UNIFIED MENTOR
YOUR SKILL, SUCCESS & JOURNEY

Uber Trip Analysis - 2014

Predictive Modeling & Data Insights

Presented by: Manju Suthar
Internship : Unofied mentor.pvt.ltd/Data Science

Agenda:

- Introduction
- Objective
- Dataset
- Data Preprocessong
- Exploratory Data Analysis
- Feature Engineering
- Model Building
- Final Takeaways & Conclusion
- Future Scope & Improvements

Introduction:

- This project focuses on analyzing Uber ride data from New York City in 2014 to uncover meaningful patterns in trip demand across different hours and days. By exploring and preprocessing the data, and applying machine learning models like Random Forest, Gradient Boosting, and XGBoost, the goal is to predict ride volume and compare model performance. The insights gained from this analysis can help ride-hailing services improve driver allocation, pricing strategies, and overall service efficiency.

Objective :

1. **Explore and preprocess Uber trip data.**
2. **Train and evaluate predictive models (XGBoost, GBTR, Random Forest).**
3. **Use ensemble techniques to improve accuracy.**
4. **Compare model performance using MAPE.**

Dataset :

Date/Time	Lat	Lon	Base
04-01-2014 00:11	40.769	-73.9549	B02512
04-01-2014 00:17	40.7267	-74.0345	B02512
04-01-2014 00:21	40.7316	-73.9873	B02512
04-01-2014 00:28	40.7588	-73.9776	B02512
04-01-2014 00:33	40.7594	-73.9722	B02512
04-01-2014 00:33	40.7383	-74.0403	B02512
04-01-2014 00:39	40.7223	-73.9887	B02512
04-01-2014 00:45	40.762	-73.979	B02512
04-01-2014 00:55	40.7524	-73.996	B02512
04-01-2014 01:01	40.7575	-73.9846	B02512
04-01-2014 01:19	40.7256	-73.9869	B02512
04-01-2014 01:48	40.7591	-73.9684	B02512
04-01-2014 01:49	40.7271	-73.9803	B02512
04-01-2014 02:11	40.6463	-73.7896	B02512
04-01-2014 02:25	40.7564	-73.9167	B02512
04-01-2014 02:31	40.7666	-73.9531	B02512
04-01-2014 02:43	40.758	-73.9761	B02512

The dataset used in this project was obtained through a Freedom of Information Law (FOIL) request by FiveThirtyEight from the NYC Taxi & Limousine Commission (TLC). It includes detailed trip-level data on over 4.5 million Uber pickups in New York City from April to September 2014. The dataset provides valuable insights into pickup times, dates, and base locations. Originally used for journalistic analysis, it serves as a rich resource for exploring ride demand patterns and building predictive models in real-world urban mobility scenarios.

Understanding Of Data :

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 564516 entries, 0 to 564515
Data columns (total 4 columns):
 #   Column     Non-Null Count   Dtype
---  ------     --------------   -----
 0   Date/Time  564516 non-null  object
 1   Lat        564516 non-null  float64
 2   Lon        564516 non-null  float64
 3   Base       564516 non-null  object
dtypes: float64(2), object(2)
memory usage: 17.2+ MB
None
```

Data Preprocessing :

