Food Delivery Demand Forecasting

This project predicts hourly food delivery demand for a food delivery service using historical data. It includes data preprocessing, feature extraction, training a linear regression model, and evaluating the model's performance.

1. Import Libraries

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_squared_error, r2_score

from datetime import datetime
```

2. Load Dataset

```
df = pd.read_csv('food_delivery_data.csv')
```

3. Preprocess Data

```
df['Date'] = pd.to_datetime(df['Date'], dayfirst=True)

df['Day'] = df['Date'].dt.day

df['Month'] = df['Date'].dt.month

df['Weekday'] = df['Date'].dt.weekday
```

4. Define Features and Target

```
X = df[['Hour', 'Day', 'Month', 'Weekday']]

y = df['Delivery_Requests']
```

5. Split Data

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

6. Train Model

```
model = LinearRegression()
model.fit(X_train, y_train)
```

7. Evaluate Model

```
y_pred = model.predict(X_test)
print("R² Score:", r2_score(y_test, y_pred))
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
```

8. Plot Results

```
plt.figure(figsize=(10, 5))
plt.plot(y_test.values[:50], label='Actual', marker='o')
plt.plot(y_pred[:50], label='Predicted', marker='x')
plt.title("Actual vs Predicted Delivery Requests")
plt.xlabel("Sample Index")
plt.ylabel("Delivery Requests")
plt.legend()
plt.grid(True)
plt.show()
```

9. Predict Future

```
future_data = pd.DataFrame({'Hour': [13], 'Day': [18], 'Month': [6], 'Weekday': [2]})
future_prediction = model.predict(future_data)
print("Predicted delivery requests for 18 June 2025, 1 PM:", int(future_prediction[0]))
```