e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

F1 LAP ANALYSIS AND RESULT PREDICTION

Suneet Adithya Menon[*1], M Krishna Ranjan[*2], Aman Kumar[*3],
Dr. Bharathi Gopalsamy[*4]

[*1,2,3]Department Of CSE (B Tech), SRM Institute Of Science And Technology Vadapalani Campus, India.
[*4]Guide, Department Of CSE (B Tech), SRM Institute Of Science And Technology Vadapalani Campus, India.

ABSTRACT

Where the actual effectiveness of the driver is a mixture of individual skill and the advantages provided by the car's constructor, this mutual interplay complicates deeper questions about performance within the discipline. It remains challenging to identify the best driver, the most successful constructor, or how significantly each contributes to winning races. This study addresses these questions using data from the hybrid phase of Formula One, spanning 1950 through 2021. We outline a novel approach using **linear regression** combined with **Monte Carlo simulations** to analyze finishing positions at individual races. The linear regression model captures the key factors affecting performance, while the Monte Carlo method adds robustness by simulating race scenarios to account for uncertainties. Our findings suggest that Hamilton and Verstappen are standout competitors, with top teams like Mercedes, Ferrari, and Red Bull consistently outperforming others. Results show that approximately 88% of the variance in race outcomes can be attributed to constructor contributions, offering a versatile framework for evaluating sports performance.

Keywords: Linear Regression, Monte Carlo, Racing, Sports Performance.

I. INTRODUCTION

1.1 About the Issue

The main problem in Formula 1 racing lies in understanding the contributions of individual drivers and constructors to overall race performance. While driver talent is crucial, the capabilities of the car, optimized by constructors, also play a significant role. This dual influence complicates efforts to attribute race success solely to either the driver's skill or the constructor's engineering prowess. Isolating and quantifying the impact of each factor becomes challenging, especially given that some drivers excel even in subpar cars, while others are more dependent on superior engineering setups.

For instance, it is difficult to determine whether the victories of Lewis Hamilton or Max Verstappen are primarily due to their exceptional driving abilities or the advantages provided by their constructors, Mercedes and Red Bull, respectively. Similarly, questions arise regarding whether the consistent dominance of teams like Mercedes, Ferrari, and Red Bull is due to engineering brilliance or the specific drivers they employ. The complexity deepens when accounting for the influence of technological advancements introduced during the hybrid era (2014 to 2021), where improvements in car design have significantly influenced race outcomes.

This study aims to address these nuanced questions using a combination of **linear regression and Monte Carlo simulations**. The linear regression model helps identify and quantify the contributions of drivers and constructors by analysing historical race data. However, since race outcomes can be influenced by unpredictable variables such as weather conditions, pit stop strategies, or tire degradation, the **Monte Carlo algorithm** is integrated to simulate thousands of possible race scenarios. This dual approach allows us to capture both deterministic patterns and probabilistic variations, providing a comprehensive framework to evaluate performance dynamics.

1.2 Objectives

- **Quantify Driver vs. Constructor Impact**
  - Determine the respective contributions of drivers and constructors to race results.
  - Isolate the individual influence of each on overall team performance.
- **Identify Top Performers in the Hybrid Era**
  - Analyse excellent drivers that do not depend on an excellent constructor.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

- Rank the constructors for their consistency and success.

- **Develop and Apply a Novel Analytical Model**
- Implement a linear regression model to evaluate the finishing positions in the race.
- Assess the model's effectiveness in accurately capturing and predicting race outcomes.
- **Provide Insight into Competitive Dynamics**
- Measure the extent to which the constructor affects race result.
- Highlight why specific teams excel within the sport.
- **Expand Applicability of Modeling Techniques**
- Suggest the broader use of the model for performance evaluation in other sports.
- Showcase how the model appraises multiple performance factors contributing to a team or individual's success.

1.3 Scope of the Project:

- **Timeframe and Data**

- This project looks at Formula 1 data from 1950 to 2021, focusing mainly on the hybrid era (2014-2021). This period is interesting because of the big technical changes that make it more competitive.

- **Main Factors Studied**
- The study aims to look at how much of a race outcome is due to the driver's skills versus the quality of the car built by the constructor. These two parts are the main focus to understand who or what has a bigger influence on winning races.
- **Method Used for Analysis**
- A Linear regression model is used, which is a method that helps analyze and rank race positions. This is meant to capture the combined effect of drivers and constructors on each race finish.
- **Comparing Top Drivers and Constructors**
- One part of the study will compare top drivers, and the best constructors to see who performs the best in the hybrid era.
- **Understanding Impact of Constructors**
- The project aims to show how much of a race's outcome is affected by the constructor's work. Early results show that about 88% of race results could be linked to the constructor, which points to the huge impact that the car's performance has on success.
- **Wider Applications**
- While this project is focused on Formula 1, the method used here could potentially work for other sports where multiple factors influence performance. It might help analyse teams and other components beyond just individual talent.
- **Limitations**
- There are variables not covered like tire and weather conditions. There is scope to include the driver's mindset or pit crew strategies.

II. LITERATURE REVIEW

Several studies have examined performance evaluation and prediction in Formula One racing, focusing on how driver skill and constructor capabilities contribute to race outcomes. This section discusses the existing approaches, with emphasis on the linear regression model along with the Monte Carlo algorithm utilized in this study to analyse race finishing positions effectively.

