Final_Project

Nethra Subramanian

2024-05-22

Load the Data

```
#Load the dataset
df <- read.csv("CTA_-_Ridership_-_Daily_Boarding_Totals_20240512.csv")
head(df)
```

```
##   service_date day_type    bus rail_boardings total_rides
## 1     1/1/2001        U 297192         126455      423647
## 2     1/2/2001        W 780827         501952     1282779
## 3     1/3/2001        W 824923         536432     1361355
## 4     1/4/2001        W 870021         550011     1420032
## 5     1/5/2001        W 890426         557917     1448343
## 6     1/6/2001        A 577401         255356      832757
```

Look at the Data

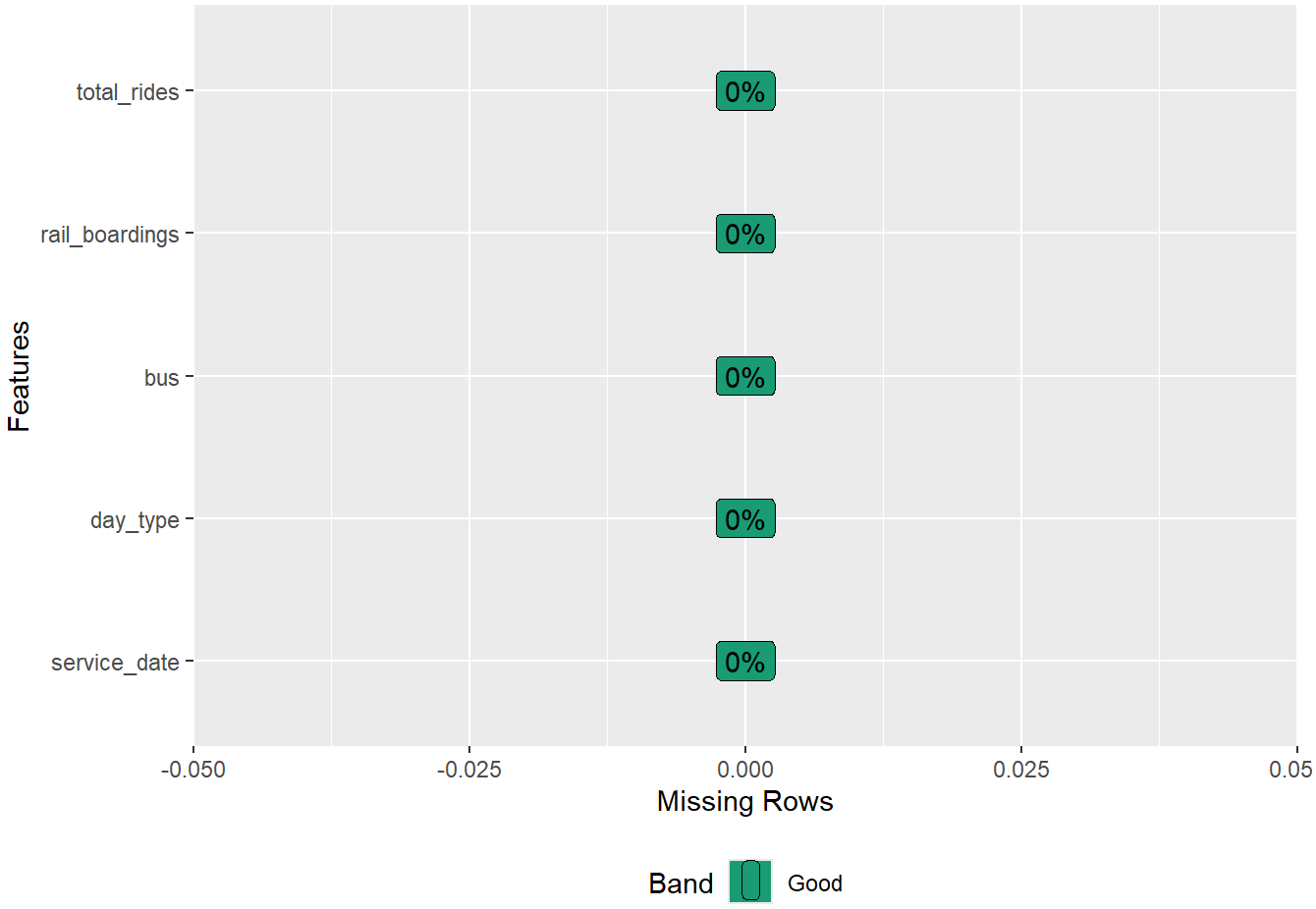
```
# Get the structure and a glimpse of the data
str(df)
```

```
## 'data.frame':    8522 obs. of  5 variables:
##  $ service_date  : chr  "1/1/2001" "1/2/2001" "1/3/2001" "1/4/2001" ...
##  $ day_type      : chr  "U" "W" "W" "W" ...
##  $ bus           : int  297192 780827 824923 870021 890426 577401 375831 985221 978377 984884
...
##  $ rail_boardings: int  126455 501952 536432 550011 557917 255356 169825 590706 599905 602052
...
##  $ total_rides   : int  423647 1282779 1361355 1420032 1448343 832757 545656 1575927 1578282
1586936 ...
```

```
glimpse(df)
```

```
## Rows: 8,522
## Columns: 5
## $ service_date   <chr> "1/1/2001", "1/2/2001", "1/3/2001", "1/4/2001", "1/5/20…
## $ day_type       <chr> "U", "W", "W", "W", "W", "A", "U", "W", "W", "W", "W", …
## $ bus            <int> 297192, 780827, 824923, 870021, 890426, 577401, 375831,…
## $ rail_boardings <int> 126455, 501952, 536432, 550011, 557917, 255356, 169825,…
## $ total_rides    <int> 423647, 1282779, 1361355, 1420032, 1448343, 832757, 545…
```

```
# Visualize missing data
plot_missing(df)
```



