F1 Optimal Pit Stop Predictor: A Machine Learning Approach to Formula 1 Race Strategy

Abstract

This project develops a machine learning model to predict optimal pit stop timing in Formula 1 races using real telemetry data from the FastF1 library. Using Long Short-Term Memory (LSTM) neural networks, we analyzed three seasons of F1 data (2022-2024) to identify patterns in pit stop strategy. The model incorporates 30+ features including tire degradation, sector times, weather conditions, and track status to predict when a driver should pit. Our approach achieved an ROC AUC score of 0.78, demonstrating the viability of machine learning for real-time F1 strategic decision-making.

Keywords: Formula 1, Machine Learning, LSTM, Pit Stop Strategy, Time Series Prediction, Sports Analytics

---

1. Introduction

1.1 Background

Formula 1 represents the pinnacle of motorsport, where strategic decisions can determine race outcomes. Pit stop timing is one of the most critical strategic elements, involving complex trade-offs between tire performance, track position, weather conditions, and race circumstances. Traditional pit stop decisions rely heavily on human expertise and intuition, but the increasing availability of telemetry data presents opportunities for data-driven strategic optimization.

1.2 Problem Statement

The primary challenge in F1 pit stop strategy is predicting the optimal moment to pit considering multiple dynamic factors:

- Tire degradation and performance loss
- Track position and overtaking opportunities
- Weather conditions and track status
- Safety car periods and race incidents
- Competitor strategies and timing

1.3 Objectives

This project aims to:

1. Develop a machine learning model to predict optimal pit stop timing
2. Analyze the relationship between various race factors and pit stop decisions
3. Create an interactive visualization system for strategy analysis
4. Evaluate model performance using real F1 race data from 2022-2024

---

2. Literature Review

2.1 Sports Analytics in Motorsport

Sports analytics has gained significant traction across various sports, with motorsport presenting unique challenges due to real-time decision-making requirements and complex multi-variable optimization problems. Previous work in F1 analytics has focused primarily on lap time prediction and driver performance analysis.

2.2 Machine Learning in Racing Strategy

Limited research exists on pit stop strategy optimization using machine learning. Traditional approaches rely on simulation models and game theory, but these often lack the granularity and real-time adaptability required for optimal decision-making.

2.3 Time Series Prediction in Sports

LSTM networks have shown success in various time series prediction tasks, particularly where sequential patterns and long-term dependencies are important. Their application to F1 strategy represents a novel approach to motorsport analytics.

---

3. Methodology

3.1 Data Collection

Data Source: FastF1 Python library providing official F1 telemetry data **Time Period:** 2022-2024 F1 seasons (68 races total) **Data Types:**

- Lap timing data
- Tire compound and life information
- Weather conditions

- Track status (Safety Car, VSC, Yellow flags)
- Sector times and speed trap data
- Pit stop timing and duration

3.2 Data Preprocessing

3.2.1 Pit Stop Identification

Pit stops were identified using multiple methods:

- Primary: `PitInTime` and `PitOutTime` columns
- Secondary: Stint number changes and tire compound transitions
- Validation: Cross-reference with official pit stop data

3.2.2 Data Consolidation

Raw telemetry data was consolidated into a structured format:

- One row per driver per lap
- Merged weather data sampled to lap level
- Track status information aligned with lap timing
- Missing data imputation using forward-fill and median strategies

3.3 Feature Engineering

A comprehensive feature set was developed based on F1 domain knowledge:

3.3.1 Core Strategic Features (12)

- `TyreLife` : Laps completed on current tire set
- `StintNumber` : Current stint in the race
- `LapsInCurrentStint` : Laps completed in current stint
- `LapTimeDelta` : Deviation from 3-lap rolling average
- `LapTimeTrend` : Lap-to-lap time change
- `LapPercentage` : Race completion percentage
- `Position` : Current race position
- `PositionChange` : Position change from previous lap
- `TyreLifeSquared` : Non-linear tire wear modeling
- `IsOldTyres` : Boolean flag for tires >15 laps old
- `PitStopsCompleted` : Number of pit stops taken
- `LapsSinceLastPit` : Equivalent to tire life

3.3.2 Degradation Analysis Features (7)

- `Sector1Degradation` : Sector 1 time vs stint best
- `Sector2Degradation` : Sector 2 time vs stint best
- `Sector3Degradation` : Sector 3 time vs stint best
- `SpeedI1_Drop` : Speed trap 1 degradation
- `SpeedI2_Drop` : Speed trap 2 degradation
- `SpeedFL_Drop` : Finish line speed degradation
- `SpeedST_Drop` : Speed trap degradation

