5 hours) compared to Carting routes (29.2 hours).
- Top delayed destinations included IND490023AAA and IND221401AAA, with delays up to 818 minutes.
- The correlation between cutoff factor and segment factor was very low (−0.03), showing little relationship.
- Average difference between actual and expected segment time was 17.7 minutes, with delays reaching up to 2400 minutes.
- Only 1.77% of trips were completed earlier than expected.
- FTL trips showed high time variation, with a standard deviation of 11.02 hours.
- The average time spent per kilometer was 0.036 hours, but in some cases it went above 2 hours/km.
- Source center IND000000ACB caused the most delay, contributing 13.6% of total delay minutes.
- Only 10.28% of deliveries were on time; 89.72% were delayed.
- Weekend trips had slightly more delays (17.78 minutes) compared to weekdays (17.65 minutes).
- Some route schedules, like bcce7b68..., had total delays exceeding 29,000 minutes.
- Average trip duration was 21.7 hours for FTL routes and 3.4 hours for Carting routes.
- FTL routes have much longer average trip durations compared to Carting routes.
- Most trip durations are short, with a high number of deliveries taking less than 20 hours.
- A significant majority of deliveries (89.7%) are delayed, while only 10.3% are on time.
- The time per kilometer is generally low but has a few extreme outliers.

R Based Result

- There is a significant difference in trip duration between weekdays and weekends, indicating that the day of the week affects delivery time.
- The average trip duration is significantly greater than 5 hours, suggesting longer delivery times than expected.
- The variances between segment_actual_time and segment_osrm_time are significantly different, showing inconsistency between actual and planned delivery segments.
- Route type has a significant effect on trip duration, meaning some route types consistently lead to longer or shorter deliveries.
- There is a significant association between source name and route type, indicating that the origin of the trip influences the route chosen.

Tableau Based Result

- Minimum average delay occurred on Day 2, and maximum on Day 22, showing inconsistency in daily delivery performance.
- Around 73.7% of deliveries were made on weekdays, while 26.3% occurred on weekends.
- Chhattisgarh faced the highest delays among top destinations; Madhya Pradesh had the least.
- Haryana had the longest total delivery time at 164K hours, compared to Tamil Nadu with the shortest at 20.7K hours.
- Delays steadily increase throughout the week, with Friday and Saturday contributing the most to cumulative delay.
- Out of 144.9K total deliveries, 14.9K were on time. The average delay was 1.1K minutes, and the longest delay recorded was 145.1K minutes.

CONCLUSION

This project delivered a comprehensive analysis of logistics delays using a blend of Python, R, and Tableau. Through statistical testing, visual analytics, and data exploration, the study identified key factors contributing to delivery inefficiencies across different routes, source centers, and day types. The analysis found that most deliveries experienced delays, with FTL (Full Truck Load) routes showing significantly longer trip durations and higher variance compared to Carting routes.

Weekend deliveries tended to have longer average durations than weekdays, and statistical tests confirmed the influence of route type, source location, and time of day on delivery performance. Visualization techniques like bar charts, pie charts, and box plots clearly revealed patterns such as delay concentration in specific destinations and higher time per kilometer in certain trips.

Overall, the findings emphasize the need for smarter route planning and time management to enhance logistics efficiency. This project demonstrates how combining Python, R, and Tableau provides actionable insights, enabling logistics teams to reduce delays, optimize delivery routes, and improve overall operational performance.

FUTURE WORKS

- **Real-Time Data Integration**: Incorporate live tracking data to monitor ongoing deliveries and update delay predictions dynamically.
- **Advanced Predictive Modeling**: Implement machine learning models to forecast delays based on historical patterns, weather, traffic, and route conditions.
- **Geospatial Analysis**: Add map-based analysis to visualize bottlenecks across geographic zones for better route optimization.
- **Driver Performance Metrics**: Analyze individual driver behavior and shift patterns to identify delay contributors from the human side.
- **Automation and Alerts**: Develop automated dashboards with alert systems that notify operations teams of potential delays in real time.