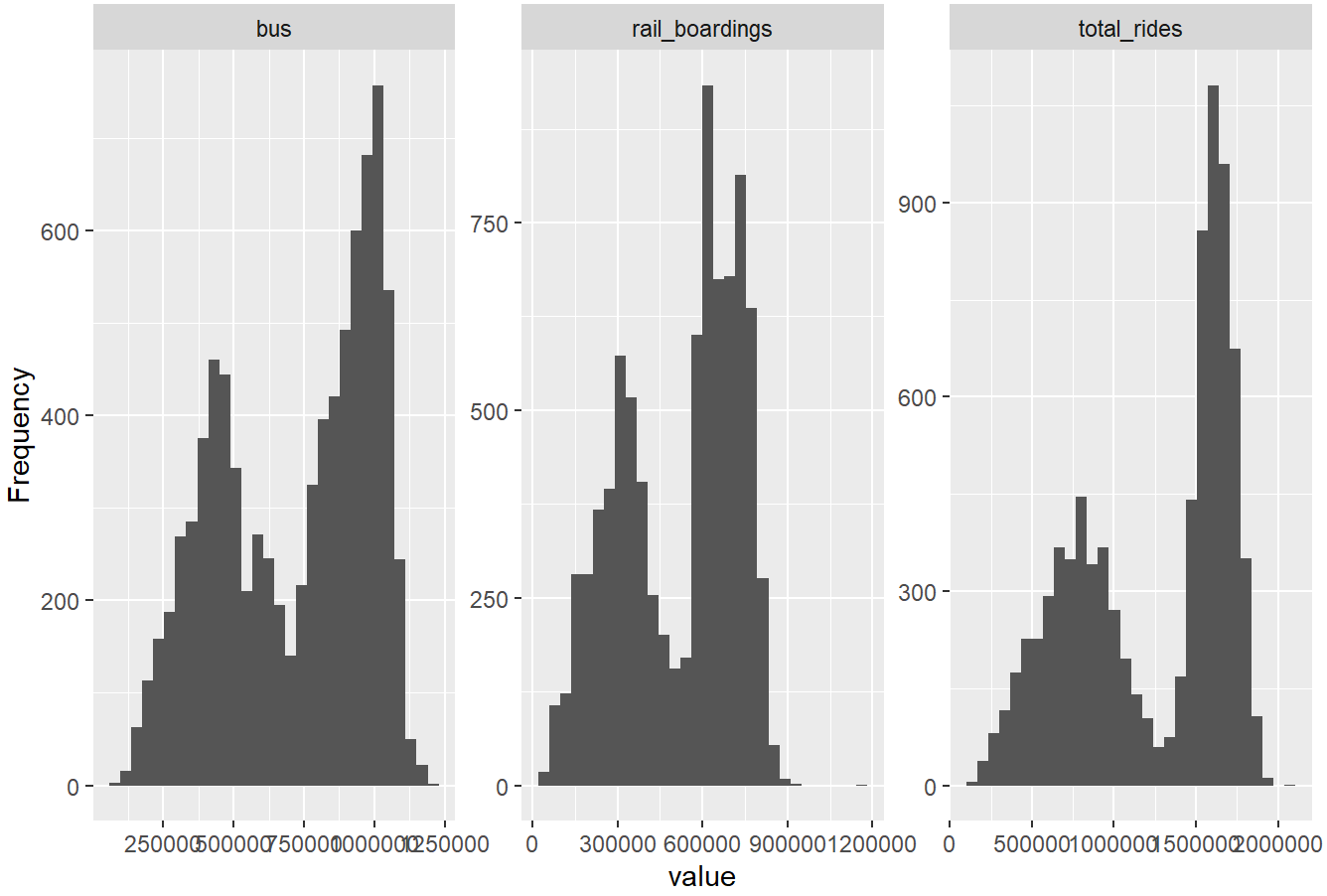
EDA

```
# Convert date column to Date type if applicable
df$service_date <- as.Date(df$service_date, format = "%m/%d/%Y")
```

```
# Descriptive statistics and distribution
summary(df)
```

```
##   service_date          day_type              bus          rail_boardings
##  Min.   :2001-01-01   Length:8522        Min.   :  80783   Min.   :  23544
##  1st Qu.:2006-11-01   Class :character   1st Qu.: 465352   1st Qu.: 328310
##  Median :2012-07-31   Mode  :character   Median : 791643   Median : 592324
##  Mean   :2012-08-02                      Mean   : 720003   Mean   : 518033
##  3rd Qu.:2018-04-30                      3rd Qu.: 964753   3rd Qu.: 698377
##  Max.   :2024-02-29                      Max.   :1211992   Max.   :1146516
##   total_rides
##  Min.   : 110047
##  1st Qu.: 805008
##  Median :1482361
##  Mean   :1238035
##  3rd Qu.:1636630
##  Max.   :2049519
```

```
plot_histogram(df)
```



1. service_date:

- Min.: The earliest date in the dataset is January 1, 2001.
- 1st Qu. (1st Quartile): The date at the 25th percentile is November 1, 2006.
- Median: The median date (50th percentile) is July 31, 2012.
- Mean: The average date is approximately August 2, 2012.

- 3rd Qu. (3rd Quartile): The date at the 75th percentile is April 30, 2018.
- Max.: The latest date in the dataset is February 29, 2024.

2. day_type:

- Length: There are 8,522 entries for day_type.
- Class: The data type is character.
- Mode: The most frequent day_type value (not shown in the summary).

3. bus:

- Min.: The minimum number of bus rides in a day is 80,783.
- 1st Qu. (1st Quartile): 25% of the days have bus rides less than 465,352.
- Median: The median number of bus rides is 791,643.
- Mean: The average number of bus rides is 720,003.
- 3rd Qu. (3rd Quartile): 75% of the days have bus rides less than 964,753.
- Max.: The maximum number of bus rides in a day is 1,211,992.