```
data['Hour'] = data['Date/Time'].dt.hour
data['Day'] = data['Date/Time'].dt.day
data['DayOfWeek'] = data['Date/Time'].dt.dayofweek
data['Month'] = data['Date/Time'].dt.month
```

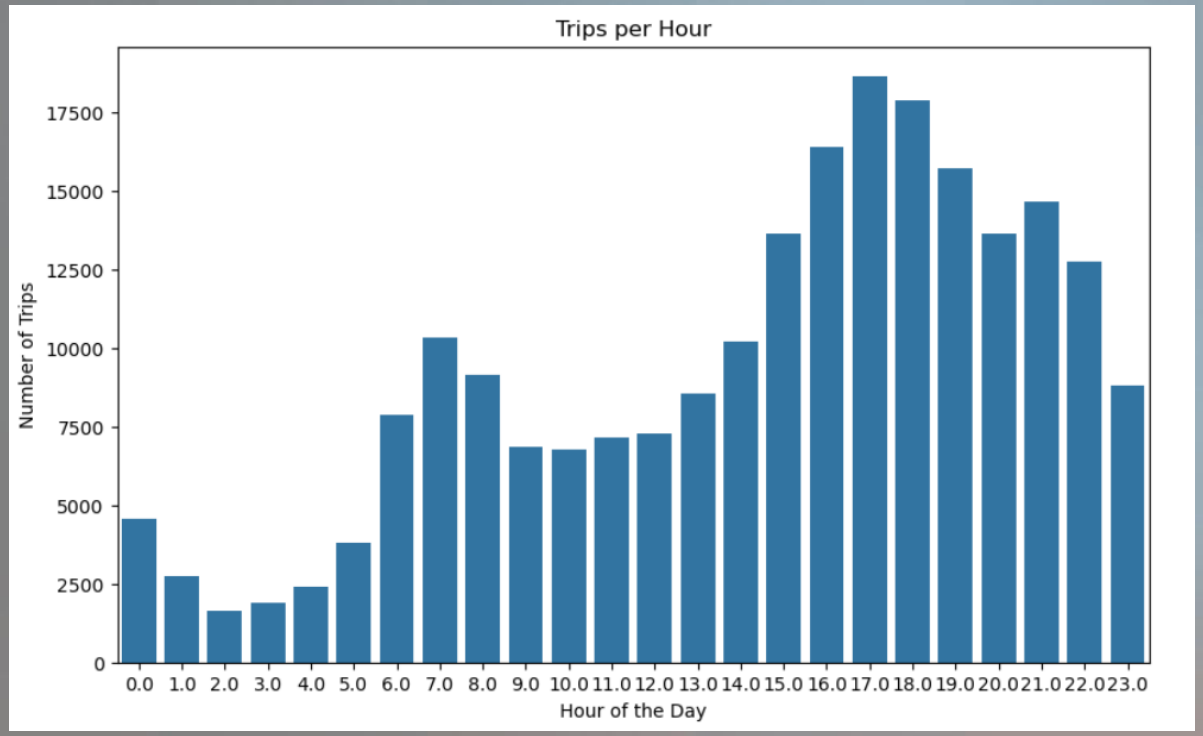
```
print(data[['Date/Time', 'Hour', 'Day', 'DayOfWeek', 'Month']].head())
```

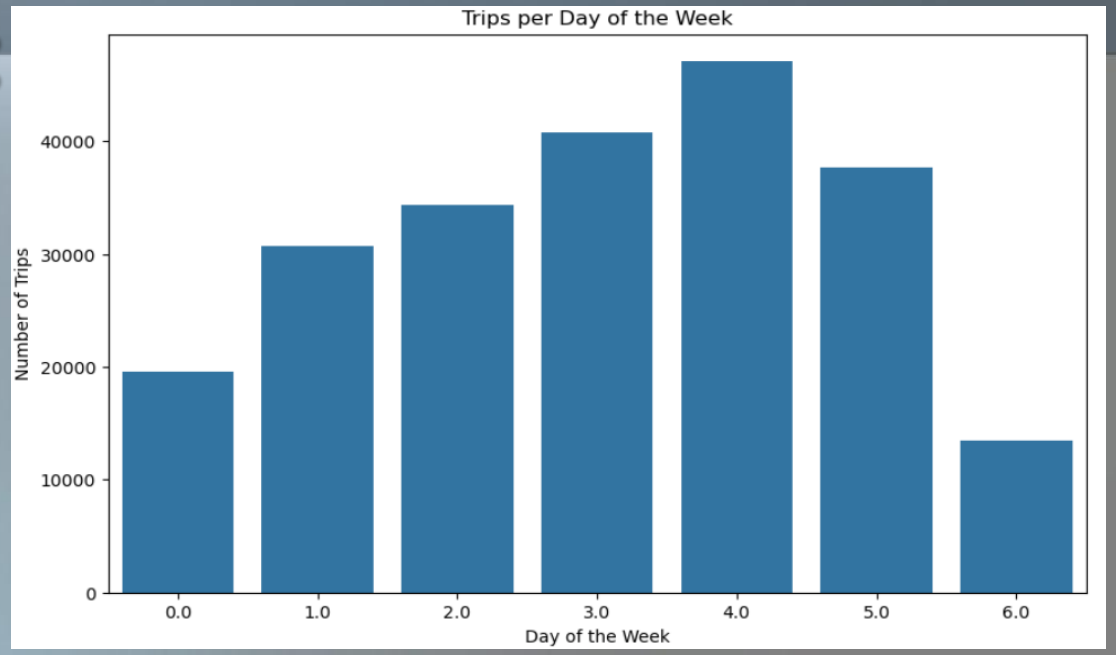
```