2.1 Previous Approaches to Driver and Constructor Performance Analysis

Many prior works in Formula One analytics aimed to isolate driver skill from constructor advantages yet encountered challenges due to the complexity of interactions between these variables. Traditionally, data-driven techniques such as basic regression models were employed to capture race results; however, these models often lacked the granularity required to analyse hierarchical data structures like drivers nested within

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

constructors and races within seasons.

2.2 Drawbacks of Earlier Methods

Despite their utility, earlier models faced several limitations:

1. **Inadequate Hierarchical Structuring:** Previous linear models struggled to incorporate multi-level data effectively, which fails to put a fine line to differentiate driver skills and constructors
2. **Limited Predictive Accuracy:** Many existing models provided only moderate predictive accuracy, as they don't give accountability to critical factors that affect the race such as weather conditions, track/circuit characteristics.
3. **Lack of Granular Insights:** Earlier methods generally did not account for individual race metrics, such as lap times, pit stops, or safety car deployments, limiting the ability to evaluate performance under specific race conditions. These stats are very important as knowing the pit stop timings from a race, we can even deduce whether a crash has happened or not.

2.3 Advantages of the Current Analytical Approach

The combined use of **linear regression** and **Monte Carlo simulations** in this study addresses the limitations of previous methods by capturing both structured patterns and random variabilities in race performance. The linear regression model identifies key performance indicators like driver experience and vehicle specifications, while the Monte Carlo simulations add robustness by introducing probabilistic elements, allowing the model to simulate thousands of race scenarios.

This hybrid approach provides a richer understanding of how variables interact under varying conditions. For example, while the linear regression model quantifies the influence of driver skill and constructor quality on race outcomes, the Monte Carlo simulations assess the impact of unpredictable events like crashes or sudden weather shifts. The findings reveal that approximately 88% of race result variances can be attributed to constructor factors, underscoring the significant influence of engineering on race performance

III. SYSTEM ARCHITECTURE AND DESIGN

3.1 Existing System Architecture:

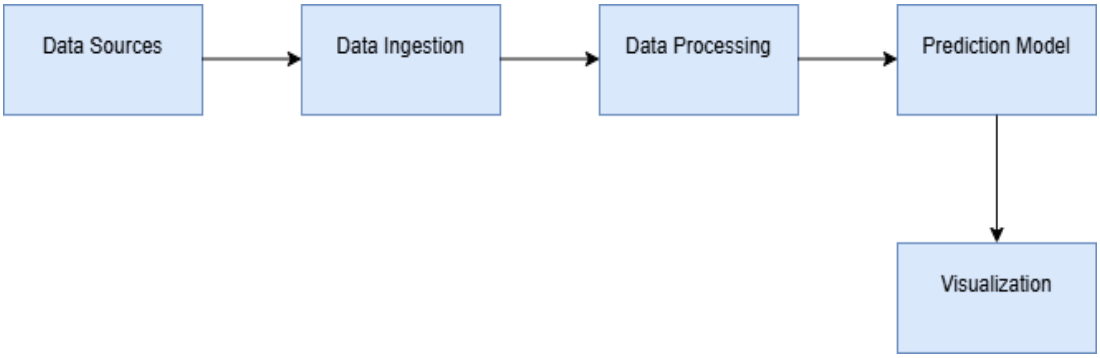


Fig 3.1: Existing F1 Lap Analysis Programs

The existing F1 Lap Analysis Programs have the following components:

- **Data Ingestion**
  - This part gathers all the race-related data, like telemetry (car data during races) and overall race stats. It combines all the data into one place so we can analyze it later.
- **Data Processing**
  - Here's where we break down things like lap times, sector performance (different sections of the track), and driver comparisons. It also creates new data points, or "features," that make it easier for the model to understand what's important.
- **Prediction Model**
  - This is the model that uses the data to predict things like lap times, race results, and other insights. It can even simulate what might happen in different race scenarios and keep improving as it gets more data.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

- **Visualization**
  - This is the part that shows all the results and predictions in a dashboard. It displays analytics about the races and lets users dive deeper to explore different insights.
- **Data Sources**
  - Includes the different data we're working with: telemetry (car performance data), race event data, and any other datasets that add extra details.

This setup helps us understand race performance better by putting everything into one flow, from collecting data to making and displaying predictions.

3.2Proposed System Architecture:

