Drive To Survive

An Analysis of Vehicle Failure in Formula 1 Auto-Racing

Contributors:
Monica Williams, Georgetown University
James Bifulco, Georgetown University
Bill Corkery, Georgetown University
Thomas Marzol, Georgetown University

Certificate in Data Science, Georgetown University School of Continuing Studies
9/18/2021

AGENDA

Topics	Est Time	Slide Number(s)
Executive Summary	2 min	3
Background & Problem Identification	3 min	4
Hypothesis Generation	2 min	5
Data Summary • Initial Ingestion • Selection of Tools & Storage • Project Architecture • Feature Selection & Engineering	5 min	6-10
Modeling Efforts	5 min	11-12
Results & Conclusion	2 min	13-15
Next Steps & Unanswered Questions	1 min	16
Appendix	--	17-21

EXECUTIVE SUMMARY

Formula One Project Overview

- Formula 1, is an international auto racing league
- Our team's objective was to understand the factors that influenced the probability of a Formula 1 car completing a given race successfully without experience a collision, accident, mechanical failure, or any other race-ending failure
- We built a dataset where one car completing one race resulted in one row of performance data
- Our data included information about the weather, the track's altitude, composition, and popularity, and finally the car's on-track performance
- We tested a variety of binary classifiers against this dataset and concluded that Bagging Classifier, Random Forest, and Extra Trees were the strongest three models for predicting vehicle failures
- We concluded that when given out-of-sample data Bagging Classifier correctly predicted a vehicle failure ~66% of the time

BACKGROUND & PROBLEM IDENTIFICATION

BACKGROUND

Formula 1 international auto-racing

- Formula 1, also known as F1 or the FIA World Championship, is the highest class of international auto-racing for single-seat formula racing cars
- The sport contains 10 teams, known as Constructors, with two Drivers and two cars per team in each race
- Winning races gives drivers and Constructors yearly points. Drivers with the most points win the World Drivers' Championship (WDC) and Constructors with the most points win the World Constructors' Championship (WCC)

PROBLEM IDENTIFICATION

Financial Structure of Formula 1

- Formula 1 does not have a level financial playing field
- Every vehicle failure matters

Results Pay

- Being a Driver or a Constructors in the top 3 of WDC or WCC has traditionally resulted in bonus payments for winning Constructor's next season
- The competition on track translates to a larger competition to acquire, manage, and efficiently use resources to produce the best car

Formula 1 is an expensive, fast-paced racing sport where the cost of crashing leads to short-term and long term complications.

HYPOTHESIS GENERATION

Hypothesis Generation Overview

- Our team’s objective was to understand the factors that influenced the probability of a Formula 1 car completing a given race successfully
- We developed detailed hypotheses of the condition and state of the car
- Unfortunately, the data we would need to validate is closely guarded by the teams
- We shifted to the final hypotheses below

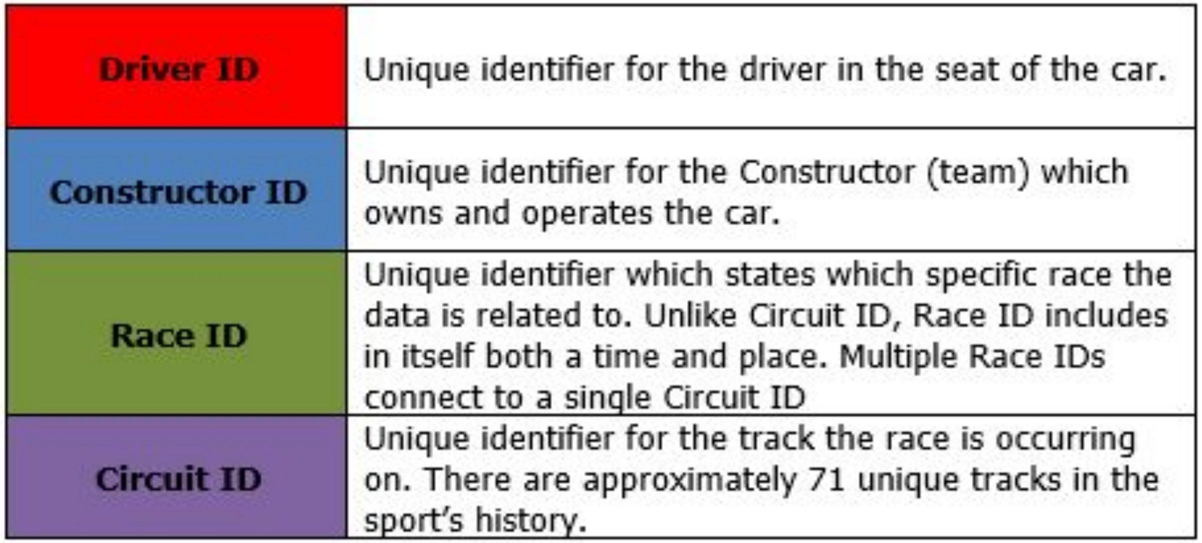
Process	Update
Original Hypotheses	**Original Hypothesis 1:** Predict when a tire needs to be replaced. There are many types of tires so an optimal strategy is to pit right before the tire needs to be replaced **Original Hypothesis 2:** Predict what position a car would finish in a race given starting position, global standing, weather, etc.
Final Hypotheses	**Final Hypothesis 1:** Starting Position (Grid) has a strong direct relationship with the probability of completing the race **Final Hypothesis 2:** Average lap time has an inverse relationship with the probability of completing the race **Final Hypothesis 3:** Precipitation and higher average temperature have an inverse relationship with the probability of completing the race