            Date/Time  Hour  Day  DayOfWeek  Month
0 2014-04-01 00:11:00   0.0  1.0        1.0    4.0
1 2014-04-01 00:17:00   0.0  1.0        1.0    4.0
2 2014-04-01 00:21:00   0.0  1.0        1.0    4.0
3 2014-04-01 00:28:00   0.0  1.0        1.0    4.0
4 2014-04-01 00:33:00   0.0  1.0        1.0    4.0
```

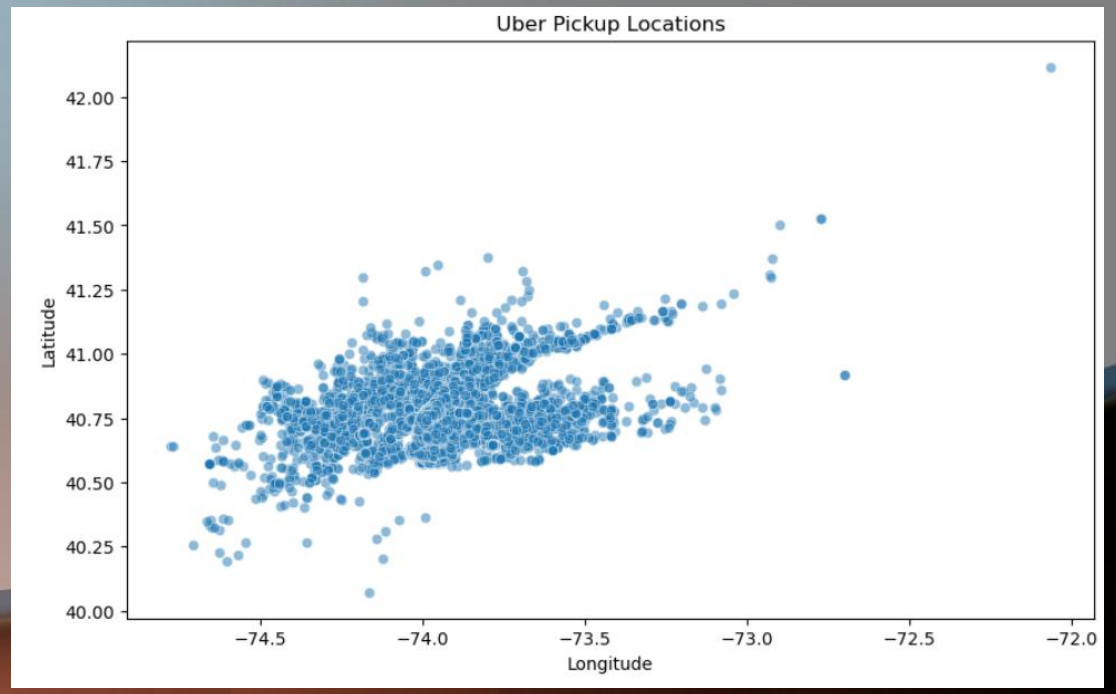
```
print(data.isnull().sum())
```