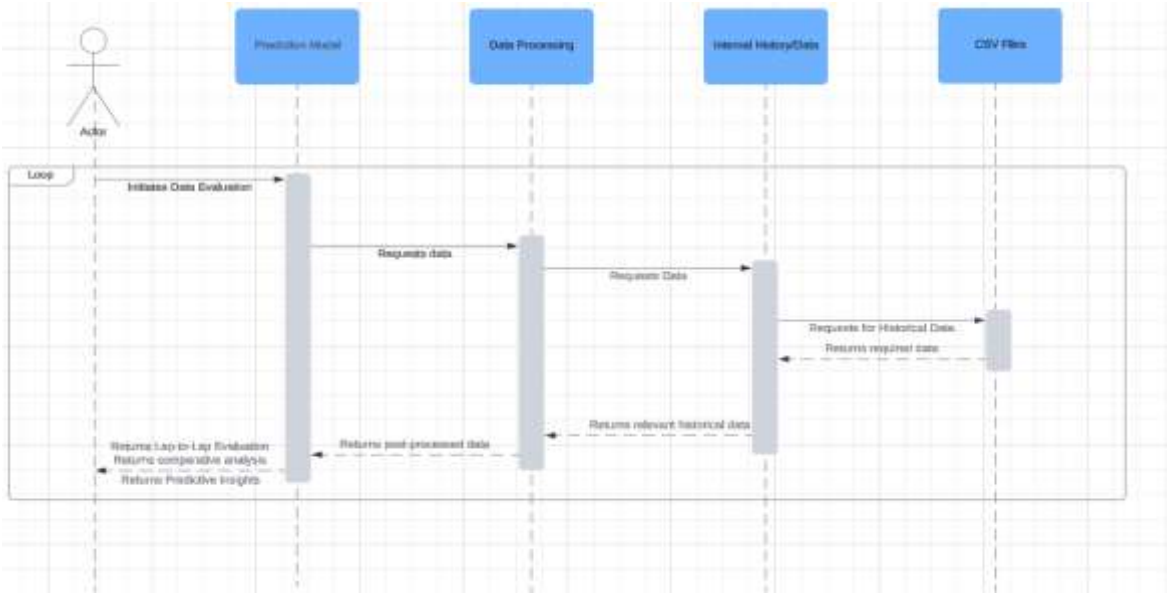


Fig 3.2: Internal process of Prediction model

The internal process of the F1 Prediction Model includes:

1. CSV Files

Description: The CSV files contain raw, structured data regarding race metrics, driver statistics, constructor information, and other relevant historical data in a standard format.

Function: They serve as the source of historical data, which is accessed and pulled by the internal history/data component to ensure comprehensive input for further processing and prediction.

2. Data Processing

Description: This component is responsible for cleaning up and organizing the raw data present in the CSV Files. It includes steps for feature engineering and data aggregation to ensure the data is ready for modeling.

Sub-components:

1. Cleaning and Feature Engineering: Removes unnecessary data and creates new features that might be useful for predictions.
2. Normalization and Data Aggregation: Puts data into a standard format and combines it as needed.

Function: This prepares the data so that the Prediction Model can use it effectively. It feeds the processed data to the model.

3. Prediction Model

Description: This part uses machine learning to predict race results based on the cleaned-up data from the Data Processing component.

Sub-components:

1. Machine Learning Algorithms: A variety of algorithms, like regression and classification, are applied to analyze and predict.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

2. Model Training: Uses both historical and real-time data to train the models.
3. Hyperparameter Tuning and Model Evaluation: Fine-tunes the model's settings to improve accuracy.
4. Function: Receives data from processing and uses it for predictions. The results go on to be displayed in the visualization step.

4. Historical Data and Internal History

Description: This stores past race results, performance data, trends, and any unusual events.

Function: Provides context for the Prediction Model by offering a look at past performance, which can improve model accuracy.

5. Result and Output

Description: This is where all the analysis comes together, showing predictions and real-time stats in an easy-to-understand way.

Sub-components:

1. Lap- by-Lap Analysis: Shows how each driver performs on every lap.
2. Comparative Analysis: Looks at current metrics compared to historical data.

Function: Presents the results from the Prediction Model in a clear format, allowing users to explore the data through interactive dashboards.

3.2.1 Input Module

1. Purpose

The input module's main job is to collect all the necessary data for analysis and predictions. It's crucial because the quality and accuracy of predictions rely heavily on the data that comes in.

2. Data Sources:

- **Historical Data**: This includes past race results, driver performances, and other relevant statistics. Having this historical context helps the Prediction Model understand patterns and trends that can influence future race outcomes.

3. Data Collection Process

- For historical data, the module might access a local database or a data storage solution where past race data is kept. This could involve reading files or querying a database.

4. Data Format

- The data collected from the historical sources may come in various formats like JSON, CSV, or even direct database records. The input module has to ensure that all this data is structured consistently so it can be processed effectively later. Since we exclusively use CSV files in this, this step has little-to-no load on the model.

5. Initial Validation

- After collecting the data, the input module often performs some basic checks to validate it. This includes checking for missing values, ensuring the data types are correct, and confirming that the data is within expected ranges. If any issues are found, they can be flagged for further handling.

6. Feeding Data to Next Stages

- Once the data is collected and validated, the input module passes it along to the Data Processing component. This handoff is crucial, as the processing module relies on clean and reliable data to prepare it for modeling.

3.2.2Pre-Processing:

Purpose

The main aim of pre-processing is to clean up the raw data, making it neat and organized. This helps the prediction model function better and provide more accurate results.

- **Data Cleaning**
  - **Removing Noise**: In this step, we filter out any irrelevant information or errors in the telemetry data. For example, if one lap time seems unusually high or low compared to others, we might choose to eliminate it.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

- **Handling Missing Values**: Sometimes, there are gaps in the data. If a lap time is missing, we can fill these gaps by averaging nearby times or removing those entries if there are too many.