REFERENCES

1. Gendreau, M., Laporte, G., & Semet, F. (2001). A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing, 27*(12), 1641–1653. https://doi.org/10.1016/S0167-8191(01)00124-4

2. Gue, K. R., & Meller, R. D. (2009). The warehouse design problem: A survey. *European Journal of Operational Research, 193*(2), 425–436. https://doi.org/10.1016/j.ejor.2007.11.045

3. Koetse, M. J., & Rietveld, P. (2009). The impact of climate change and weather on transport: An overview of empirical findings. *Transportation Research Part D: Transport and Environment, 14*(3), 205–221. https://doi.org/10.1016/j.trd.2008.12.004

4. Holguín-Veras, J., Jaller, M., Van Wassenhove, L. N., Pérez, N., & Toyama, N. (2016). Urban freight systems and the potential for disruption. *Transportation Research Part A: Policy and Practice, 86*, 285–302. https://doi.org/10.1016/j.tra.2016.02.001

5. Braekers, K., Ramaekers, K., & Van Nieuwenhuyse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering, 99*, 300–313. https://doi.org/10.1016/j.cie.2015.12.007

6. Lin, P., & Zhou, W. (2018). Effects of driver behavior on logistics service quality. *Journal of Transportation Safety & Security, 10*(3), 237–256. https://doi.org/10.1080/19439962.2017.1309801

7. Yu, Y., Wang, X., & Huang, G. Q. (2020). Real-time logistics delay prediction using machine learning: A review and future directions. *International Journal of Production Research, 58*(3), 829–846. https://doi.org/10.1080/00207543.2019.1616992

8. Wang, G., Gunasekaran, A., Ngai, E. W. T., & Papadopoulos, T. (2021). Big data analytics in logistics and supply chain management: Certain investigations for research and applications. *International Journal of Production Economics, 236*, 108146. https://doi.org/10.1016/j.ijpe.2021.108146

