Data Preparation

The first step involves loading the dataset and preparing it for analysis by removing unnecessary columns.

```
# Load dataset
df <- read.csv("data_table.csv")
View(df)

# Remove unnecessary columns
df2 <- df[, -c(1:5)]
```

Splitting the Data

The dataset is divided into training (75%) and testing (25%) subsets.

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.4.2
```

```
## randomForest 4.7-1.2
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
# Split the data
train <- df2[1:3269, ]
test <- df2[-(1:3269), ]

# Verify split sizes
cat("Training rows:", nrow(train), "\n")
```

```
## Training rows: 3269
```

```
cat("Testing rows:", nrow(test), "\n")
```

```
## Testing rows: 1089
```

Random Forest Model

A Random Forest model is trained to predict the number of orders. The R^2 and RMSE metrics are computed to evaluate performance.

```
set.seed(78)

# Train Random Forest model
ranFor <- randomForest(Orders ~ ., data = train)

# Predict on test data
pred2 <- predict(ranFor, newdata = test)

# Calculate metrics
SST <- sum((test$Orders - mean(test$Orders))^2)
SSE <- sum((test$Orders - pred2)^2)
R2 <- 1 - SSE / SST
RMSE <- sqrt(SSE)

# Display metrics
cat("R^2:", R2, "\n")
```

```
## R^2: 0.8772429
```

```
cat("RMSE:", RMSE, "\n")
```

```
## RMSE: 949.7882
```

Save Predictions

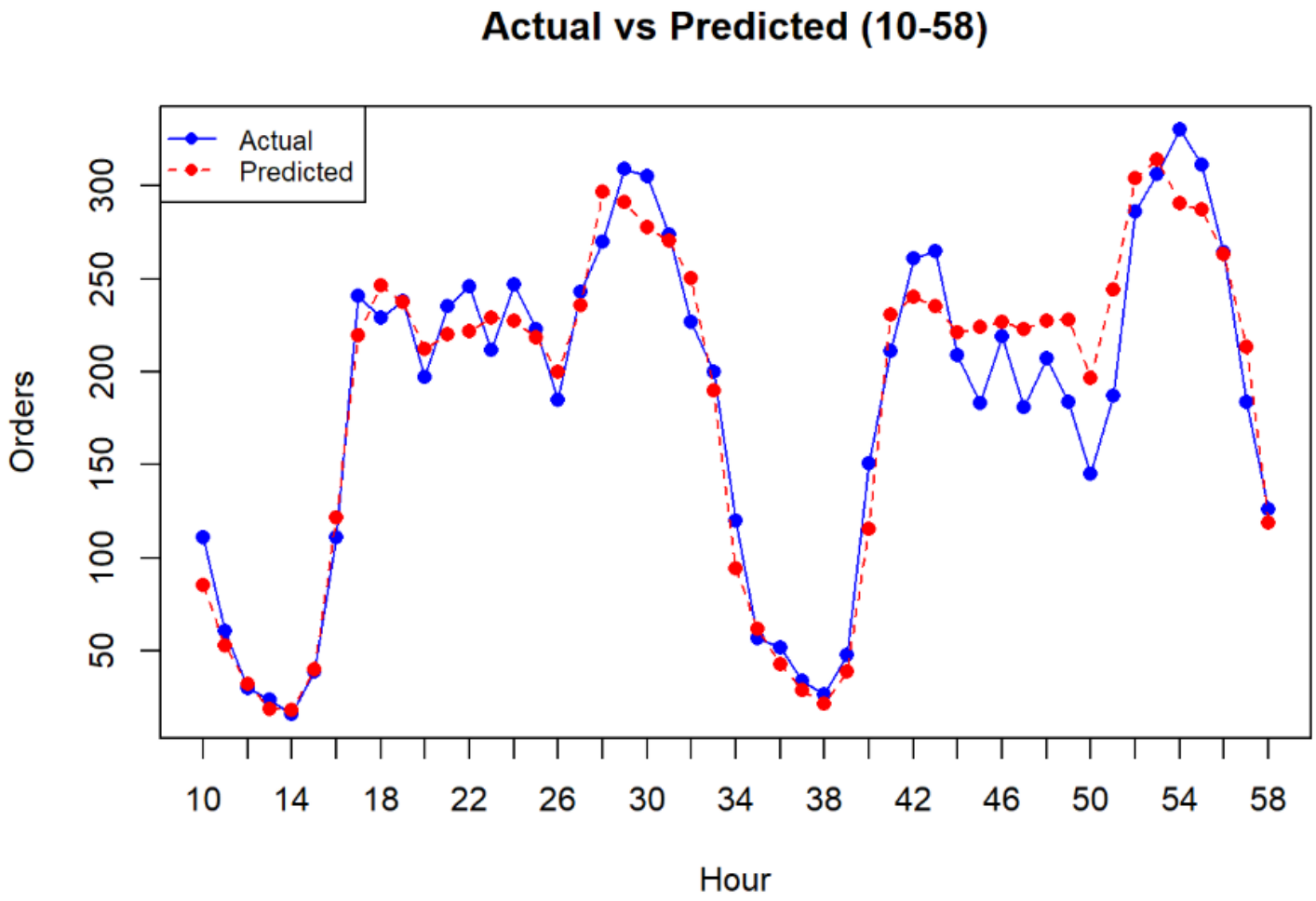
The predicted and actual values are saved to a CSV file for further analysis.