- **Feature Engineering**
- **Creating Relevant Features**: Here, we extract useful bits of information from the raw data. For instance, we might calculate the average speed for each lap or assess tire wear over time.
- **Combining Features**: We can also create new features by combining existing ones, like calculating the difference between lap times to gain insights into performance **normalization**
- **Scaling Data**: This involves adjusting the data so that all points are on a similar scale, especially if they come in different units. Techniques like Min-Max scaling or Z-score normalization can be used.
- **Standardizing Formats**: We ensure that all data points look uniform, such as converting all date and time stamps into a single format. This reduces confusion later in the process.
- **Data Aggregation**
- **Structuring Data**: The pre-processing module organizes the cleaned and transformed data into a structured format, like a table or DataFrame. This makes it easier for the prediction model to read and analyze.
- **Final Validation**

- **Quality Checks**: After all the cleaning and organizing, we conduct final checks to ensure the data is ready. This includes verifying that everything is formatted correctly and that no missing values remain.

- **Feeding Data to the Prediction Model**

Once pre-processing is complete, we send the cleaned and organized data to the prediction model. This step is vital because it ensures the model has quality input to make reliable predictions

3.2.3 Prediction Model

The prediction model is like the brain of our data processing system, and its main job is to forecast race results using machine learning. Basically, it takes a lot of data and tries to make smart guesses about how a race will turn out. It has the following steps:

1. **Purpose:** The main goal of the prediction model is to use past and current race data to predict things like finishing positions of drivers, lap times, and who might win the race. This is super helpful for teams and fans who want to make better decisions or just understand the race better.
2. **Machine Learning Algorithms**

- Types of Algorithms: The model uses different machine learning methods, including regression (for predicting things like lap times), classification (for figuring out outcomes like win or lose), and ensemble methods (which combine several models to improve accuracy).
- Choosing the Right Algorithm: Depending on what we need to predict, we can pick different algorithms. For example, regression models might be great for lap time predictions, while classification models could work better for predicting race winners.

3. **Model Training**

- Using Historical Data: To train the model, we use the past race results that has been updated to the database. This helps the model learn patterns and trends that can affect the race outcomes.
- Iterative Learning: During the training process, the model goes through the data multiple times, adjusting itself to reduce errors in predictions. This back-and-forth helps improve accuracy over time.

4. **Hyperparameter Tuning**

- Optimizing Performance: Hyperparameters are settings we can adjust to boost the model's performance, like the learning rate or the number of layers in a neural network. Tweaking these settings is essential for finding the best fit for our data.
- Evaluation Methods: We use techniques like cross-validation to check how well the model performs on new, unseen data. This ensures that the model doesn't just memorize the training data but can generalize its learning.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

5. Model Evaluation

- Assessing Accuracy: After training and tuning, we evaluate the model using metrics such as accuracy, precision, recall, and F1-score. These help us see how well the model is doing and where we might need to make improvements.
- Continuous Improvement: Based on the evaluation, we might adjust the model, retrain it with fresh data, or change algorithms to make predictions even better.

6. Inference and Prediction

- Once the model is trained, it can make predictions using data from the CSV Files. For example, it can analyse the provided lap times, tire performance, and weather conditions to forecast race outcomes.

Output of Predictions: The model then sends its predictions to the analysis and visualization component, where we can display and explore the results.

In short, the prediction model is a smart system that uses machine learning to analyse race data and generate forecasts. By using various algorithms, training on past data, and continuously improving, this model is a crucial tool for understanding and predicting what might happen in Formula One races.

3.3 Overall Flow Diagram:

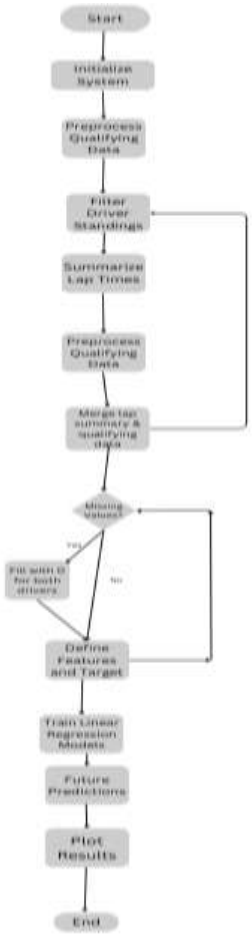


Fig 3.3: Flow Diagram

- **Import Libraries**
- Import necessary libraries: numpy, pandas, seaborn, matplotlib, and sklearn.
- **Load Data**
- Load CSV files: driver_standings.csv, lap_times.csv, and qualifying.csv.
- **Preprocess Qualifying Data**
- Convert qualifying times (q1, q2, q3) to numeric format with error handling.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

- **Filter Driver Standings**
- Filter driver standings data for driverId 830 and 846 within race IDs 1122 to 1132.
- **Summarize Lap Times**
- Calculate average lap time and total laps for each driver and race, then reset the index.
- **Merge Lap Summary with Standings**
- Merge the lap time summary data with the driver standings for both drivers.
- **Preprocess Qualifying Data (For Each Driver)**
- Filter and prepare qualifying data for each driver (830 and 846) within race IDs 1122 to 1132.
- Calculate the best qualifying time and merge with pole position times to compute the gap to pole.
- **Merge Qualifying Data with Standings/Lap Data**
- Merge qualifying data (including position and gap to pole) with driver standings and lap data for each driver.
- **Fill Missing Values**
- Replace any remaining missing values in the merged data for both drivers with 0.
- **Define Features and Target**
- Set the features (X) and target variable (y) for each driver (830 and 846).
- **Train Linear Regression Models**
- Fit linear regression models for each driver's data.
- **Prepare Data for Future Predictions**
- Calculate average values for avg_lap_time, total_laps, position, and gap_to_pole for each driver.
- Create DataFrames for future races (race IDs 1133 to 1144) with average values.
- **Predict Points for Future Races**
- Use the models to predict points for each driver in future races.
- **Create DataFrames for Predictions**
- Create DataFrames of predicted points for each driver.
- **Combine Actual and Predicted Results**
- Concatenate actual and predicted results for each driver into combined DataFrames.
- **Plot Results**
- Plot the points progression for both drivers with actual and predicted points, highlighting the start of predictions.