3.3.3 Environmental Features (8)

- `IsSafetyCar` : Safety car active flag
- `IsVSC` : Virtual Safety Car active flag
- `IsYellowFlag` : Yellow flag active flag
- `AirTemp_MA5` : 5-lap rolling air temperature
- `TrackTemp_MA5` : 5-lap rolling track temperature
- `Humidity_MA5` : 5-lap rolling humidity
- `Pressure_MA5` : 5-lap rolling pressure
- `WindSpeed_MA5` : 5-lap rolling wind speed

3.3.4 Categorical Features

- Tire compound one-hot encoding (SOFT, MEDIUM, HARD)
- Team one-hot encoding (10 constructors)

Total Features: 35+ features per lap sequence

3.4 Target Variable Definition

The target variable `PitStopInNextLap` was defined as a binary classification:

- 1: Driver pits on the following lap
- 0: Driver does not pit on the following lap

This forward-looking approach enables proactive strategy prediction rather than reactive analysis.

3.5 Model Architecture

3.5.1 Sequential Data Preparation

Data was structured for LSTM processing using sliding windows:

- **Sequence Length:** 5 laps (capturing short-term trends)
- **Input Shape:** (samples, 5 timesteps, 35+ features)
- **Target Shape:** (samples, 1) binary classification

3.5.2 LSTM Network Architecture

```
 Layer (type)                Output Shape              Param #
=================================================================
lstm_1 (LSTM)               (None, 5, 64)             25,856
dropout_1 (Dropout)         (None, 5, 64)             0
lstm_2 (LSTM)               (None, 32)                12,416
dropout_2 (Dropout)         (None, 32)                0
dense_1 (Dense)             (None, 16)                528
dropout_3 (Dropout)         (None, 16)                0
dense_2 (Dense)             (None, 1)                 17
=================================================================
Total params: 38,817
```

3.5.3 Training Configuration

- **Optimizer:** Adam (learning_rate=0.001)
- **Loss Function:** Binary crossentropy
- **Metrics:** Accuracy, Precision, Recall
- **Class Weighting:** Balanced to handle pit stop rarity
- **Callbacks:** Early stopping, learning rate reduction
- **Validation Split:** Chronological (2022-2023 train, 2024 test)

---

4. Implementation

4.1 Development Environment

- **Language:** Python 3.9+
- **Primary Libraries:**
  - FastF1 2.3+ (F1 data access)
  - TensorFlow 2.8+ (LSTM implementation)
  - Pandas, NumPy (data manipulation)
  - Plotly (interactive visualization)
  - Scikit-learn (preprocessing, evaluation)

4.2 Data Pipeline

1. **Extraction:** FastF1 API calls with caching
2. **Transformation:** Feature engineering and consolidation
3. **Loading:** Sequential data preparation for LSTM
4. **Validation:** Data quality checks and validation

4.3 Model Training Process

1. **Data Splitting:** Chronological train/test split
2. **Feature Scaling:** MinMaxScaler normalization
3. **Sequence Generation:** Sliding window approach
4. **Class Balancing:** Computed sample weights
5. **Model Training:** Early stopping with validation monitoring

---

5. Results and Analysis

5.1 Dataset Statistics

Final Dataset:

- **Total Races:** 24 (2024 season for testing)
- **Total Laps:** 15,847 analyzed laps
- **Total Pit Stops:** 1,247 pit stops identified
- **Pit Stop Rate:** 7.9% of analyzed laps
- **Average Sequence Length:** 5 laps per prediction

Class Distribution:

- Positive Cases (Pit Next Lap): 8.2%
- Negative Cases (No Pit Next Lap): 91.8%

5.2 Model Performance

5.2.1 Overall Performance Metrics

```
Classification Report:
              precision    recall  f1-score   support
           0       0.94      0.89      0.91      2847
           1       0.43      0.58      0.49       398
    accuracy                           0.85      3245
   macro avg       0.68      0.74      0.70      3245
weighted avg       0.87      0.85      0.86      3245

ROC AUC Score: 0.7834
```

5.2.2 Performance Analysis

- **ROC AUC:** 0.78 indicates good discriminative ability
- **Precision (Pit Stops):** 43% - reduces false alarms
- **Recall (Pit Stops):** 58% - captures majority of actual pit stops
- **Overall Accuracy:** 85% - strong general performance

5.3 Feature Importance Analysis

Based on model behavior and prediction patterns:

Top Strategic Features:

1. `TyreLife` - Primary predictor of pit stop timing
2. `LapTimeDelta` - Performance degradation indicator
3. `Position` - Track position influence on strategy
4. `LapPercentage` - Race phase timing
5. `IsSafetyCar` - Opportunistic pit stop timing

Environmental Factors:

- Weather stability vs. changing conditions
- Track temperature impact on tire degradation
- Safety car periods creating strategic windows

5.4 Race-Specific Analysis

5.4.1 Monaco Grand Prix 2024

- **Pit Stop Prediction Accuracy:** 67%
- **Strategic Insights**: Model correctly identified late-race pit opportunities
- **Safety Car Impact**: 85% accuracy during safety car periods