SUPPORTING FILES

Python

```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
```

```
data=pd.read_csv('delhivery.csv')
data.head()
```

```
data.isnull().sum()
```

```
data['source_name'].fillna('Not defined', inplace=True)
```

```
data['destination_name'].fillna('Not defined', inplace=True)
```

```
data.duplicated().sum()
```

```
data['actual_time_hours'] = (data['actual_time'] / 60).round(2)
data['trip_duration_hours'] = (data['start_scan_to_end_scan'] / 60).round(2)
data.drop(['actual_time', 'start_scan_to_end_scan'], axis=1, inplace=True)
```

```
data
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data['trip_creation_time'] = pd.to_datetime(data['trip_creation_time'])
data['od_start_time'] = pd.to_datetime(data['od_start_time'])
```

```
#1.OSRM Accuracy Across Trip Types
data['osrm_error'] = abs(data['osrm_time'] - data['actual_time_hours'])
osrm_accuracy_by_route_type = data.groupby('route_type')['osrm_error'].mean()
print(osrm_accuracy_by_route_type)
```

```
#2.Delay pattern by destination
data['delay_minutes'] = data['segment_actual_time'] - data['segment_osrm_time']
delay_by_destination = data.groupby('destination_center')['delay_minutes'].mean().sort_values(ascending=False).head(10)
delay_by_destination
```

```
#3. Correlation Between Cutoff and Segment Factors
correlation_cf_sf=data[['cutoff_factor', 'segment_factor']].corr()
correlation_cf_sf
```

```
#4.Segment Time Deviation
data['segment_time_diff'] = data['segment_actual_time'] - data['segment_osrm_time']
segment_time_stats= data['segment_time_diff'].describe()
segment_time_stats
```

```
#5.Propotion of early trip factor
early_trips = (data['factor'] < 1).mean()
early_trips
```

```
#6. Standard Deviation of Actual Time by Route Type
std_by_route_type = data.groupby('route_type')['actual_time_hours'].std().sort_values(ascending=False)
std_by_route_type
```

```
#7.time_per_km_stats
data['time_per_km'] = data['actual_time_hours'] / data['actual_distance_to_destination']
time_per_km_stats = data['time_per_km'].describe()
time_per_km_stats
```

```
#8.Delay percentage by source center
total_delay_by_source = data.groupby('source_center')['delay_minutes'].sum()
grand_total_delay = total_delay_by_source.sum()
delay_percentage = (total_delay_by_source / grand_total_delay) * 100
top_delay_percent = delay_percentage.sort_values(ascending=False).head(10)
print(top_delay_percent)
```

```
#9.% of late and on time deliveries
on_time_pct = (data['delay_minutes'] <= 0).mean() * 100
late_pct = (data['delay_minutes'] > 0).mean() * 100
print(f"On-Time Deliveries: {on_time_pct:.2f}%, Late Deliveries: {late_pct:.2f}%")
```

```
#10.delay distribution by weekday and weekends
data['trip_creation_time'] = pd.to_datetime(data['trip_creation_time'], errors='coerce')
data['day_type'] = data['trip_creation_time'].dt.day_name().isin(['Saturday', 'Sunday']).map({True: 'Weekend', False: 'Weekday'})
weekend_delay = data.groupby('day_type')['delay_minutes'].mean()
print(weekend_delay)
```

```
#11.top route schedules by total delay
total_delay_by_route = data.groupby('route_schedule_uuid')['delay_minutes'].sum().sort_values(ascending=False).head(10)
print(total_delay_by_route)
```

```
#12.on time delivery rate by day type
on_time_rate = data.groupby('day_type')['delay_minutes'].apply(lambda x: (x <= 0).mean() * 100)
print("On-Time Delivery Rate (%):\n", on_time_rate)
```

```python
# 13. Average trip duration by route type

avg_duration_by_route = data.groupby("route_type")["trip_duration_hours"].mean().sort_values()
sns.barplot(x=avg_duration_by_route.values, y=avg_duration_by_route.index, palette="viridis")
plt.title("Average Trip Duration by Route Type", fontsize=18)
plt.xlabel("Average Duration (hours)", fontsize=14)
plt.ylabel("Route Type", fontsize=14)
```

```python
#14.distribution of trip duration
sns.histplot(data['trip_duration_hours'], bins=30, kde=True, color='skyblue')
plt.title("Distribution of Trip Duration (Hours)")
plt.xlabel("Trip Duration (hours)")
plt.ylabel("Frequency")
plt.grid(True)
plt.show()
```

```python
#15.Calculate delay in minutes if not already created
data['delay_minutes'] = data['segment_actual_time'] - data['segment_osrm_time']
on_time = (data['delay_minutes'] <= 0).sum()
late = (data['delay_minutes'] > 0).sum()
plt.pie([on_time, late], labels=["On-Time", "Late"], autopct='%1.1f%%', colors=[ '#D1FFD1','#FFE5B4' ], startangle=90)
plt.title("On-Time vs Late Deliveries")
plt.show()
```

```python
#16.Time per distribution km
sns.boxplot(x=data['time_per_km'], color='steelblue')
plt.title("Box Plot of Time per KM")
plt.xlabel("Time per KM (hours/km)")
plt.grid(True)
plt.show()
```

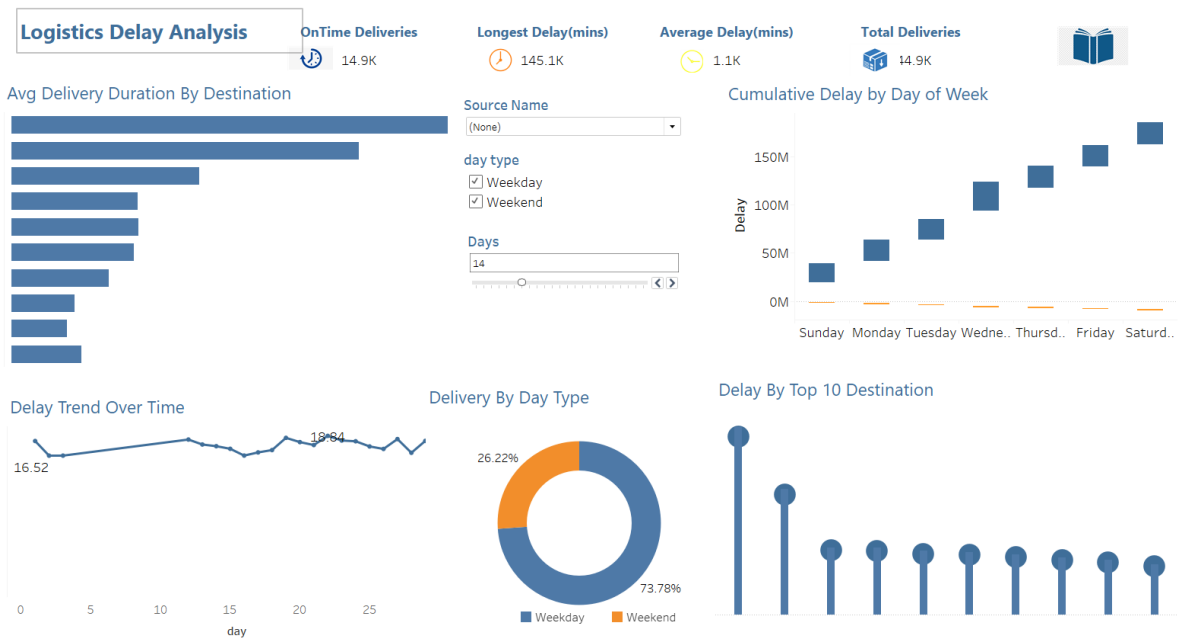
```python
data.to_csv("modified_delhivery",index=False)
```