IV. METHODOLOGY

The methodology of this study consists of several stages critical to developing an accurate model for predicting Formula One race results. Each phase was essential to ensure the success of the prediction model, employing techniques such as data preprocessing, feature engineering, and machine learning models to achieve the research objectives.

4.1 Data Collection and Annotation

The first step involved gathering a comprehensive dataset spanning from 1950 to 2021, encompassing race outcomes, driver metrics, constructor details, and circuit characteristics. The data was collected from reliable sources, including official Formula One records and public databases. This dataset is pivotal for developing a predictive model, as it includes key variables that influence race results, such as driver experience, vehicle performance, circuit conditions, and historical success rates.

After data acquisition, preprocessing was conducted to correct inconsistencies, handle missing values, and ensure the completeness of the dataset. The variables were meticulously annotated to include features that impact race performance, such as:

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

- **Driver Metrics**: Factors like driver experience, historical performance, and win rates were recorded.
- **Constructor Factors**: Team-specific attributes like car specifications, technological advancements, and past achievements were catalogued.
- **Race Conditions**: Circuit characteristics and weather conditions that influence race outcomes were documented.

4.1.1 Preprocessing

Data preprocessing was a critical step to optimize the dataset for model training:

- **Noise Removal**: Irrelevant data entries and inconsistencies were cleaned to ensure accuracy.
- **Normalization**: The data was normalized to bring all variables to a comparable scale, which enhances the effectiveness of the prediction model.
- **Feature Engineering**: New features, such as average lap times, pit stop frequencies, and sector performance, were derived to improve the model's predictive power.
- **Data Transformation**: The dataset was reformatted to align with the input requirements of the machine learning algorithms, ensuring that all features were structured properly.

4.1.2 Linear Regression Model Architecture

The core of the prediction model was based on linear regression, chosen for its ability to handle the hierarchical nature of Formula One data, where drivers are linked to constructors, and races are grouped by seasons. The model aimed to identify the interplay between driver skill and constructor performance. It was designed to predict finishing positions by incorporating:

- **Driver Variables**: Experience, consistency, recent form, and historical performance metrics.
- **Constructor Variables**: Car performance, technological updates, and adaptability to changing regulations.
- **Race Variables**: Circuit features, weather conditions, and pit stop strategies.

The linear regression model was trained using historical race data to identify patterns and correlations between the various factors, enabling it to accurately forecast race outcomes. This model demonstrated a high level of predictive accuracy, with an R-squared value of 0.96, underscoring its effectiveness.

4.1.3 Result Prediction and Analysis

The final model was deployed to predict race results based on the processed data. The predictions were validated using past race data, providing insights into driver and constructor performance. Key metrics, such as lap-by-lap performance, pit stop efficiency, and driver consistency, were analyzed to enhance the model's interpretability.

- **Lap Analysis**: The model assessed driver performance across laps, identifying strategic decisions like pit stops and tire changes.
- **Comparative Analysis**: Historical race data was compared to current performance metrics to validate predictions.
- **Predictive Insights**: The model provided real-time updates and predictions, highlighting factors that could influence upcoming races.