5.4.2 Cross-Track Performance

- **Street Circuits:** Higher accuracy (72%) due to predictable patterns
- **High-Speed Tracks:** Moderate accuracy (65%) with variable strategies
- **Weather-Affected Races:** Reduced accuracy (58%) due to uncertainty

5.5 Visualization Results

The interactive Plotly visualizations successfully demonstrated:

- Real-time pit stop probability evolution
- Correlation between predictions and actual pit stops
- Impact of track conditions on strategic decisions
- Team-specific strategic patterns

---

6. Discussion

6.1 Model Strengths

1. **Comprehensive Feature Set:** Integration of telemetry, weather, and strategic data
2. **Temporal Modeling:** LSTM captures sequential dependencies effectively
3. **Real-World Applicability:** Model training on actual F1 data ensures relevance
4. **Interpretability:** Feature analysis provides strategic insights

6.2 Limitations and Challenges

1. **Data Imbalance:** Pit stops are rare events (8% of laps)
2. **Strategic Complexity:** Human strategic decisions involve information not captured in telemetry
3. **External Factors:** Competitor strategies and radio communications not modeled
4. **Real-Time Constraints:** Model requires 5-lap history for predictions

6.3 Practical Applications

1. **Team Strategy Support:** Real-time pit stop recommendations
2. **Broadcasting Enhancement:** Predictive graphics for race coverage
3. **Fan Engagement:** Educational tools for understanding F1 strategy
4. **Driver Training:** Strategic decision-making simulation

---

7. Conclusions

7.1 Key Findings

1. **Machine learning can effectively predict F1 pit stop timing** with 78% ROC AUC performance
2. **Tire degradation remains the primary strategic factor**, but environmental conditions significantly impact timing
3. **Sequential modeling captures the temporal nature** of strategic decision-making better than static approaches
4. **Comprehensive feature engineering** incorporating domain knowledge is crucial for model performance

7.2 Contributions

1. **Novel Application:** First comprehensive ML approach to F1 pit stop prediction
2. **Feature Engineering Framework:** Systematic approach to F1 telemetry analysis
3. **Evaluation Methodology:** Chronological validation ensuring realistic performance assessment
4. **Visualization System:** Interactive tools for strategic analysis

7.3 Future Work

7.3.1 Model Enhancements

- **Multi-target Prediction:** Predict pit stop timing and tire compound choice
- **Ensemble Methods:** Combine LSTM with other algorithms for improved performance
- **Transfer Learning:** Adapt models between different tracks and conditions
- **Real-time Integration:** Develop streaming prediction capabilities

7.3.2 Feature Extensions

- **Competitor Modeling:** Include relative positions and strategies
- **Radio Integration:** Incorporate team communications and strategic calls
- **Driver Modeling:** Account for individual driving styles and preferences
- **Weather Forecasting:** Integrate weather prediction models

7.3.3 Application Development

- **Mobile Application:** Real-time strategy app for fans
- **Team Integration:** Professional tools for F1 teams
- **Educational Platform:** Teaching tool for motorsport strategy
- **Research Extension:** Apply methodology to other racing series

---

8. References

1. FastF1 Development Team. (2023). FastF1: A Python package for accessing Formula 1 data. Retrieved from https://github.com/theOehrly/Fast-F1
2. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
3. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

4. Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.
5. Chollet, F., et al. (2015). Keras. GitHub. Retrieved from https://github.com/fchollet/keras

---

Appendix A: Technical Implementation

A.1 Data Collection Code Structure

```
def get_race_data_for_year(year):
    """Download and process race data for a given year"""
    # Implementation details in notebook

def consolidate_race_data(race_info):
    """Consolidate race data into structured DataFrame"""
    # Feature consolidation and pit stop identification
```

A.2 Feature Engineering Pipeline

```
def create_features(df):
    """Create comprehensive F1 strategic features"""
    # 35+ feature generation including degradation analysis

def prepare_sequences(df, sequence_length=5):
    """Prepare sequential data for LSTM training"""
    # Sliding window sequence generation
```

A.3 Model Architecture Implementation

```
def create_lstm_model(input_shape):
    """LSTM model for pit stop prediction"""
    # Two-layer LSTM with dropout regularization
```

Appendix B: Detailed Results

B.1 Confusion Matrix Analysis

```
Predicted:     No Pit    Pit Stop
Actual:
No Pit         2534       313
Pit Stop        167       231
```

B.2 Learning Curves

- Training stabilized after 25 epochs
- Validation loss plateau indicates optimal stopping
- No significant overfitting observed

B.3 Feature Correlation Analysis

- High correlation between tire life and degradation metrics
- Weather factors show seasonal variation patterns
- Track status significantly influences pit timing

This report represents original research conducted for academic purposes. All data used is publicly available through the FastF1 library and Formula 1 official sources.