```
Date/Time    0
Lat          0
Lon          0
Base         0
dtype: int64
```

Exploratory Data Analysis:







Insights :

- **Hourly Trips:** Uber trips are lowest from midnight to 5 AM, rise sharply after 6 AM, and peak between 5–6 PM, with another smaller peak around 7–8 AM. Demand drops after 7 PM but stays higher than early morning levels.
- **Weekly Trends:** Trips increase from the start of the week, peaking mid-to-late week (likely Thursday/Friday), and drop on weekends, with the lowest on Sunday.
- **Location Patterns:** Most pickups are clustered in a dense metro area, likely NYC, with fewer in surrounding regions. Peak commute times, especially evenings, show highest demand—helpful for optimizing driver supply and pricing.

Feature Engineering

```
# Redefining the data frame for model training
data = data.groupby(['Hour', 'Day', 'Month', 'DayOfWeek']).size().reset_index(name='TripCount')
```

```
# Define features (X) and target variable (y)
X = data[['Hour', 'Day', 'Month', 'DayOfWeek']]
y = data['TripCount']
```

Model Building & Training

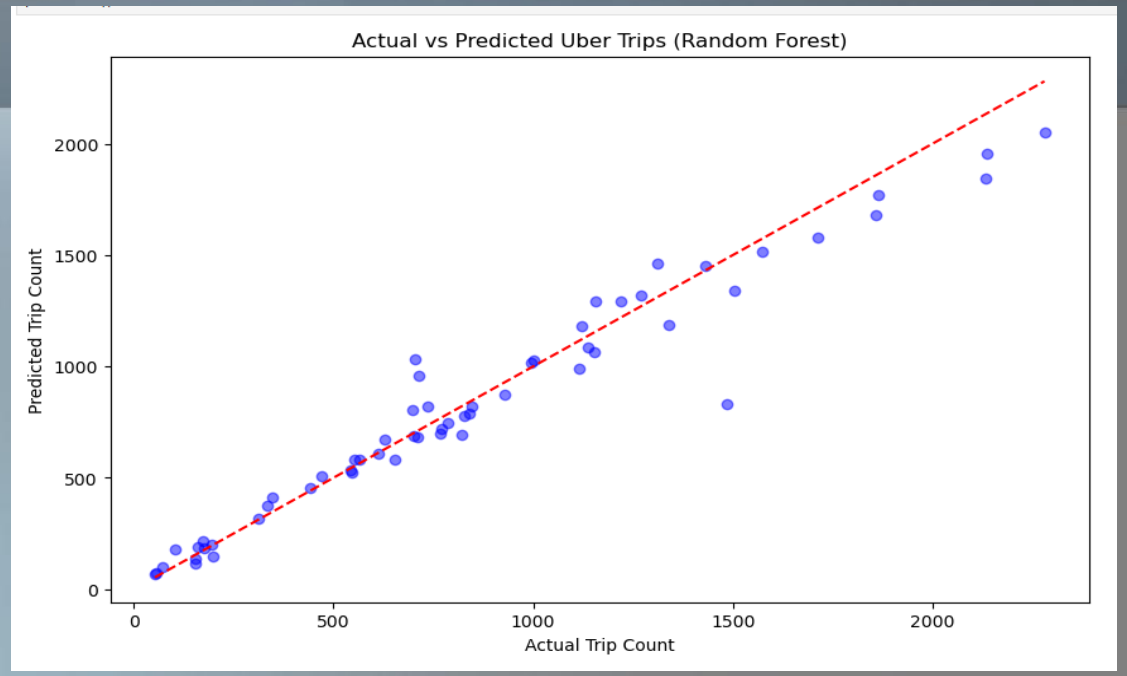
Trained Models:

1. Random Forest Regressor
2. XGBoost (with GridSearchCV)
3. Gradient Boosting Regressor (tuned similarly)

Model Performance

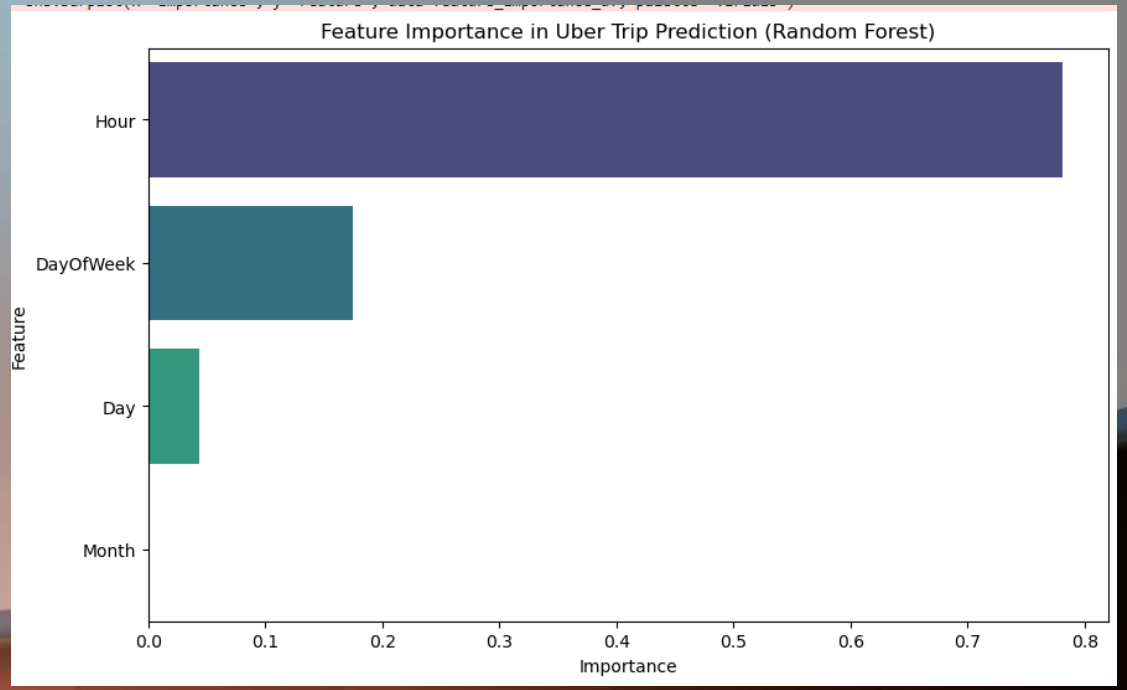
➢ Model 1 - Random Forest