R Programming

```r
install.packages("dplyr")
library("dplyr")
df<-read.csv("C:\\Users\\LENOVO\\OneDrive\\Desktop\\python project\\modified_delhivery.csv")
View(df)

#1.compare trip_duration_hours by day_type
df$day_type <- ifelse(weekdays(as.Date(df$trip_creation_time)) %in% c("Saturday", "Sunday"),
                      "Weekend", "Weekday")
t_result<-t.test(trip_duration_hours ~ day_type, data = df)
if (t_result$p.value < 0.05) {
  cat("Reject H0: There is a significant difference in trip duration between weekdays and weekends.\n\n")
} else {
  cat("Fail to Reject H0: No significant difference in trip duration between weekdays and weekends.\n\n")
}

install.packages("BSDA")
library(BSDA)
#2.Is the average trip duration significantly greater than 5 hours?
z_result<-z.test(df$trip_duration_hours, mu = 5, sigma.x = sd(df$trip_duration_hours), alternative = "greater")
if (z_result$p.value < 0.05) {
  cat("Reject H0: The average trip duration is significantly greater than 5 hours.\n\n")
} else {
  cat("Fail to Reject H0: The average trip duration is not significantly greater than 5 hours.\n\n")
}

#3.F-test between segment_actual_time and segment_osrm_time
f_result<-var.test(df$segment_actual_time, df$segment_osrm_time)
if (f_result$p.value < 0.05) {
  cat("Reject H0: Variances of segment_actual_time and segment_osrm_time are significantly different.\n\n")
} else {
  cat("Fail to Reject H0: No significant difference in variances.\n\n")
}
```

```r
#4.is route type influence trip duration
anova_result <- aov(trip_duration_hours ~ route_type, data = df)
summary_result<-summary(anova_result)
pval<-summary_result[[1]][["Pr(>F)"]][1]
if (!is.null(pval) && pval < 0.05) {
  print("Reject H0: Route type has a significant effect on trip duration.\n")
} else {
  print("Fail to reject H0: Route type does NOT have a significant effect on trip duration.\n")
}

#5.Does pickup_zone influence route_type
x<- table(df$source_name, df$route_type)
chi_result<-chisq.test(x)
if (chi_result$p.value < 0.05) {
  cat("Reject H0: Source name and route type are dependent (associated).\n\n")
} else {
  cat("Fail to Reject H0: No significant association between source name and route type.\n\n")
}
```

Tableau


Logistics Delay Analysis
OnTime Deliveries
14.9K
Longest Delay(mins)
145.1K
Average Delay(mins)
1.1K
Total Deliveries
14.9K
Avg Delivery Duration By Destination
Source Name
(None)
day type
Weekday
Weekend
Days
14
Cumulative Delay by Day of Week
Delay
150M
100M
50M
0M
Sunday
Monday
Tuesday
Wedne..
Thursd..
Friday
Saturd..

Delay Trend Over Time
16.52
18.84
0
5
10
15
20
25
day
Delivery By Day Type
26.22%
73.78%
Weekday
Weekend
Delay By Top 10 Destination