DATA SUMMARY: INITIAL INGESTION

Initial Ingestion:

- Our data came from two primary places: Kaggle and NOAA Climate Data. We have 4 primary areas of data:
  1. On track performance
  2. Track conditions
  3. Track features
  4. Weather Data
- Our data has 4 major keys which connect our csv's together. Those key ID's are Driver ID, Constructor ID, Race ID, and Circuit ID

Key IDs for Data



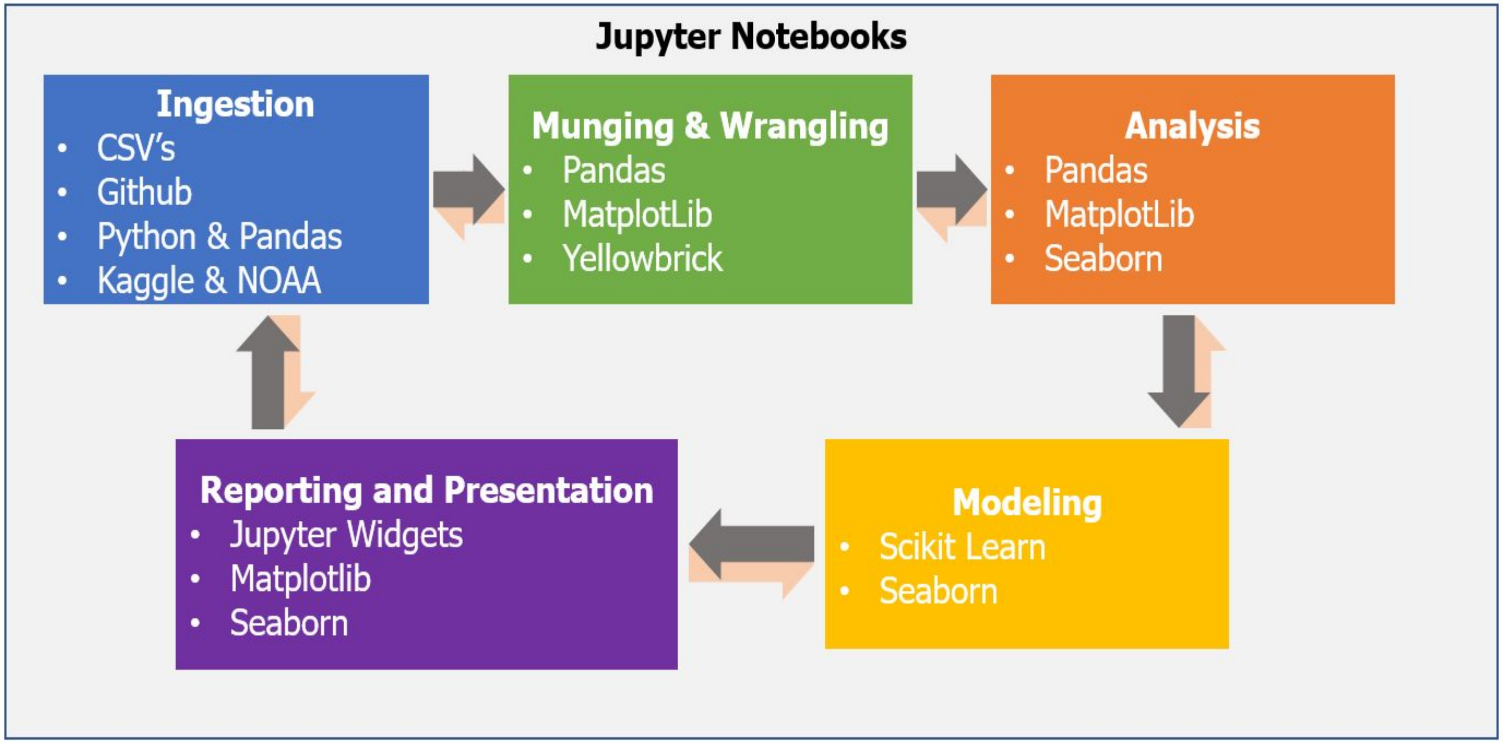
Driver ID	Unique identifier for the driver in the seat of the car.
Constructor ID	Unique identifier for the Constructor (team) which owns and operates the car.
Race ID	Unique identifier which states which specific race the data is related to. Unlike Circuit ID, Race ID includes in itself both a time and place. Multiple Race IDs connect to a single Circuit ID
Circuit ID	Unique identifier for the track the race is occurring on. There are approximately 71 unique tracks in the sport's history.