```
Random Forest - Mean Squared Error: 17933.226646551724
Random Forest - R^2 Score: 0.9436420589492389
```



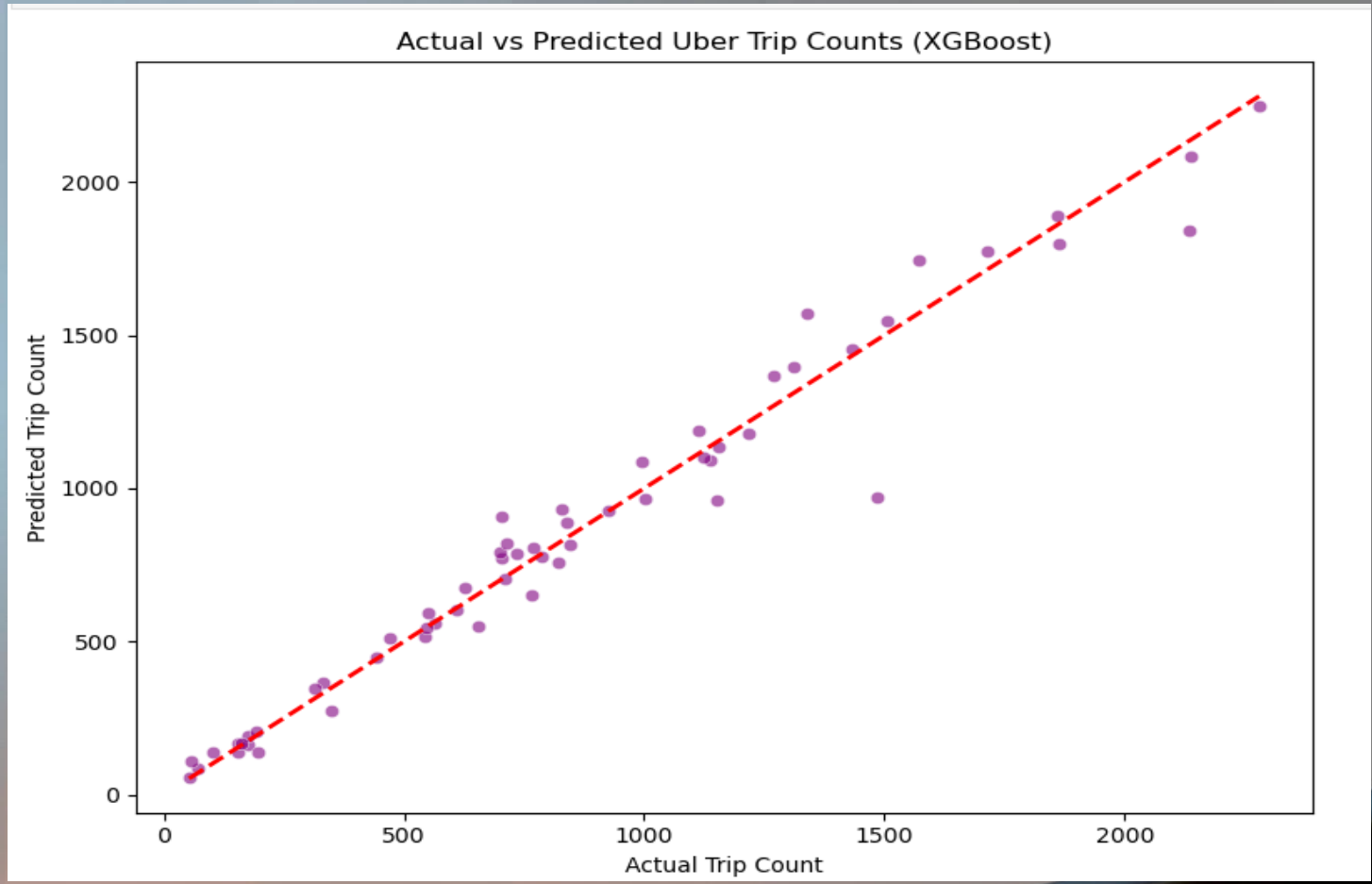
➢ Conclusion:

- Good base performance.
- The scatter plot shows that the predicted Uber trip counts from the Random Forest model closely align with the actual values, indicating strong predictive performance.
- Feature Importance Bar Chart – highlights which features the model found most useful.



➢ Model 2 – XGBoost