4. rail_boardings:

- Min.: The minimum number of rail boardings in a day is 23,544.
- 1st Qu. (1st Quartile): 25% of the days have rail boardings less than 328,310.
- Median: The median number of rail boardings is 592,324.
- Mean: The average number of rail boardings is 518,033.
- 3rd Qu. (3rd Quartile): 75% of the days have rail boardings less than 698,377.
- Max.: The maximum number of rail boardings in a day is 1,146,516.

5. total_rides:

- Min.: The minimum total rides in a day is 110,047.
- 1st Qu. (1st Quartile): 25% of the days have total rides less than 805,008.
- Median: The median number of total rides is 1,482,361.
- Mean: The average number of total rides is 1,238,035.
- 3rd Qu. (3rd Quartile): 75% of the days have total rides less than 1,636,630.
- Max.: The maximum number of total rides in a day is 2,049,519.

Interpretation:

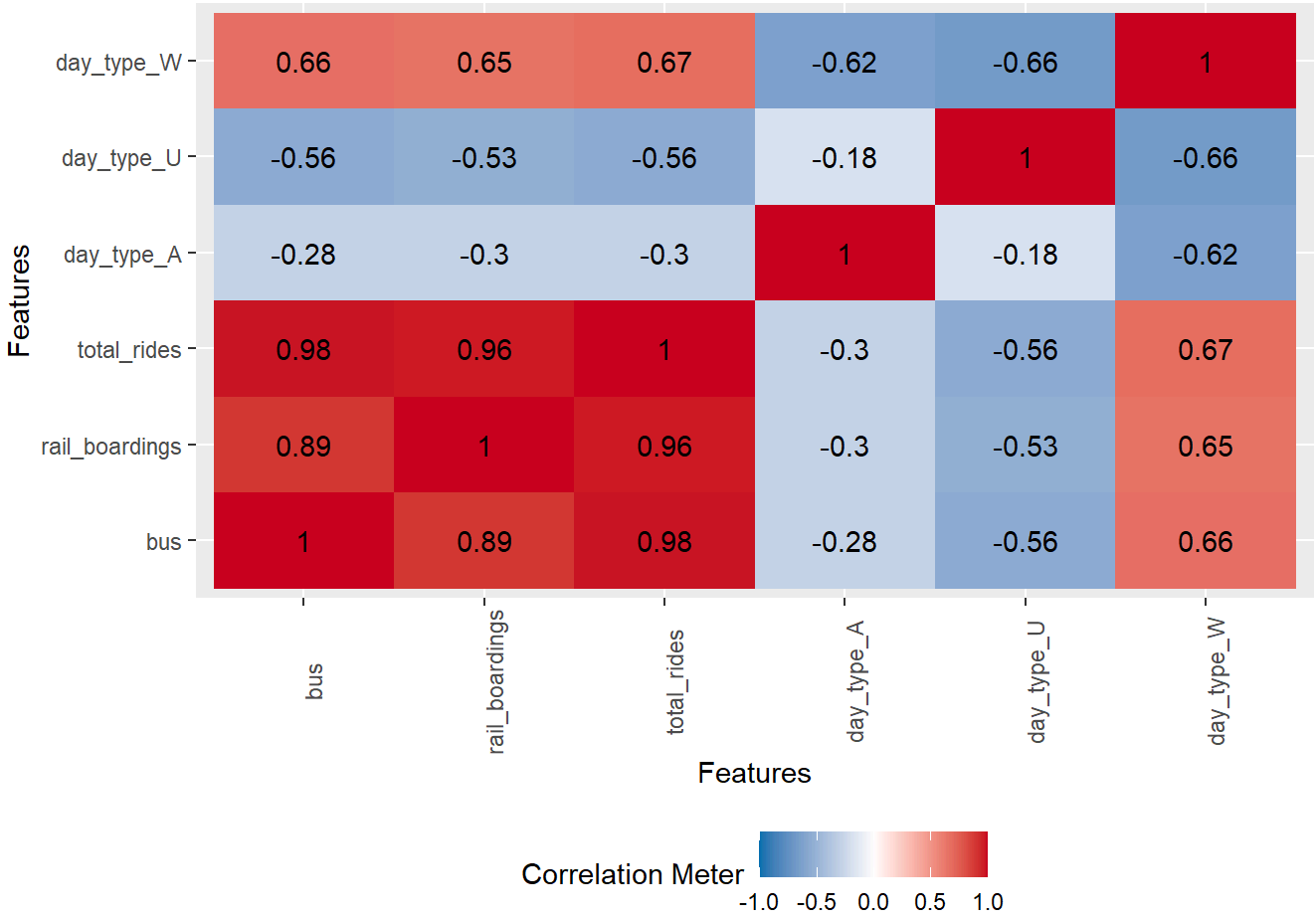
- service_date: The dataset spans from January 1, 2001, to February 29, 2024.
- day_type: This is a categorical variable indicating the type of day (e.g., weekday, weekend, holiday).
- bus, rail_boardings, total_rides: These columns represent the number of rides on buses, rail boardings, and the total number of rides, respectively, for each day in the dataset.

```
# Check for duplicates
sum(duplicated(df))
```