```
# Save predictions
df_results <- data.frame(Actual = test$Orders, Predicted = pred2)
View(df_results)
write.csv(df_results, "Actual_vs_Pred_RanFor.csv")
```

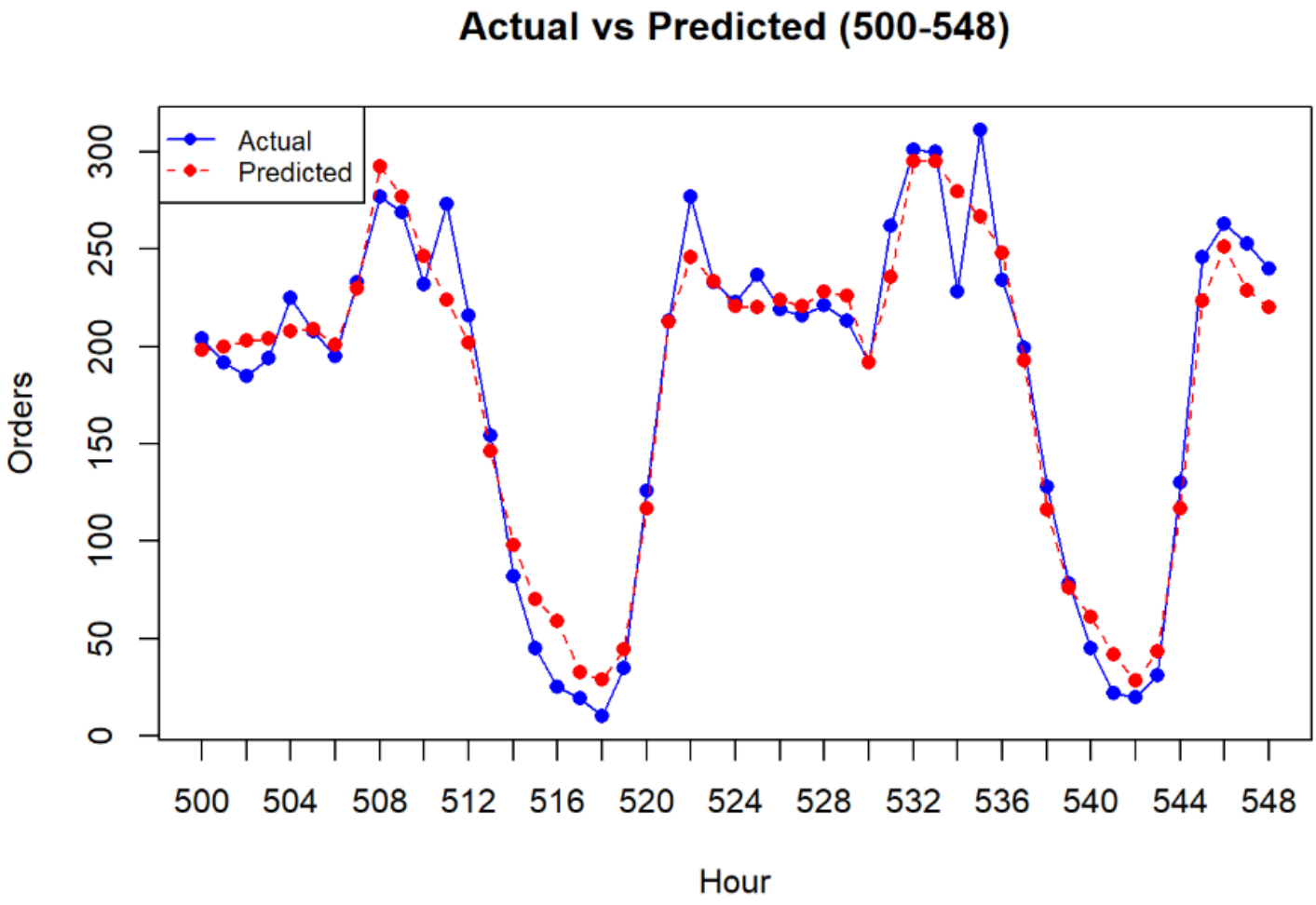
Visualization: Actual vs Predicted

The following plots compare actual and predicted orders over specific time intervals.

```
plot(10:58, test$Orders[10:58], type = "o", col = "blue", pch = 16, lty = 1, xaxt = "n",
     xlab = "Hour", ylab = "Orders", main = "Actual vs Predicted (10-58)")
lines(10:58, pred2[10:58], type = "o", col = "red", pch = 16, lty = 2)
axis(1, at = seq(10, 58, by = 2), labels = seq(10, 58, by = 2))
legend("topleft", legend = c("Actual", "Predicted"), col = c("blue", "red"),
       lty = c(1, 2), pch = 16, cex = 0.8)
```




```
plot(500:548, test$Orders[500:548], type = "o", col = "blue", pch = 16, lty = 1, xaxt = "n",
     xlab = "Hour", ylab = "Orders", main = "Actual vs Predicted (500-548)")
lines(500:548, pred2[500:548], type = "o", col = "red", pch = 16, lty = 2)
axis(1, at = seq(500, 548, by = 2), labels = seq(500, 548, by = 2))
legend("topleft", legend = c("Actual", "Predicted"), col = c("blue", "red"),
       lty = c(1, 2), pch = 16, cex = 0.8)
```



Conclusion

The Random Forest model effectively predicts taxi orders, achieving an R^2 value of 0.8772429 and an RMSE of 949.7882492. Further optimization may involve hyperparameter tuning or feature engineering. “‘

Key Features:

1. **Sections**: Organized into *Data Preparation*, *Splitting the Data*, *Model Training*, *Visualization*, and *Conclusion*.
2. **Code Chunks**: Includes R code chunks (`{r}`) for clarity and reproducibility.
3. **Metrics Display**: Displays R^2 and RMSE directly within the document.
4. **Plots**: Visual comparisons of actual and predicted values.