```
XGBoost - Mean Squared Error: 11272.175810693017
XGBoost - R^2 Score: 0.9645754424246344
```

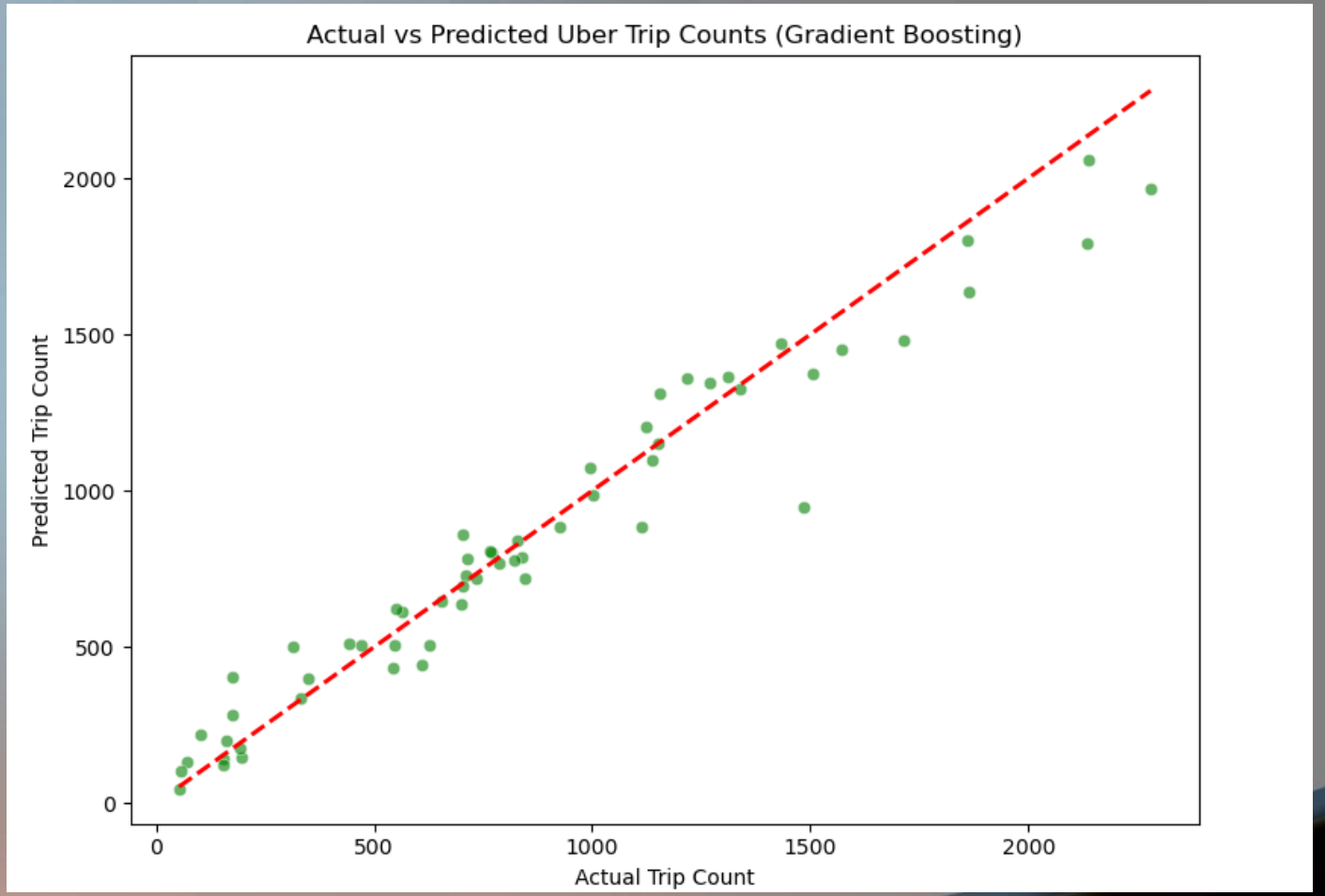


➢ Conclusion:

- Good base performance.
- The XGBoost model demonstrates excellent predictive accuracy, with predictions closely aligning with actual Uber trip counts.

➢ Model 3 – Gradient Boosting