DATA SUMMARY:
PROJECT ARCHITECTURE

Project Architecture & Tools:

- Our team initially chose to use Pandas, Matplotlib, Plotly, Seaborn, Scikit-Learn, and Yellowbrick as core tools
- Because the low volume of data elected to stick with CSVs as our storage method and used GitHub to store updated versions of our data

Project Architecture



Jupyter Notebooks

Ingestion
- CSV's
- Github
- Python & Pandas
- Kaggle & NOAA

Munging & Wrangling
- Pandas
- MatplotLib
- Yellowbrick

Analysis
- Pandas
- MatplotLib
- Seaborn

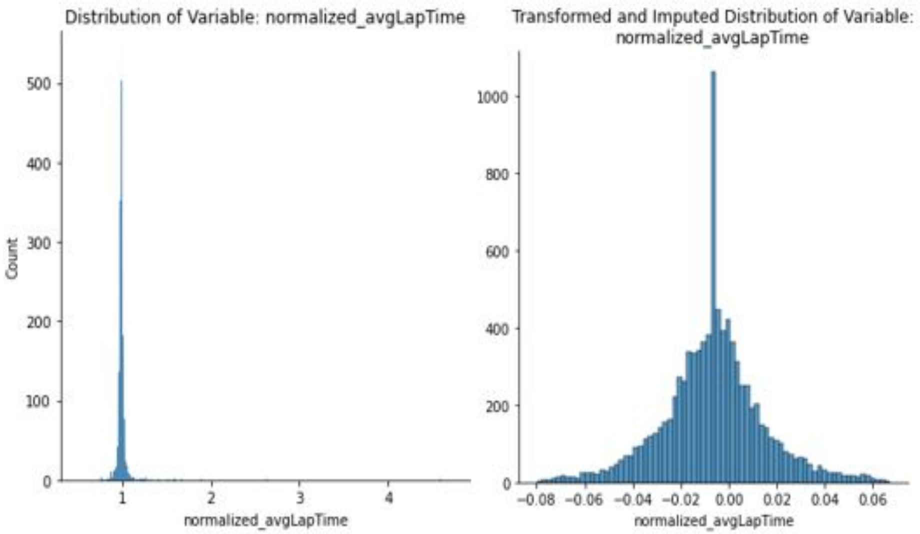
Modeling
- Scikit Learn
- Seaborn

Reporting and Presentation
- Jupyter Widgets
- Matplotlib
- Seaborn

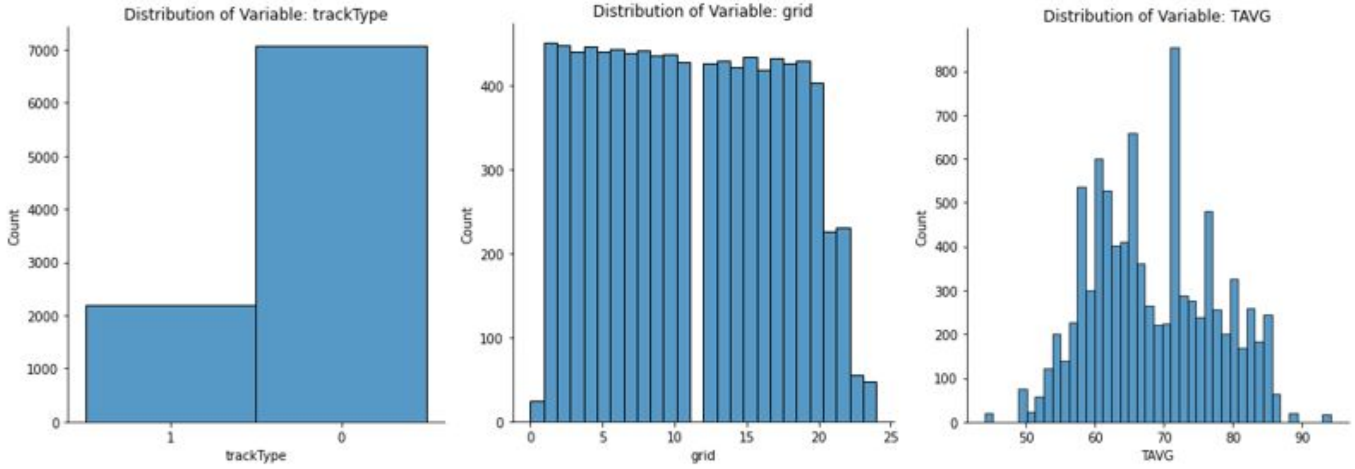
DATA SUMMARY:
FEATURE SELECTION & ENGINEERING

- The construction of our final dataset started with ‘results.csv’. This CSV covers the outcomes of all F1 races within its domain
- In the process of feature selection and engineering, our team discovered issues with the data prior to 1996. We decided to focus our engineering on 1996-present
- We also normalized various other features