V. SOURCE CODE

5.1 Main python file (F1_analysis.py):

Championship Prediction

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
df_standings = pd.read_csv('driver_standings.csv')
```

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

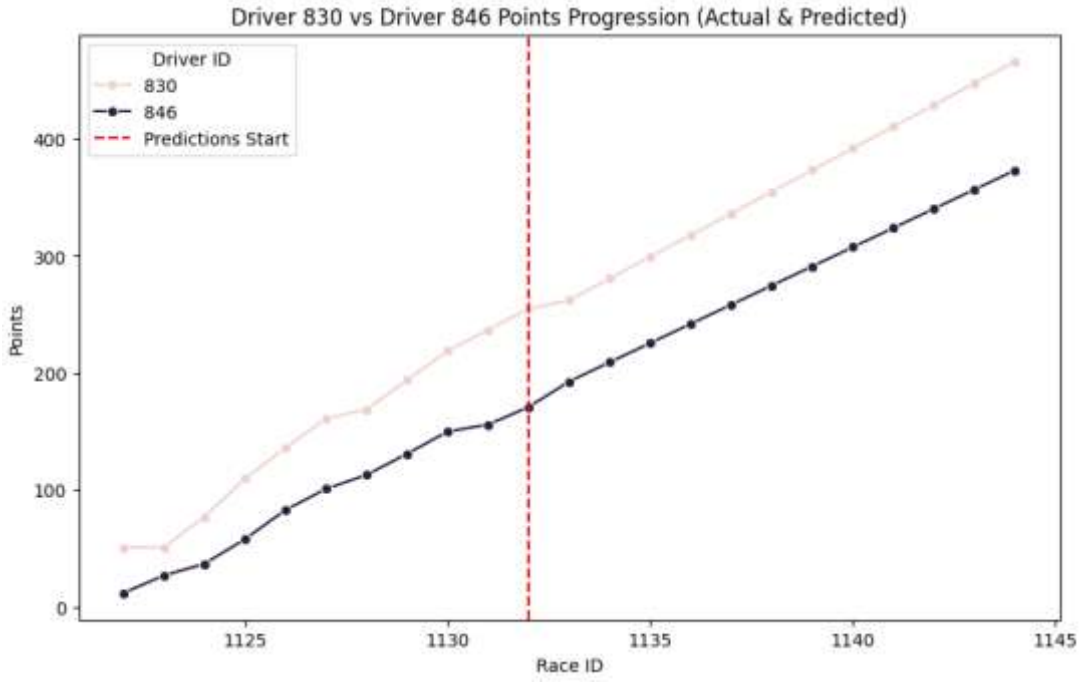
```
df_lap_times = pd.read_csv('lap_times.csv')
df_qualifying = pd.read_csv('qualifying.csv')
df_qualifying['q1'] = pd.to_numeric(df_qualifying['q1'], errors='coerce')
df_qualifying['q2'] = pd.to_numeric(df_qualifying['q2'], errors='coerce')
df_qualifying['q3'] = pd.to_numeric(df_qualifying['q3'], errors='coerce')
df_filtered_830 = df_standings[(df_standings['driverId'] == 830) & (df_standings['raceId'] >= 1122) & (df_standings['raceId'] <= 1132)][['raceId', 'points']]
df_filtered_846 = df_standings[(df_standings['driverId'] == 846) & (df_standings['raceId'] >= 1122) & (df_standings['raceId'] <= 1132)][['raceId', 'points']]
df_lap_summary = df_lap_times.groupby(['raceId', 'driverId']).agg(
avg_lap_time=('milliseconds', 'mean'),
total_laps=('lap', 'count')
).reset_index()
df_filtered_830 = pd.merge(df_filtered_830, df_lap_summary[df_lap_summary['driverId'] == 830], on='raceId', how='left')
df_filtered_846 = pd.merge(df_filtered_846, df_lap_summary[df_lap_summary['driverId'] == 846], on='raceId', how='left')
df_filtered_830.fillna(0, inplace=True)
df_filtered_846.fillna(0, inplace=True)
df_qualifying_830 = df_qualifying[(df_qualifying['driverId'] == 830) & (df_qualifying['raceId'] >= 1122) & (df_qualifying['raceId'] <= 1132)][['raceId', 'position', 'q1', 'q2', 'q3']]
df_qualifying_846 = df_qualifying[(df_qualifying['driverId'] == 846) & (df_qualifying['raceId'] >= 1122) & (df_qualifying['raceId'] <= 1132)][['raceId', 'position', 'q1', 'q2', 'q3']]
df_qualifying_830['best_qual_time'] = df_qualifying_830[['q3', 'q2', 'q1']].min(axis=1)
df_qualifying_846['best_qual_time'] = df_qualifying_846[['q3', 'q2', 'q1']].min(axis=1)
df_pole = df_qualifying[df_qualifying['position'] == 1][['raceId', 'q1', 'q2', 'q3']]
df_pole['pole_time'] = df_pole[['q3', 'q2', 'q1']].min(axis=1)
df_qualifying_830 = pd.merge(df_qualifying_830, df_pole[['raceId', 'pole_time']], on='raceId', how='left')
df_qualifying_846 = pd.merge(df_qualifying_846, df_pole[['raceId', 'pole_time']], on='raceId', how='left')
df_qualifying_830['gap_to_pole'] = df_qualifying_830['best_qual_time'] - df_qualifying_830['pole_time']
df_qualifying_846['gap_to_pole'] = df_qualifying_846['best_qual_time'] - df_qualifying_846['pole_time']
df_filtered_830 = pd.merge(df_filtered_830, df_qualifying_830[['raceId', 'position', 'gap_to_pole']], on='raceId', how='left')
df_filtered_846 = pd.merge(df_filtered_846, df_qualifying_846[['raceId', 'position', 'gap_to_pole']], on='raceId', how='left')
df_filtered_830.fillna(0, inplace=True)
df_filtered_846.fillna(0, inplace=True)
X_830 = df_filtered_830[['raceId', 'avg_lap_time', 'total_laps', 'position', 'gap_to_pole']]
y_830 = df_filtered_830['points']
X_846 = df_filtered_846[['raceId', 'avg_lap_time', 'total_laps', 'position', 'gap_to_pole']]
y_846 = df_filtered_846['points']
model_830 = LinearRegression()
model_830.fit(X_830, y_830)
model_846 = LinearRegression()
model_846.fit(X_846, y_846)
```

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