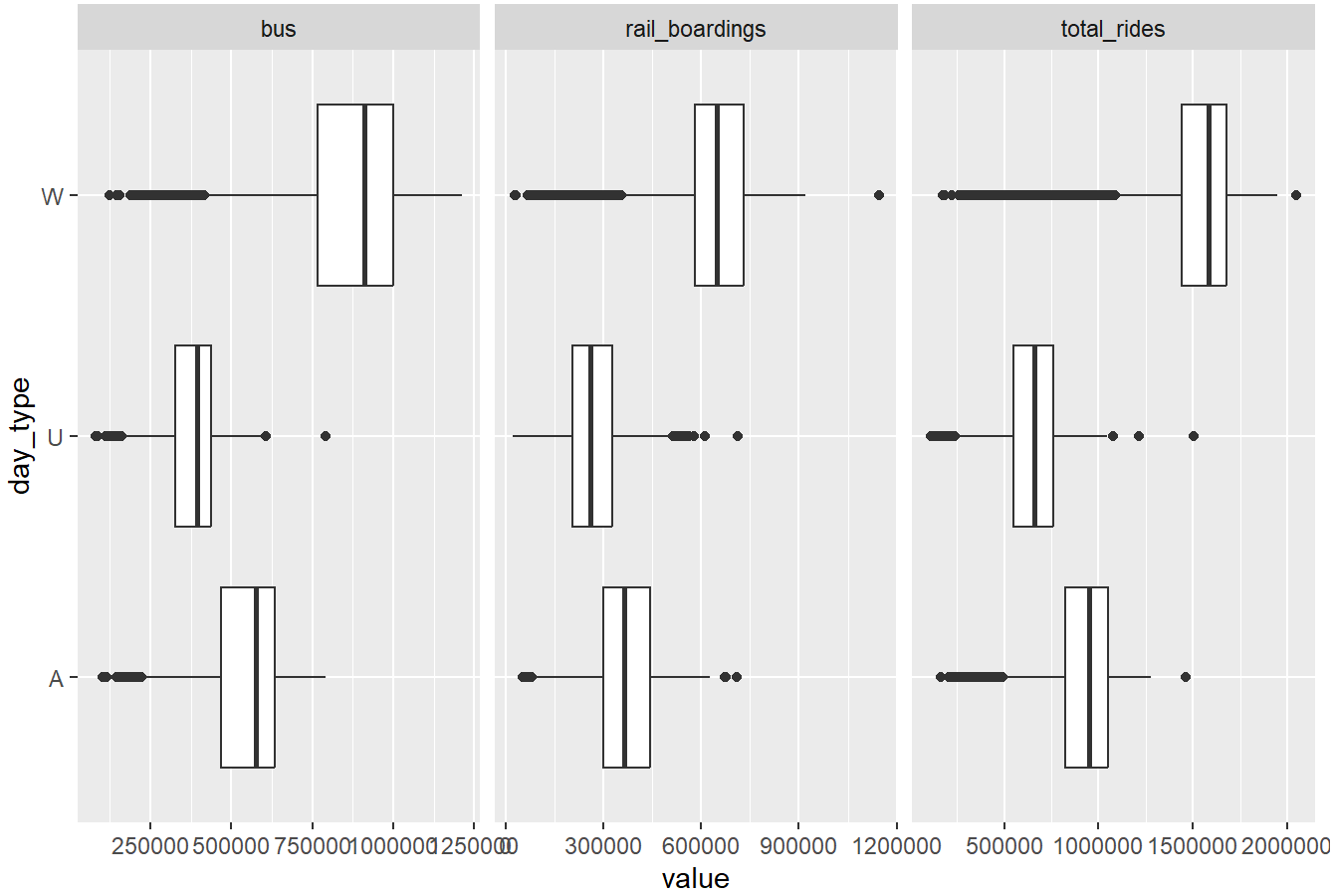
```
## [1] 62
```

```
# Correlation analysis
plot_correlation(df)
```

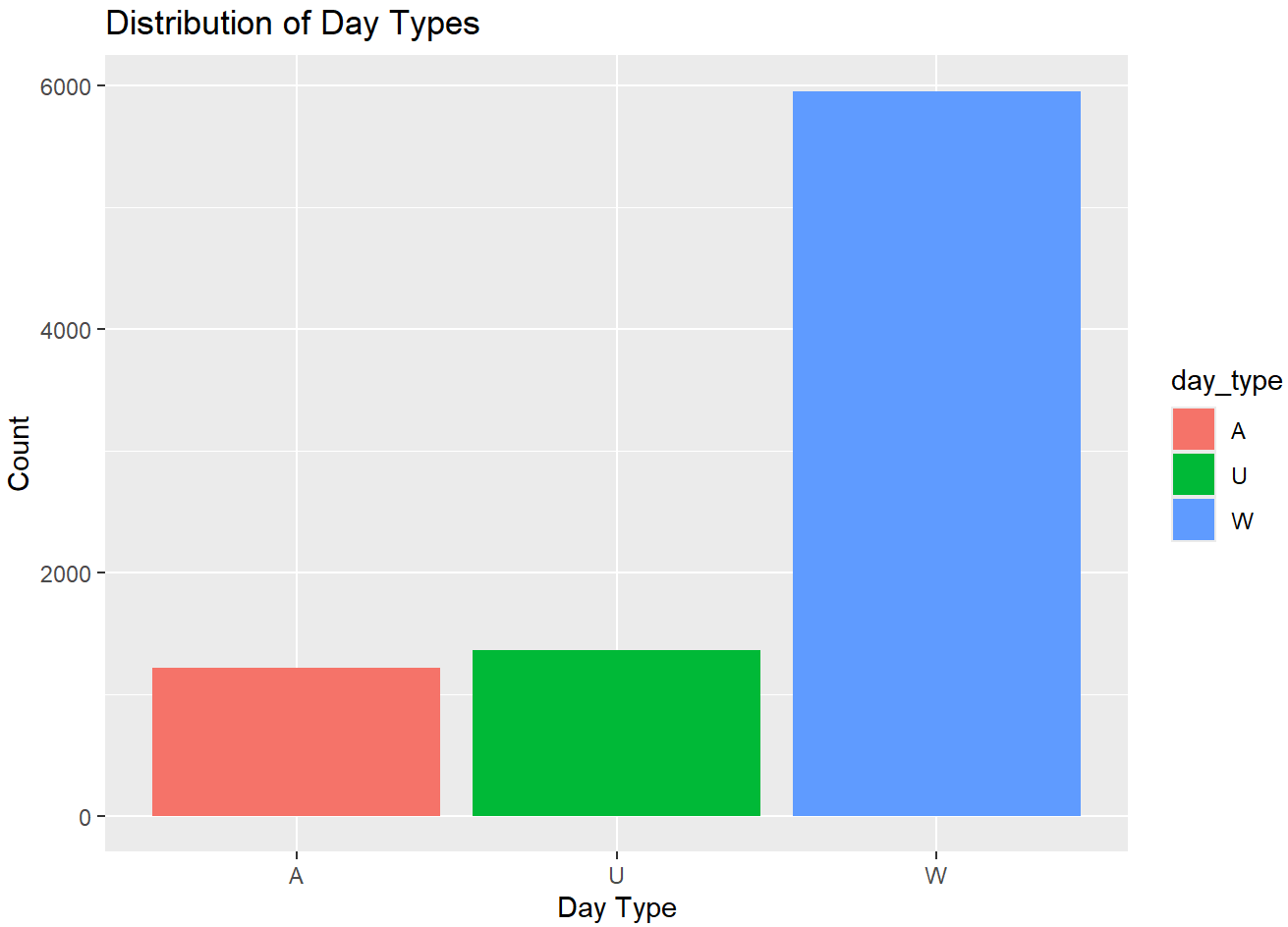
```
## 1 features with more than 20 categories ignored!
## service_date: 8460 categories
```



