F1 Regression Analysis Code Explanation

Overview

This code analyzes Formula 1 race data from 2024 to understand the relationship between **race position** and **lap time** using two different regression approaches: Linear and Polynomial regression.

Data Setup

```python
from .Lab4 import preprocessing
merged_df = preprocessing()
df_2024 = merged_df[merged_df['Race_Year'] == 2024].copy()
df_2024 = df_2024.dropna(subset=['Time(s)'])
```

- Imports preprocessed F1 data from a Lab4 module
- Filters to only include 2024 race data
- Removes any rows with missing lap times to ensure clean analysis

Variables Definition

- **X (Independent Variable)**: `Position` - The finishing position in the race (1st, 2nd, 3rd, etc.)
- **Y (Dependent Variable)**: `Time(s)` - The lap time in seconds

Research Question: Can we predict lap times based on race position?

Model 1: Linear Regression

```python
lin_reg = LinearRegression()
lin_reg.fit(X, y)
y_pred_linear = lin_reg.predict(X)
```

- Fits a straight line through the data points
- Assumes a constant linear relationship between position and lap time
- Simple but may miss non-linear patterns

Model 2: Polynomial Regression (Degree 5)

```python
```

```
poly = PolynomialFeatures(degree=5)
X_poly = poly.fit_transform(X)
poly_reg = LinearRegression()
poly_reg.fit(X_poly, y)
```

- Creates polynomial features (Position, Position^2, Position^3, Position^4, Position^5)
- Fits a curved line that can capture more complex relationships
- More flexible but risk of overfitting with higher degrees

Visualization

The code creates two side-by-side plots:

Left Plot: Linear Regression

- Red dots: Actual data points (Position vs Time)
- Blue line: Linear prediction line

Right Plot: Polynomial Regression

- Red dots: Actual data points
- Green curve: Polynomial prediction curve (smooth, curved line)

Model Evaluation

```
python

print("Linear Regression R²:", r2_score(y, y_pred_linear))
print("Polynomial Regression R²:", r2_score(y, poly_reg.predict(X_poly)))
```

R^2 Score Interpretation:

- Values range from 0 to 1 (higher is better)
- 0.7+ = Good fit
- 0.8+ = Very good fit
- 0.9+ = Excellent fit

Expected Results & Business Insights

Typical F1 Pattern:

- Drivers in better positions (1st, 2nd, 3rd) usually have faster lap times
- The relationship might not be perfectly linear due to:
  - Strategic pit stops

- Tire degradation
- Traffic effects
- Weather conditions

Model Comparison:

- Linear model provides simple, interpretable baseline
- Polynomial model can capture curves but may overfit
- Compare R^2 values to determine which model better explains the data

Key Takeaways for the Team

1. **Purpose**: Understanding position-time relationships in F1 races
2. **Method**: Comparing simple vs complex regression models
3. **Output**: Visual comparison + quantitative performance metrics
4. **Decision**: Use R^2 scores to determine which model better fits our 2024 F1 data

This analysis helps identify whether race position is a good predictor of lap times and which modeling approach works best for our F1 dataset.