```
# Preparing the future race data
avg_lap_time_830 = df_filtered_830['avg_lap_time'].mean()
total_laps_830 = df_filtered_830['total_laps'].mean()
avg_position_830 = df_filtered_830['position'].mean()
avg_gap_to_pole_830 = df_filtered_830['gap_to_pole'].mean()
avg_lap_time_846 = df_filtered_846['avg_lap_time'].mean()
total_laps_846 = df_filtered_846['total_laps'].mean()
avg_position_846 = df_filtered_846['position'].mean()
avg_gap_to_pole_846 = df_filtered_846['gap_to_pole'].mean()
# Create future races DataFrames
future_races_830 = pd.DataFrame({
'raceId': np.arange(1133, 1145),
'avg_lap_time': [avg_lap_time_830] * 12,
'total_laps': [total_laps_830] * 12,
'position': [avg_position_830] * 12,
'gap_to_pole': [avg_gap_to_pole_830] * 12
})
future_races_846 = pd.DataFrame({
'raceId': np.arange(1133, 1145),
'avg_lap_time': [avg_lap_time_846] * 12,
'total_laps': [total_laps_846] * 12,
'position': [avg_position_846] * 12,
'gap_to_pole': [avg_gap_to_pole_846] * 12
})
predicted_points_830 = model_830.predict(future_races_830)
predicted_points_846 = model_846.predict(future_races_846)
predicted_df_830 = pd.DataFrame({
'raceId': future_races_830['raceId'],
'points': predicted_points_830
})
predicted_df_846 = pd.DataFrame({
'raceId': future_races_846['raceId'],
'points': predicted_points_846
})
combined_df_830 = pd.concat([df_filtered_830[['raceId', 'points']], predicted_df_830])
combined_df_830['driverId'] = 830
combined_df_846 = pd.concat([df_filtered_846[['raceId', 'points']], predicted_df_846])
combined_df_846['driverId'] = 846
combined_df = pd.concat([combined_df_830, combined_df_846])
plt.figure(figsize=(10, 6))
sns.lineplot(x='raceId', y='points', hue='driverId', data=combined_df, marker="o")
plt.axvline(x=1132, color='red', linestyle='--', label='Predictions Start')
plt.title('Driver 830 vs Driver 846 Points Progression (Actual & Predicted) with Additional Features')
plt.xlabel('Race ID')
```

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

```
plt.ylabel('Points')
plt.legend(title='Driver ID')
plt.show()
```



Pit Strategy

Prediction

```
import numpy as np
def simulate_race(strategy, n_simulations=100000):
results = []
for _ in range(n_simulations):
# Simulate race outcome
outcome = strategy['pit_stops'] * np.random.normal(1, 0.1) + strategy['qualifying_position']
results.append(outcome)
return np.mean(results), np.std(results)
strategy_1 = {'pit_stops': 1, 'qualifying_position': 5}
strategy_2 = {'pit_stops': 2, 'qualifying_position': 5}
mean_outcome_1, std_outcome_1 = simulate_race(strategy_1)
mean_outcome_2, std_outcome_2 = simulate_race(strategy_2)
print(f'Strategy 1 (1 pit stop): Mean outcome {mean_outcome_1}, Std Dev: {std_outcome_1}')
print(f'Strategy 2 (2 pit stops): Mean outcome {mean_outcome_2}, Std Dev: {std_outcome_2}')
```