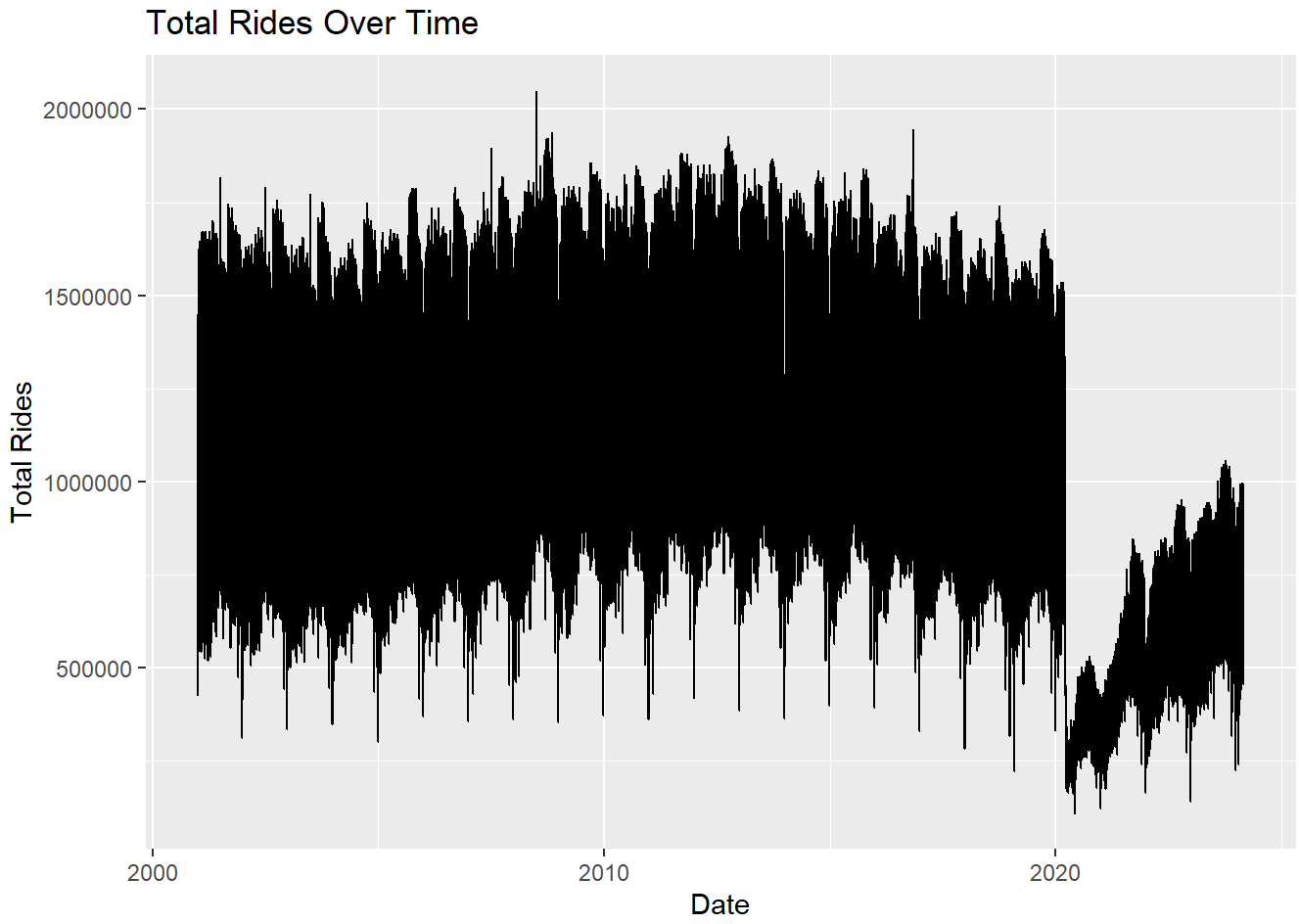

```
# Visualize the data
# Boxplots for numerical variables by day_type
plot_boxplot(df, by = "day_type")
```




```
# Analyze day_type distribution
df %>%
  count(day_type) %>%
  ggplot(aes(x = day_type, y = n, fill = day_type)) +
  geom_bar(stat = "identity") +
  labs(title = "Distribution of Day Types", x = "Day Type", y = "Count")
```