Original vs. Transformed Average Lap Time

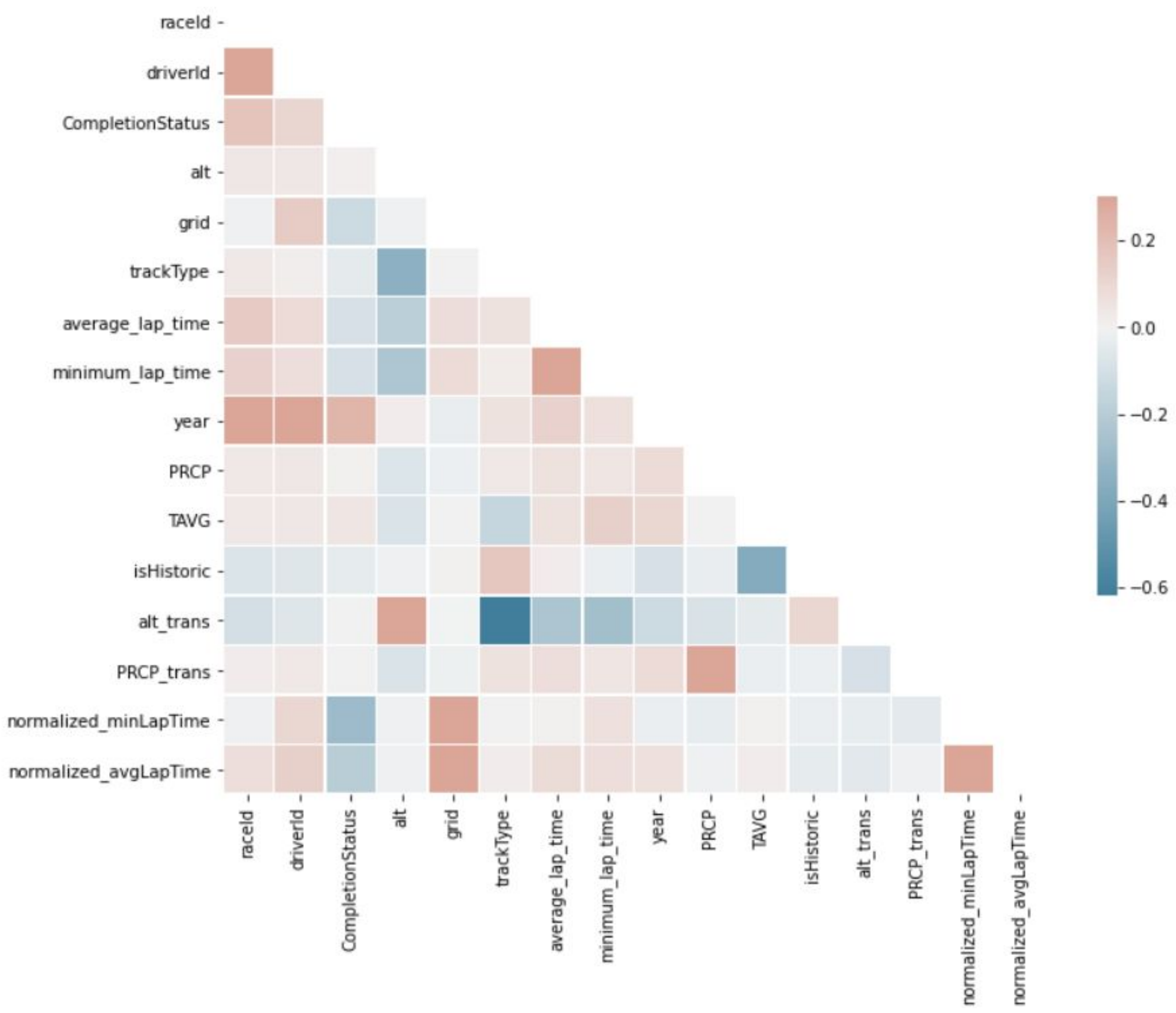


Distribution of some Key Features



DATA SUMMARY:
FEATURE SELECTION & ENGINEERING

Correlation Matrix

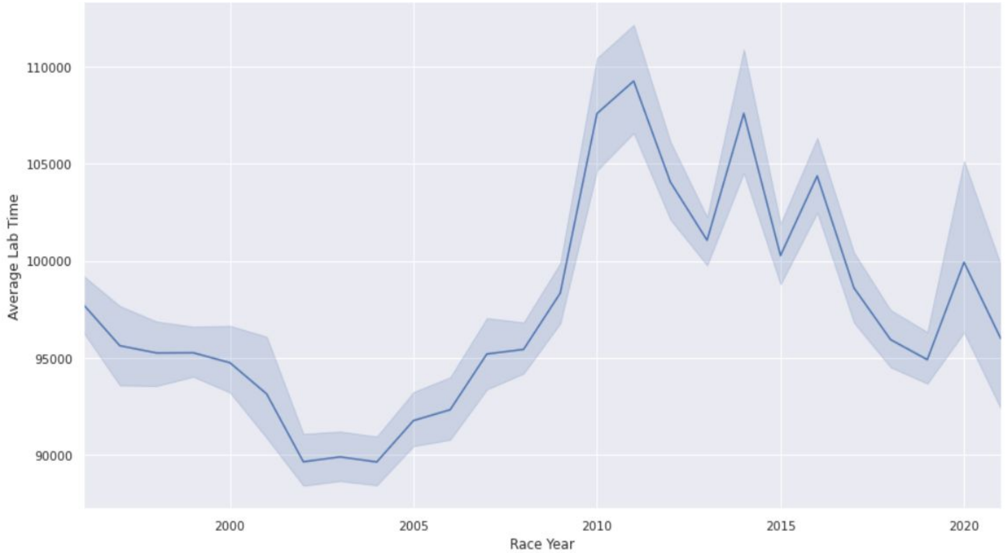


DATA SUMMARY:
FEATURE SELECTION & ENGINEERING

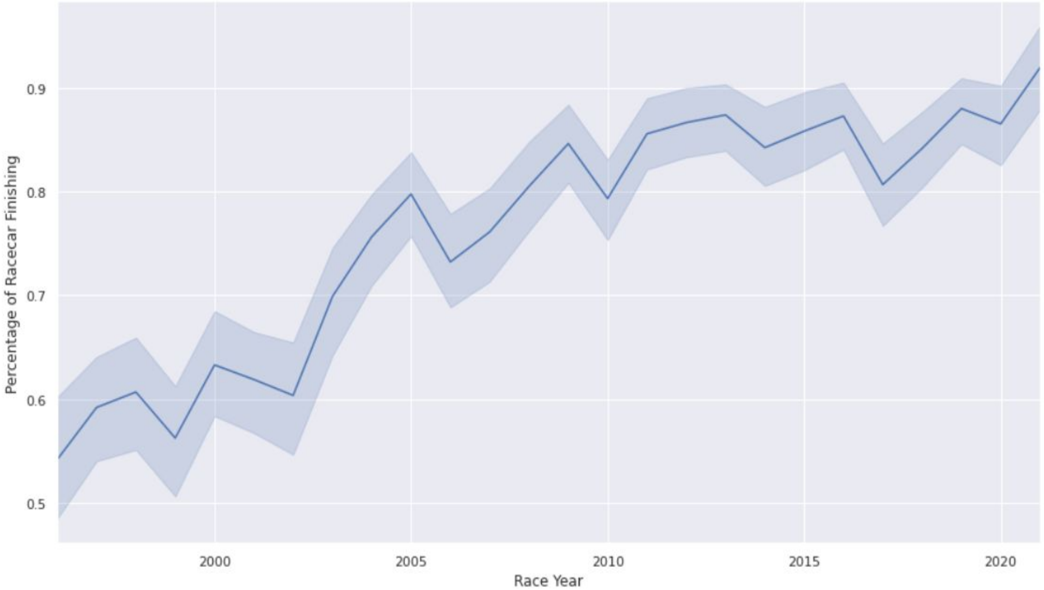
Formula 1 Safety:

- Rules have evolved to create safer and slower cars
- This has resulted in a steady increase in consistency in cars finishing, but a loss in net lap time

Average Lap Time by Year



Percentage of Cars Finishing by Year



MODELING EFFORTS

Modeling Efforts:

- Focus was on using binary classification models to estimate vehicle failure because our data was labeled and we had a binary target. We evaluated our data using:
    - Support Vector Classification (SVC),
    - Nu-Support Vector Classification (NuSVC),
    - Linear Support Vector Classification (LinearSVC),
    - Stochastic Gradient Descent (SGD) Classifier,
    - K-Neighbors Classifier,
    - Logistic Regression,
    - Logistic Regression using Cross Validation (CV),
    - Bagging Classifier,
    - Extra Trees Classifier, and
    - Random Forest Classifier
- We note immediately that we strongly suspect there is a data leak occurring in our modeling workbook. This is an issue we struggled with for the duration of the project and although we implemented several changes, we still suspect the issue persists on at least some level

MODELING EFFORTS

Modeling Efforts:

- After implementing our best preprocessing work, we received the following results, which initially indicated that Random Forest Classifier was the strongest model overall for predicting race outcomes