```
Gradient Boosting - Mean Squared Error: 17785.374543070087
Gradient Boosting - R^2 Score: 0.9441067070740007
```



➢ Conclusion:

- Good base performance.
- The Gradient Boosting model accurately captures the trend in Uber trip counts, with predictions closely following the actual values.

Hyperparameter Tuning (XGBoost & Gradient Boosting)

Method Used:

- **GridSearchCV** with **TimeSeriesSplit** was used to perform hyperparameter tuning.
- Goal: Find the best combination of parameters for **XGBoost** and **Gradient Boosting** models.
- We applied **GridSearchCV** with **TimeSeriesSplit** to ensure no data leakage in our time-based dataset.
  The objective was to find optimal parameters for both **XGBoost** and **Gradient Boosting** models.
  Below are the best hyperparameter combinations, which were then used to train and evaluate the final models.

🧪 Define Param Grid → 🌀 GridSearchCV + TimeSeriesSplit →
📈 Best Params → ✅ Final Model

```
Fitting 5 folds for each of 243 candidates, totalling 1215 fits
Best XGBoost Parameters: {'colsample_bytree': 1.0, 'learning_rate': 0.1, 'max_depth': 6, 'n_estimators': 300, 'subsample': 0.8}
```

Model	n_estimators	learning_rate	max_depth	subsample	colsample_bytree
XGBoost	300	0.1	6	0.8	1.0
Gradient Boost	200	0.1	4	0.8	-

Tuned Models Performance