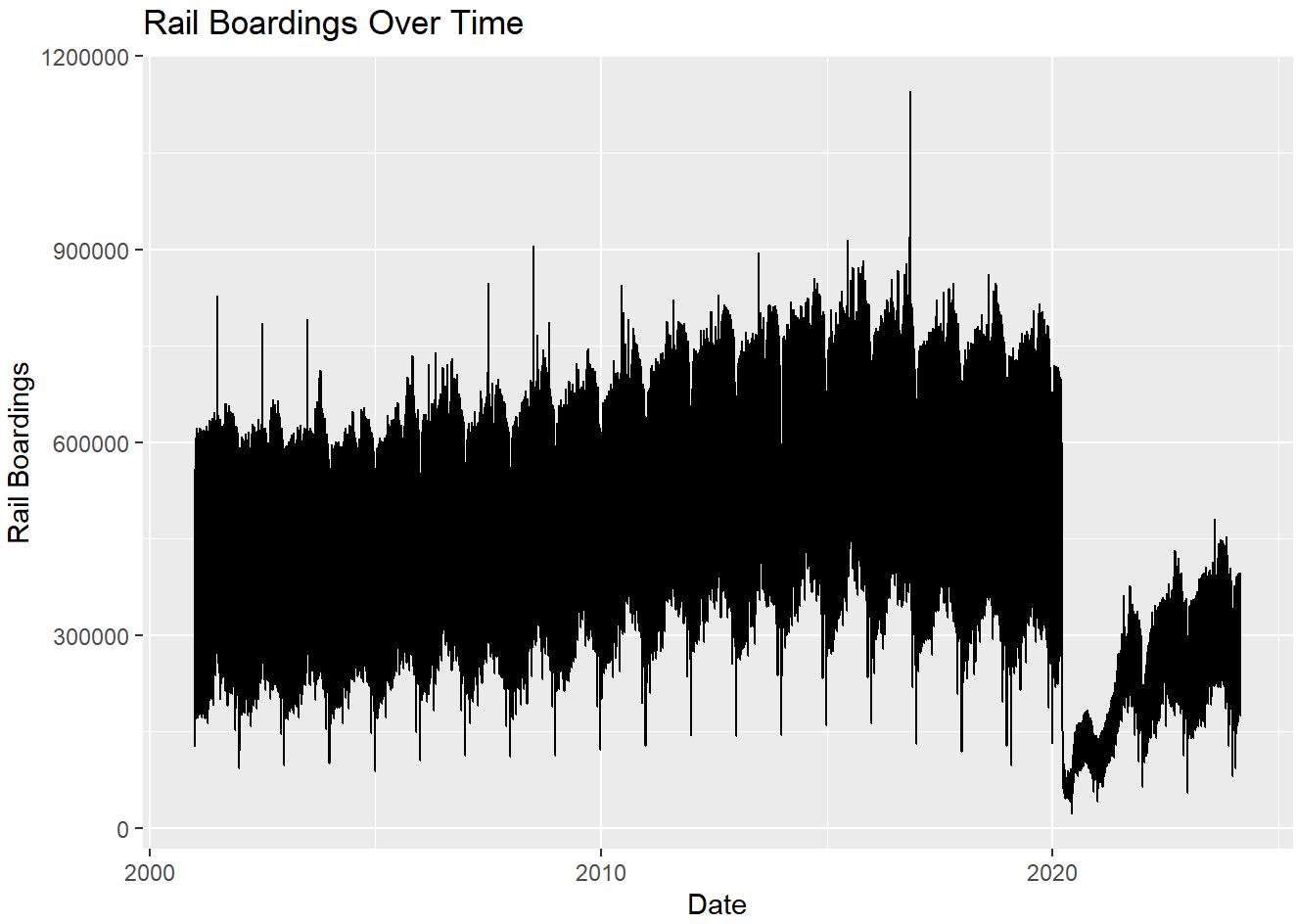



```
# Time series plot for total rides
ggplot(df, aes(x = service_date, y = total_rides)) +
  geom_line() +
  labs(title = "Total Rides Over Time", x = "Date", y = "Total Rides")
```