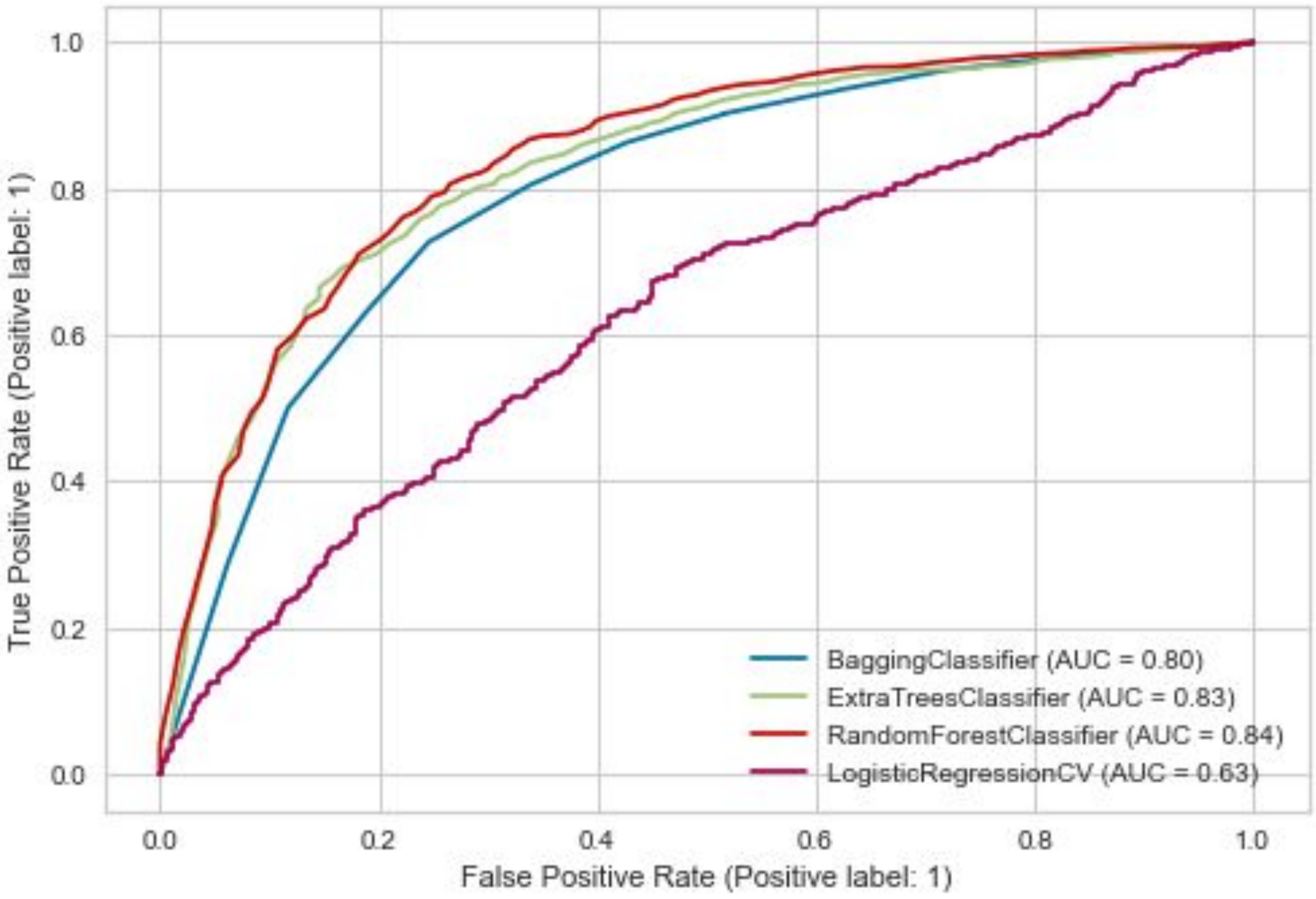
Final Evaluation Results

Model Name	Evaluation Result
Logistic Regression CV	0.8342632097144733
Bagging Classifier	0.8448212311094729
Extra Trees Classifier	0.8811813186813187
Random Forest Classifier	0.8881094352858645

RESULTS & CONCLUSION

ROCAUC Curve Analysis:

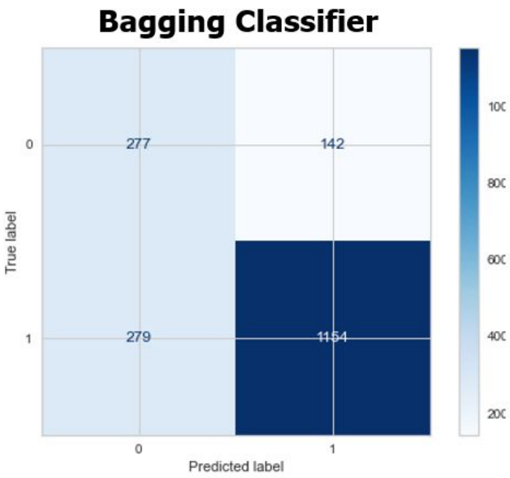
- Bagging Classifier, Extra Trees, and Random Forest continue to show promise

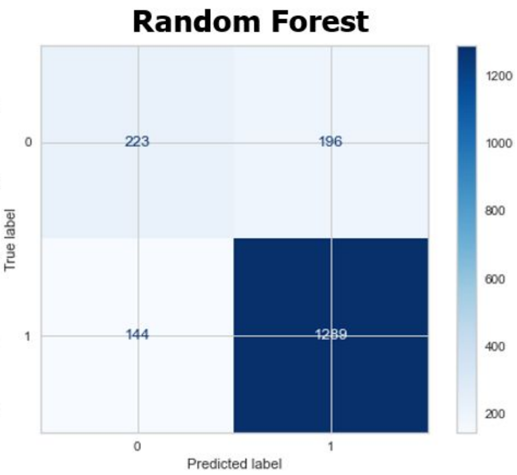


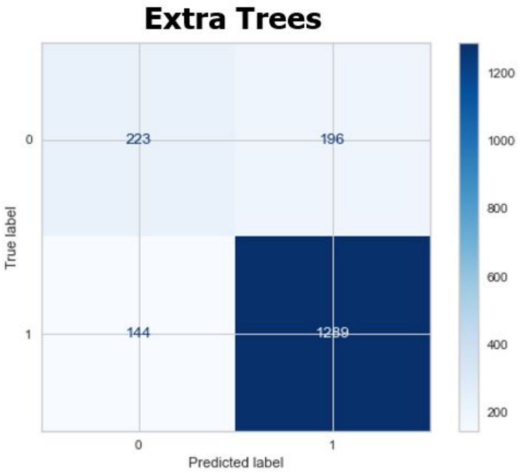
RESULTS & CONCLUSION

Confusion Matrices:

- We prioritized True Negatives


Bagging Classifier


Random Forest


Extra Trees

Bagging Classifier:

(TN / (TN + FP)) = 0.6610978520286396

Extra Trees Classifier:

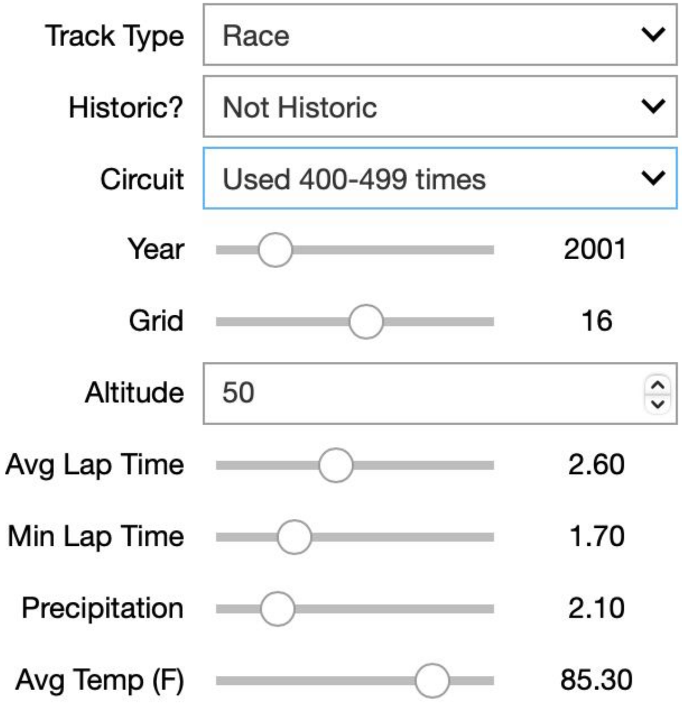
(TN / (TN + FP)) = 0.5322195704057279

Random Forest Classifier:

(TN / (TN + FP)) = 0.6252983293556086

RESULTS & CONCLUSION

Jupyter Widget:



Track Type: Race

Historic?: Not Historic

Circuit: Used 400-499 times

Year: 2001

Grid: 16

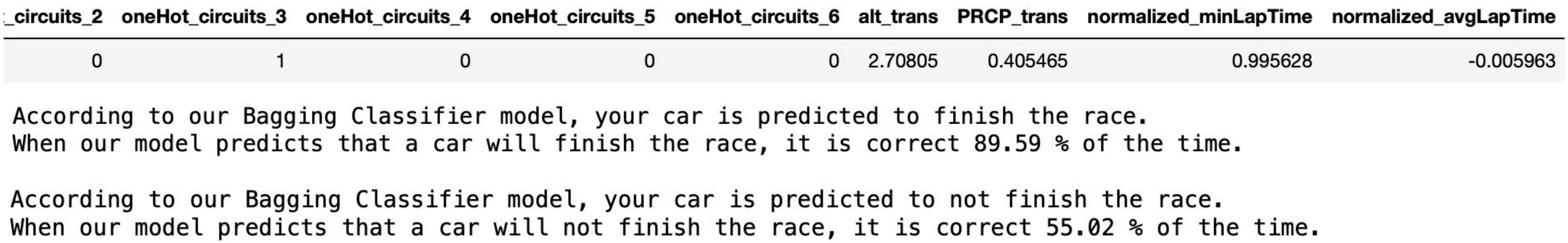
Altitude: 50

Avg Lap Time: 2.60

Min Lap Time: 1.70

Precipitation: 2.10

Avg Temp (F): 85.30



_circuits_2	oneHot_circuits_3	oneHot_circuits_4	oneHot_circuits_5	oneHot_circuits_6	alt_trans	PRCP_trans	normalized_minLapTime	normalized_avgLapTime
0	1	0	0	0	2.70805	0.405465	0.995628	-0.005963

```
According to our Bagging Classifier model, your car is predicted to finish the race.
When our model predicts that a car will finish the race, it is correct 89.59 % of the time.

According to our Bagging Classifier model, your car is predicted to not finish the race.
When our model predicts that a car will not finish the race, it is correct 55.02 % of the time.
```

NEXT STEPS & UNANSWERED QUESTIONS

NEXT STEPS

- Our team identified five improvements that would significantly improve the performance of our models that we did not implement due to a lack of time and know-how.
- The team believes these future deliverables would enhance the overall project value.

Unanswered Questions	Future Work (Top 5)
1. Database	▪Use a PostgreSQL database to store all data. ▪Would enhance the data integrity and accessibility compared to CSVs on GitHub.
2. Multi-class Model	▪Develop a multi-class model to assess the probability of specific types of vehicle failure, such as mechanical issues, collisions, accidents, etc.
3. Ensemble	▪Improve model performance and F1 score with ensembling.
4. Web Application	▪Develop a web application for hosting our final widget. The team discussed using Django to enable users to make real time predict.
5. API Integration	▪ Build two application program interfaces (APIs) to directly pull data from the Formula 1 website (for race data) and the NOAA website (for weather data).