```
Fitting 5 folds for each of 36 candidates, totalling 180 fits
Tuned GBR - Best Parameters: {'learning_rate': 0.1, 'max_depth': 4, 'n_estimators': 200, 'subsample': 0.8}
Tuned GBR - MSE: 9001.952504707331
Tuned GBR - R²: 0.9717099706259722
```

- Parameter grid used (n_estimators, learning_rate, etc.)
- Best parameters found:

bash
{'learning_rate': 0.1, 'max_depth': 4, 'n_estimators': 200, 'subsample': 0.8}

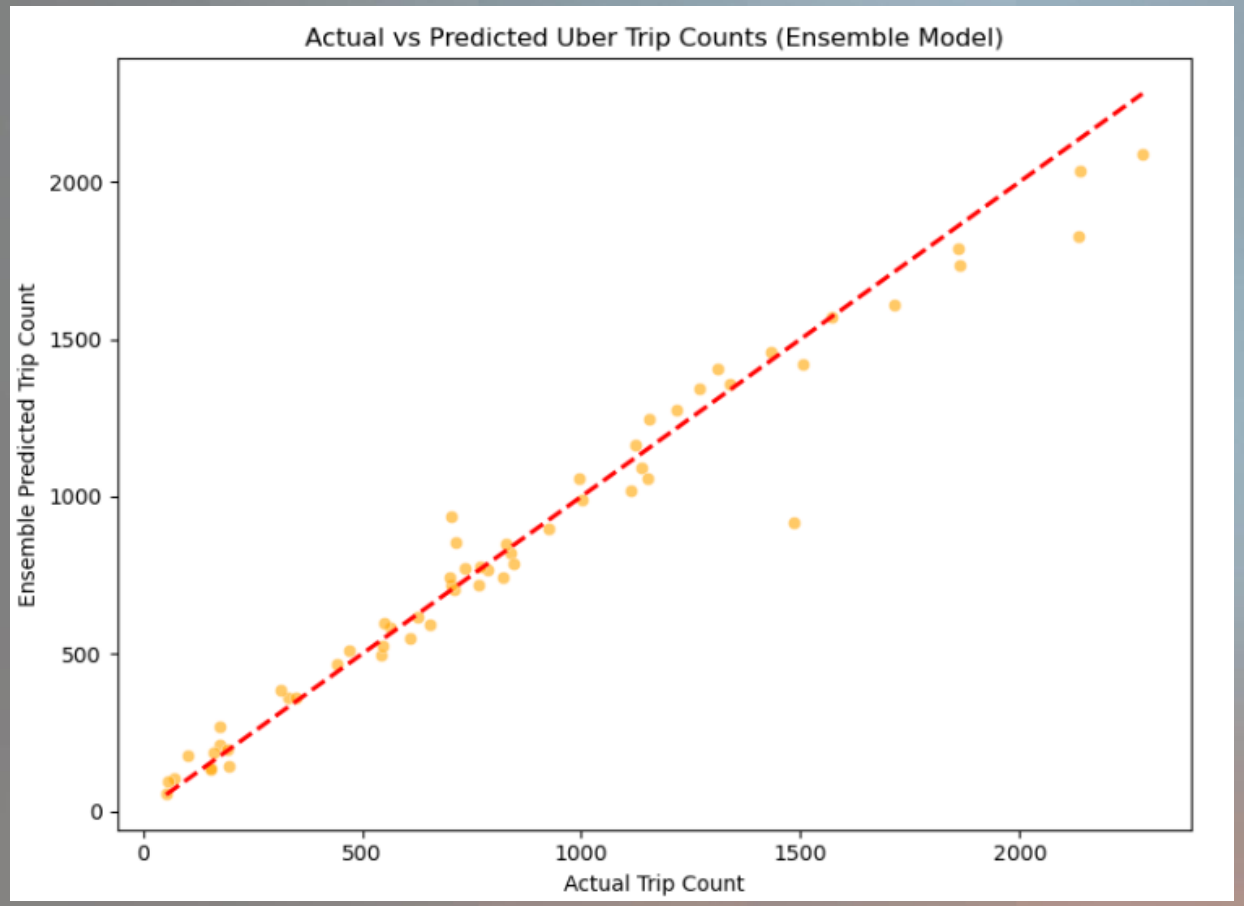
- Tuned GBR results:
- MSE: **9001.95**
- R^2: **0.9717**

Model Performance Comparison

Title: "Model Comparison – MSE & R^2"

Model	MSE	R^2 Score
Random Forest	17933.23	0.9436
XGBoost	11272.17	0.9646
Gradient Boosting	17785.37	0.9441
Tuned GBR	**9001.95**	**0.9717**
Ensemble (Avg)	11952.29	0.9624

Ensemble Model Evaluation



- The ensemble model achieved an **R^2 of 0.9624** and **MSE of 11,952.29**, demonstrating high accuracy in predicting Uber trip counts.
- Scatter plot shows a strong alignment between actual and predicted trip counts, indicating high model accuracy.

Final Takeaways & Conclusion

In this project, we successfully analyzed Uber trip data from NYC for the year 2014, uncovering key temporal and spatial patterns in ride demand. Through effective preprocessing, feature engineering, and model building using Random Forest, XGBoost, and Gradient Boosting, we achieved robust prediction accuracy.

Among individual models, **XGBoost delivered the best performance**, while our **ensemble approach** further enhanced accuracy with an **R^2 of 0.9624** and **MSE of 11,952**, demonstrating strong predictive capabilities.

- The insights derived—such as peak demand hours, weekday trends, and hotspot locations—can be invaluable for optimizing ride-hailing operations in urban settings.

Future Scope & Improvements

1. **Model Enhancements:**
    1. Incorporate **deep learning models** (like LSTM or GRU) to capture complex temporal dependencies.
    2. Explore **automated hyperparameter tuning** using libraries like Optuna or Hyperopt for better efficiency.
2. **Additional Features:**
    1. Integrate **external datasets** such as weather, traffic, or event data to enhance prediction accuracy.
    2. Add **geospatial features** like proximity to landmarks or transit hubs for improved spatial modeling.
3. **Real-Time Deployment:**
    1. Convert the model into a **real-time prediction API** to assist in live demand forecasting.
    2. Use dashboard tools (like Power BI or Streamlit) for **interactive trip demand visualization**.
4. **Business Application:**
    1. Extend the model to assist **driver allocation strategies**, **surge pricing mechanisms**, and **fleet management** decisions.

Thank You

Presented by : Manju Suthar
Intern at : Unified mentor,pvt.ltd