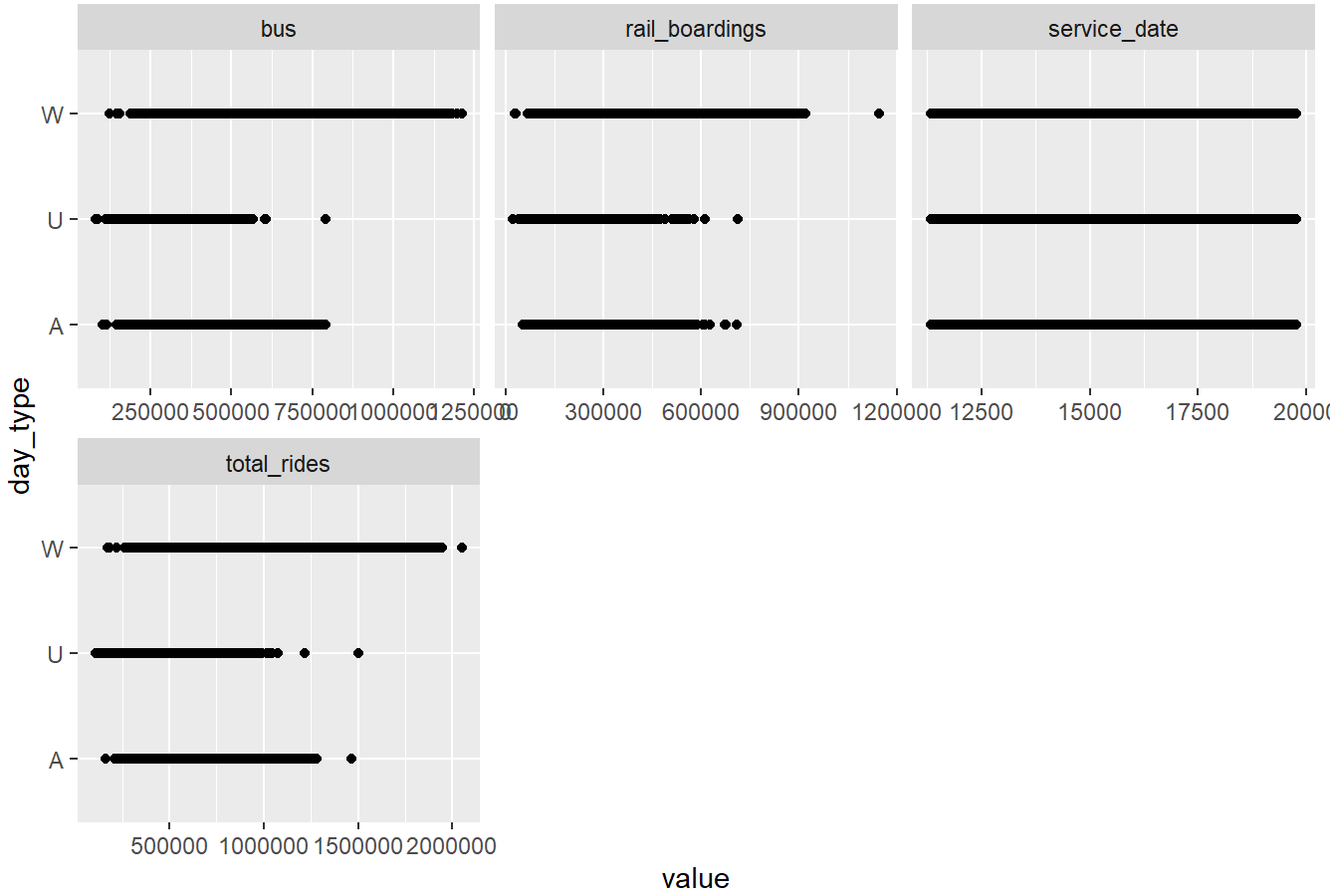



```
# Time series plot for bus rides
ggplot(df, aes(x = service_date, y = bus)) +
  geom_line() +
  labs(title = "Bus Rides Over Time", x = "Date", y = "Bus Rides")
```


```
# Time series plot for rail boardings
ggplot(df, aes(x = service_date, y = rail_boardings)) +
  geom_line() +
  labs(title = "Rail Boardings Over Time", x = "Date", y = "Rail Boardings")
```




```
# Advanced visualizations
plot_scatterplot(df, by = "day_type")
```



Piecewise Holt-Winters

Convert the date column: Ensure that the service_date column is converted to a Date type for proper time series handling.

```
# Filter the post-COVID period
post_covid_df <- df %>% filter(service_date >= as.Date("2020-03-01"))
```

Filter post-COVID data: Extract data from the post-COVID period starting March 1, 2020, since this period is more relevant for current analysis.

Split the dataset: Divide the dataset into a training set (up to the end of 2023) and a test set (2024).

```
# Split into training and test sets
train_df <- post_covid_df %>% filter(service_date < as.Date("2024-01-01"))
test_df <- post_covid_df %>% filter(service_date >= as.Date("2024-01-01"))
```

Create a time series object: Convert the training dataset into a time series object. The time series starts from the first date in the training set, and the frequency is set to 365 (daily data).

Fit the model: Apply the Holt-Winters exponential smoothing model to the training time series data. This model accounts for level, trend, and seasonal components.

Forecast future values: Use the fitted Holt-Winters model to forecast the total rides for the test period (2024).

Extract actual and forecasted values: Extract the actual values for 2024 from the test set and the forecasted values from the Holt-Winters model.

Calculate error metrics: Compute RMSE (Root Mean Square Error), MAE (Mean Absolute Error), and MAPE (Mean Absolute Percentage Error) to evaluate the accuracy of the forecasts.

Plot actual vs forecasted values: Create a plot to compare the actual total rides with the forecasted values.

Create a table of error metrics: Summarize the error metrics in a table for easy interpretation.

```
# Create a time series object
train_ts <- ts(train_df$total_rides, start = c(2020, as.numeric(format(min(train_df$service_dat
e), "%j"))), frequency = 365)

# Fit Holt-Winters model
hw_model <- HoltWinters(train_ts)

# Forecast for 2024
forecast_length <- nrow(test_df)
hw_forecast <- forecast(hw_model, h = forecast_length)

# Actual values for the test period
actual_values <- test_df$total_rides

# Forecasted values
forecasted_values <- hw_forecast$mean

# Calculate RMSE, MAE, and MAPE
rmse_value <- rmse(actual_values, forecasted_values)
mae_value <- mae(actual_values, forecasted_values)
mape_value <- mape(actual_values, forecasted_values)

# Print the values
rmse_value
```

```
## [1] 196500.1
```

