CASE STUDY

NYC Yellow Taxi Fare Prediction using Linear & Multiple Linear Regression

==

1. Business Background

==

New York City's Taxi & Limousine Commission (TLC) collects detailed data for every taxi trip.

The dataset contains millions of real taxi rides including distance, passenger count,

timestamps, and fare amount.

Problem Statement:

Taxi fares vary widely even for similar trips.

Can historical data be used to understand what drives taxi fares and estimate fares accurately?

==

2. Objectives

==

1. Study relationship between trip distance and fare amount
2. Build Simple Linear Regression model
3. Apply Multiple Linear Regression
4. Evaluate model performance
5. Validate real-world usability

==

3. Dataset Description

==

Dataset: NYC Yellow Taxi Trip Data (Kaggle)

Key Columns:

- trip_distance
- fare_amount
- passenger_count

==

4. Data Loading & Cleaning (Code)

==

```
import pandas as pd

# Load dataset
df = pd.read_csv("yellow_tripdata.csv")

# Select relevant columns
data = df[['trip_distance', 'fare_amount', 'passenger_count']]

# Remove invalid values
data = data[(data['trip_distance'] > 0) & (data['fare_amount'] > 0)]

# View cleaned data
print(data.head())
print(data.describe())
```

==

5. Correlation Analysis

==

Correlation matrix

corr = data.corr()

print(corr)

Interpretation:

- Correlation between trip_distance and fare_amount ≈ 0.95
- Indicates a very strong positive relationship
- Linear Regression is suitable

==

6. Simple Linear Regression

==

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score

import numpy as np

Feature and target

X = data[['trip_distance']]

y = data['fare_amount']

```
# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

# Model training
model_simple = LinearRegression()
model_simple.fit(X_train, y_train)

# Prediction
y_pred = model_simple.predict(X_test)

# Evaluation
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)

print("Slope:", model_simple.coef_[0])
print("Intercept:", model_simple.intercept_)
print("MSE:", mse)
print("MAE:", mae)
print("RMSE:", rmse)
print("R2:", r2)
```

Model Output:

- Slope   ≈ 2.74

- Intercept ≈ 4.20

- R^2 ≈ 0.91

Interpretation:

- Fare increases by ~2.7 units per unit distance

- Distance explains 91% of fare variation

==

7. Multiple Linear Regression

==

```
# Multiple features
X_multi = data[['trip_distance', 'passenger_count']]
y = data['fare_amount']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X_multi, y, test_size=0.2, random_state=42
)

# Model training
model_multi = LinearRegression()
model_multi.fit(X_train, y_train)

# Prediction
```

```
y_pred_multi = model_multi.predict(X_test)

# Evaluation
print("Coefficients:", model_multi.coef_)
print("Intercept:", model_multi.intercept_)
print("R2:", r2_score(y_test, y_pred_multi))
```

Interpretation:

- Trip distance is dominant predictor
- Passenger count has negligible impact
- R^2 does not significantly improve

==

8. Visualization & Assumption Check

==

```
import matplotlib.pyplot as plt

# Scatter plot with regression line
plt.figure(figsize=(8,6))
plt.scatter(X_test['trip_distance'], y_test, alpha=0.3)
plt.plot(X_test['trip_distance'], y_pred, color='red')
plt.xlabel("Trip Distance")
plt.ylabel("Fare Amount")
plt.title("Trip Distance vs Fare Amount")
plt.show()
```

```
# Residual plot
residuals = y_test - y_pred
plt.figure(figsize=(8,6))
plt.scatter(y_pred, residuals, alpha=0.3)
plt.axhline(0, color='red')
plt.xlabel("Predicted Fare")
plt.ylabel("Residuals")
plt.title("Residual Plot")
plt.show()
```

Observation:

- Linear trend observed
- Residuals randomly scattered
- Assumptions reasonably satisfied

==

9. Model Evaluation Metrics

==

- MAE ≈ 1.46
- RMSE ≈ 2.88
- R^2 ≈ 0.91

These indicate strong predictive performance.

==

10. Final Business Summary

===

- Trip distance is the strongest predictor of taxi fare
- Simple Linear Regression performs exceptionally well
- Multiple Linear Regression adds minimal improvement
- Model is accurate, interpretable, and practical
- Suitable for real-world fare estimation

===

11. Conclusion

===

This case study demonstrates how real-world transportation data can be used to build effective machine learning models. A simple distance-based regression model explains most fare variation and is suitable for analytical and operational use.

=================== END OF CASE STUDY ====================