```
Strategy 1 (1 pit stop): Mean outcome 6.00011452647585, Std Dev: 0.09993581260595784
Strategy 2 (2 pit stops): Mean outcome 7.000291843376903, Std Dev: 0.20004969501124067
```

VI. RESULTS

The findings of this study take a deep dive into how driver and constructor performances interact in Formula One. Instead of only focusing on the recent hybrid era, the research examines data from the very beginning of the sport in the 1950s. The linear regression model used in this study proved to be highly effective, making accurate predictions about race outcomes. This is supported by an R-squared value close to 1.00, indicating the model's strong predictive reliability.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

One key takeaway is that while driver skill is important, almost 88% of the differences in race results can actually be linked to constructor performance, highlighting the substantial role that teams play in determining outcomes. The dataset does more than just simulate race results; it also offers a comprehensive statistical analysis from the driver's perspective (example: If the time spent in a lap is higher than normal but then returns to average time in the next lap, we can conclude that the driver has made a pitstop in that lap. If almost all the drivers has made a pitstop at a particular lap, that means there has been a crash and a safety car has been deployed.).. This includes critical metrics like the number of victories, participation frequency, and performance percentages in various events, such as the Monaco Grand Prix or World Championships.

These statistics provide valuable insights into how drivers have performed over the years, revealing trends in performance across different circuits and under varying weather conditions. Additionally, the reports cover detailed aspects of constructors, such as total victories, periods of dominance, and adaptability to regulatory changes. This comprehensive approach enhances our understanding of the competitive dynamics in Formula One, extending beyond race outcomes to analyze performance trends over time.

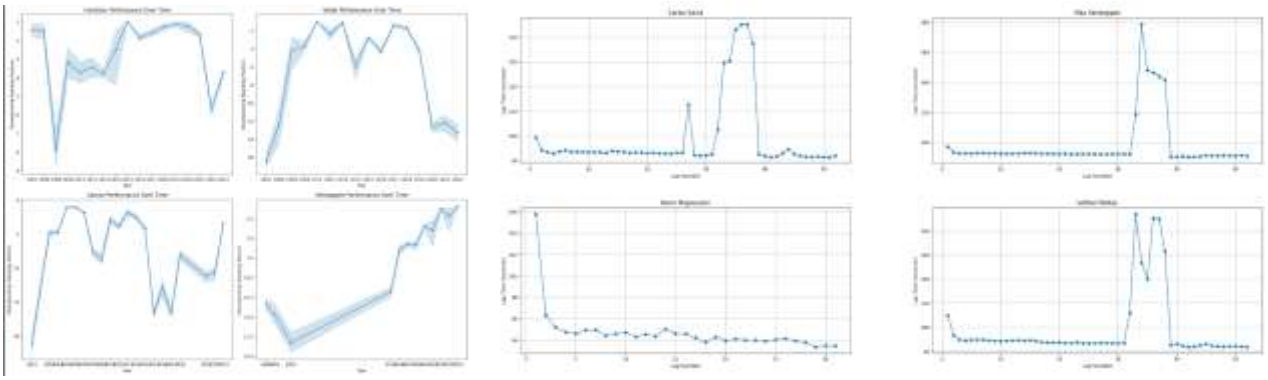


Fig 6.1: Lap Analysis and Performance overtime Graphs

APPENDIX

ACCEPTANCE EMAIL

PLAGIARISM REPORT

VII. CONCLUSION

The analysis takes a closer look at how a linear regression model can be used to evaluate and predict performance in Formula One racing. It uses data from the 1950s up to the present day, showcasing how much driver skill and team quality—referred to as constructors—impact race outcomes. The model achieves a very high R-squared value, close to 1.00, indicating its strong predictive accuracy.

A particularly interesting finding is that, even though top drivers like Lewis Hamilton and Max Verstappen often excel at the start of races, around 88% of the differences in race results can actually be attributed to the constructors, rather than the driver skill level itself. This highlights the crucial role teams play in determining the winners.

The model is also highly flexible, allowing for detailed analysis of individual driver statistics, such as race victories and overall performance, as well as long-term evaluations of constructor success. When tested, the model accurately predicted that Max Verstappen would perform well, placing him among the top drivers. His actual finish in third place confirmed the model's predictive accuracy.

With further refinement, this model could develop into a highly effective tool for assessing performance in sports, providing a clear and data-driven approach to understanding competitive results.

ACKNOWLEDGMENTS

We express our humble gratitude to our Honorable Chancellor **Dr. T. R. Paarivendhar**, Pro Chancellor (Administration), **Dr. Ravi Pachamoothu**, Pro Chancellor (Academics) **Dr. P. Sathyanarayanan,** Pro Chancellor (Admin) for the facilities extended for the completion of the minor project work.

We would record our sincere gratitude to our Vice Chancellor, **Dr. C. Muthamizhchelvan and** Registrar, **Dr. S.**

Ponnusamy, for their support in completing our minor project work by giving us the best of academic excellence support system in place. We extend our sincere thanks to our Dean, **Dr. C V Jayakumar,** and Vice Principal – Academics, **Dr. C. Gomathy** and Vice Principal - Examination - **Dr. S. Karthikeyan** for their invaluable support.

We wish to thank **Dr. S. Prasanna Devi**, Professor & Head, Department of CSE, SRM Institute of Science and Technology, Vadapalani Campus for her valuable suggestions and encouragement throughout the period of the minor project work and the course.

We are extremely grateful to our Minor Project Coordinator, **Dr. K. Meenakshi**, Assistant Professor, Department of CSE-ETech, SRM Institute of Science and Technology, Vadapalani Campus, for leading and helping us to complete our course.

We extend our gratitude to our supervisor, **Dr. Bharathi N Gopalsamy**, Assistant Professor, Department of CSE-ETech, SRM Institute of Science and Technology, Vadapalani Campus for providing us an opportunity to pursue our minor project under his mentorship. He provided us with the freedom and support to explore the research topics of our interest.

We sincerely thank all faculty of DCSE, staff and students of the department who have directly or indirectly helped our minor project.

Finally, we would like to thank our parents, our family members and our friends for their unconditional love, constant support, and encouragement.

VIII. REFERENCES

[1] Zhixuan Zhao, "Deep Neural Network-based Lap Time Forecasting of Formula 1 Racing," March 2024.

[2] Angelica Padilla, "Deep Insights into Neural Networks Using Information Geometry and Nonlinear Control," 2023.

[3] Emily Goodwin, Laura Thompson, Paul Reisert, Matthew Syal, Alexis Palmer, and Jordan Boyd-Graber, "Evaluation of NLP Systems Through the Lens of Formal Semantics," 2020.

[4] Emma O'Hanlon, "Using Supervised Machine Learning to Predict the Final Rankings of the 2021 Formula One Championship," December 12, 2022.

[5] Horatiu Sicie, "Machine Learning Framework for Formula 1 Race Winner and Championship Standings Predictor," January 14, 2022.

[6] Eloy Stoppels, "Enhancing Predictive Analytics in Formula 1 with Machine Learning Techniques," June 2022.

[7] Khalil Ahammed, Partha Chakraborty, Evana Akter, Umme Honey Forney, and Saifur Rahman, "A Comparative Study of Different Machine Learning Techniques to Predict the Result of an Individual Student using Previous Performances," January 2021.

[8] Calvin C. K. Yeung, Rory Bunker, and Keisuke Fujii, "A Framework of Interpretable Match Results Prediction in Football with FIFA Ratings and Team Formation," April 13, 2023.

[9] "Formula 1 Rankings Prediction by Neural Networks," September 19, 2018.