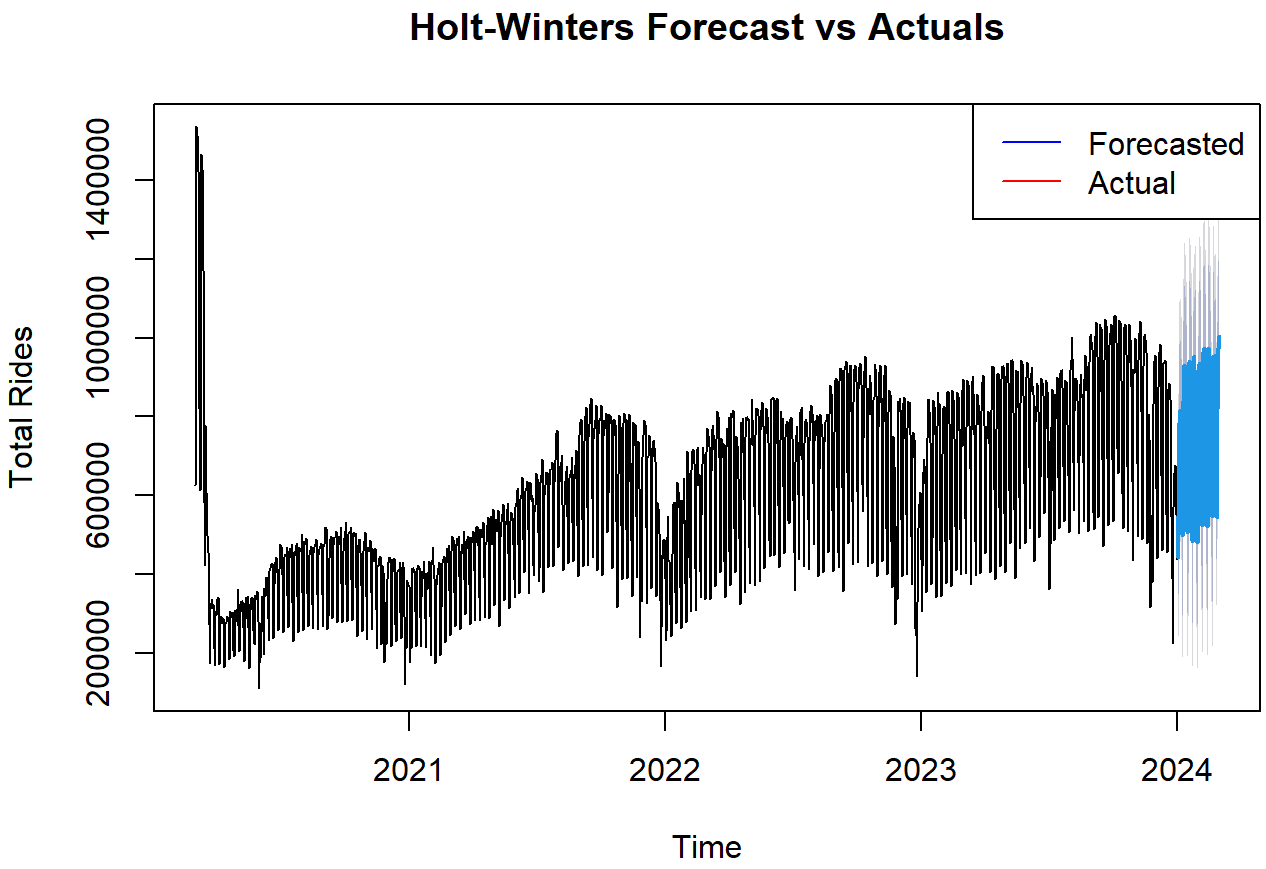
```
mae_value
```

```
## [1] 154657.8
```

```
mape_value
```

```
## [1] 0.2633082
```

```
# Plot actual vs forecasted values
plot(hw_forecast, main = "Holt-Winters Forecast vs Actuals", xlab = "Time", ylab = "Total Ride
s")
lines(test_df$service_date, actual_values, col = "red", lwd = 2)
legend("topright", legend = c("Forecasted", "Actual"), col = c("blue", "red"), lty = 1)
```




```
# Create a data frame for the error metrics
error_metrics <- data.frame(
  Metric = c("RMSE", "MAE", "MAPE"),
  Value = c(rmse_value, mae_value, mape_value)
)

# Print the error metrics table
print(error_metrics)
```

```
##   Metric        Value
## 1   RMSE 1.965001e+05
## 2    MAE 1.546578e+05
## 3   MAPE 2.633082e-01
```

The provided plot compares the forecasted values using the Holt-Winters model against the actual values of total rides over time. Here's a detailed explanation:

Time Series Data:

- The x-axis represents the time period from 2020 to 2024.
- The y-axis represents the total number of rides.

Data Points:

- Black Line: Represents the actual total rides data from 2020 to the end of 2023.
- Blue Line and Shaded Area: Represents the forecasted total rides for the year 2024, with the shaded area indicating the confidence intervals around the forecast.
- Red Line: Represents the actual total rides data for the year 2024.

Pre-COVID and Post-COVID Trends:

- The initial sharp drop in total rides around early 2020 corresponds to the onset of the COVID-19 pandemic, which significantly impacted public transportation usage.
- Following the initial drop, there is a gradual recovery and fluctuation in total rides, reflecting changing patterns and gradual recovery over time.

Forecasting Post-COVID:

- The forecast for 2024 shows a continued pattern, accounting for seasonal variations and trends observed in the post-COVID period.
- The confidence intervals around the forecasted values provide an estimate of the uncertainty in the predictions.

Model Evaluation:

- The red line for actual values in 2024 is within the confidence intervals of the forecast, indicating that the model captures the overall trend and seasonality reasonably well.
- Comparing the red line (actual values) with the blue line (forecasted values) helps in evaluating the accuracy of the Holt-Winters model.

Error Metrics:

- RMSE: The forecast errors have a standard deviation of approximately 196,500 rides.
- MAE: The average absolute error in the forecasted total rides is about 154,658 rides.
- MAPE: The average absolute percentage error is approximately 26.33%, indicating that the forecasted values are, on average, off by about 26.33% from the actual values.

The plot effectively demonstrates the ability of the Holt-Winters model to forecast ridership trends post-COVID, with actual 2024 data aligning reasonably well with the forecasted values. The confidence intervals give an indication of the forecast's reliability, and the error metrics would further substantiate the model's accuracy. This visual comparison and quantitative evaluation are crucial for understanding the model's performance and reliability in making future predictions. The error metrics collectively suggest the model's accuracy and can help identify areas for improvement. Lower values for these metrics generally indicate a more accurate model, but each metric provides unique insights into the nature of the forecast errors.