Backup

Literature Review
Lessons Learned
References & GitHub

LITERATURE REVIEW & LESSONS LEARNED

LITERATURE REVIEW

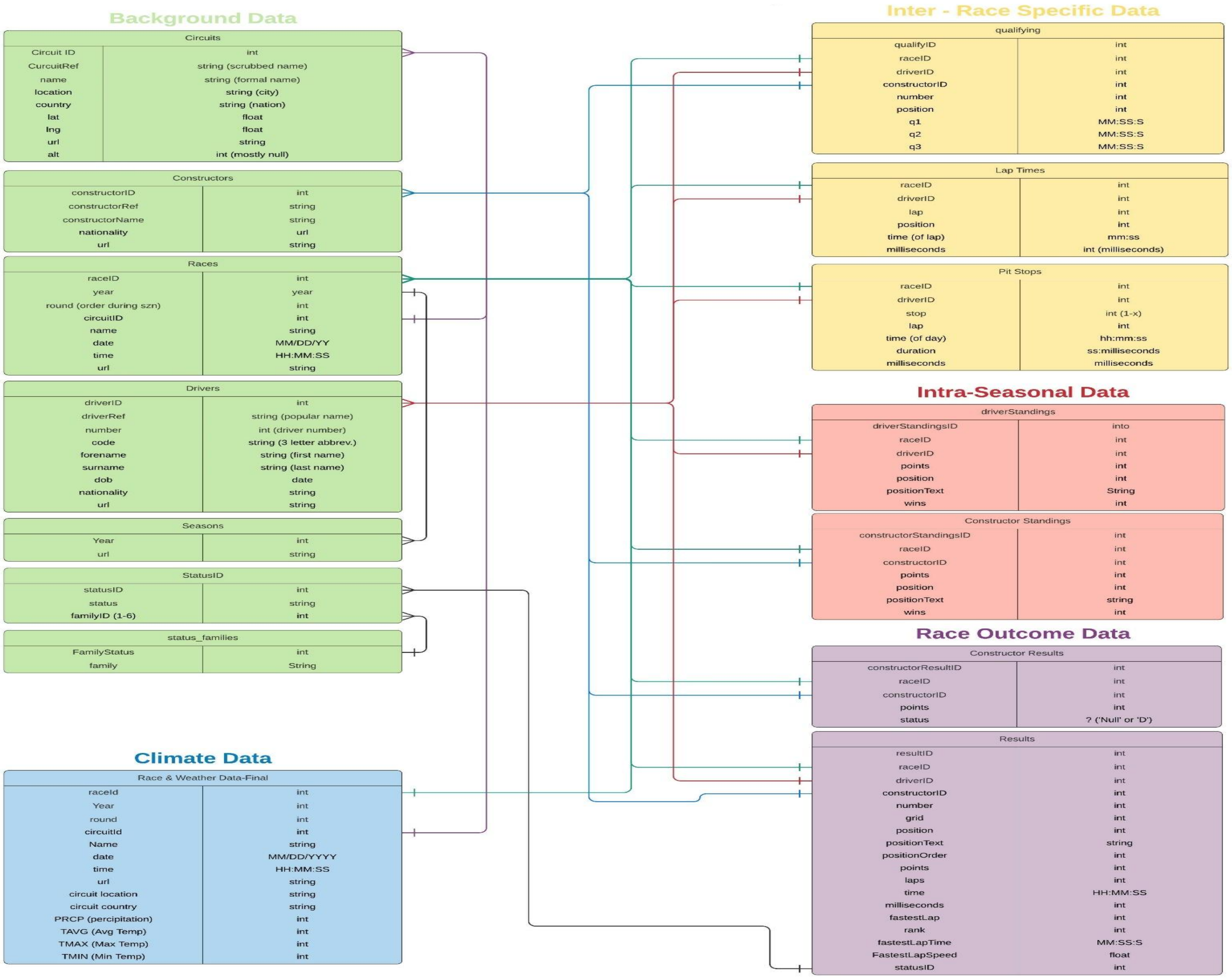
- **"ACCELERATING THE FAN EXPERIENCE: How FORMULA 1 is driving the future of racing using machine learning and AWS,"**
  - Formula 1 describes how they teamed up with AWS to prioritize the fan experience and make the sport more entertaining

LESSONS LEARNED

- **Biggest lesson we learned the hard way during this project was to do your own exploratory data analysis (EDA) and not trust the work of others.**
  - Kaggle data had an almost perfect score. As such, we did not do extensive validation to confirm what data was present.
  - The data replaced null values with "\N", meaning that the cell was functionally null, but counted as filled by Kaggle.
  - This trick made us blind to several issues.
  - As a result, we realized late in the project schedule that we were missing lap times for over two decades of total racing.

Formula 1 Project GitHub Link: https://github.com/georgetown-analytics/Formula1

PROJECT DATA MAP



Background Data

Circuits	
Circuit ID	int
CurcuitRef	string (scrubbed name)
name	string (formal name)
location	string (city)
country	string (nation)
lat	float
lng	float
url	string
alt	int (mostly null)

Constructors	
constructorID	int
constructorRef	string
constructorName	string
nationality	url
url	string

Races	
raceID	int
year	year
round (order during szn)	int
circuitID	int
name	string
date	MM/DD/YY
time	HH:MM:SS
url	string

Drivers	
driverID	int
driverRef	string (popular name)
number	int (driver number)
code	string (3 letter abbrev.)
forename	string (first name)
surname	string (last name)
dob	date
nationality	string
url	string

Seasons	
Year	int
url	string

StatusID	
statusID	int
status	string
familyID (1-6)	int

status_families	
FamilyStatus	int
family	String

Climate Data

Race & Weather Data-Final	
raceId	int
Year	int
round	int
circuitId	int
Name	string
date	MM/DD/YYYY
time	HH:MM:SS
url	string
circuit location	string
circuit country	string
PRCP (percipitation)	int
TAVG (Avg Temp)	int
TMAX (Max Temp)	int
TMIN (Min Temp)	int

Inter - Race Specific Data

qualifying	
qualifyID	int
raceID	int
driverID	int
constructorID	int
number	int
position	int
q1	MM:SS:S
q2	MM:SS:S
q3	MM:SS:S

Lap Times	
raceID	int
driverID	int
lap	int
position	int
time (of lap)	mm:ss
milliseconds	int (milliseconds)

Pit Stops	
raceID	int
driverID	int
stop	int (1-x)
lap	int
time (of day)	hh:mm:ss
duration	ss:milliseconds
milliseconds	milliseconds

Intra-Seasonal Data

driverStandings	
driverStandingsID	into
raceID	int
driverID	int
points	int
position	int
positionText	String
wins	int

Constructor Standings	
constructorStandingsID	int
raceID	int
constructorID	int
points	int
position	int
positionText	string
wins	int

Race Outcome Data

Constructor Results	
constructorResultID	int
raceID	int
constructorID	int
points	int
status	? ('Null' or 'D')

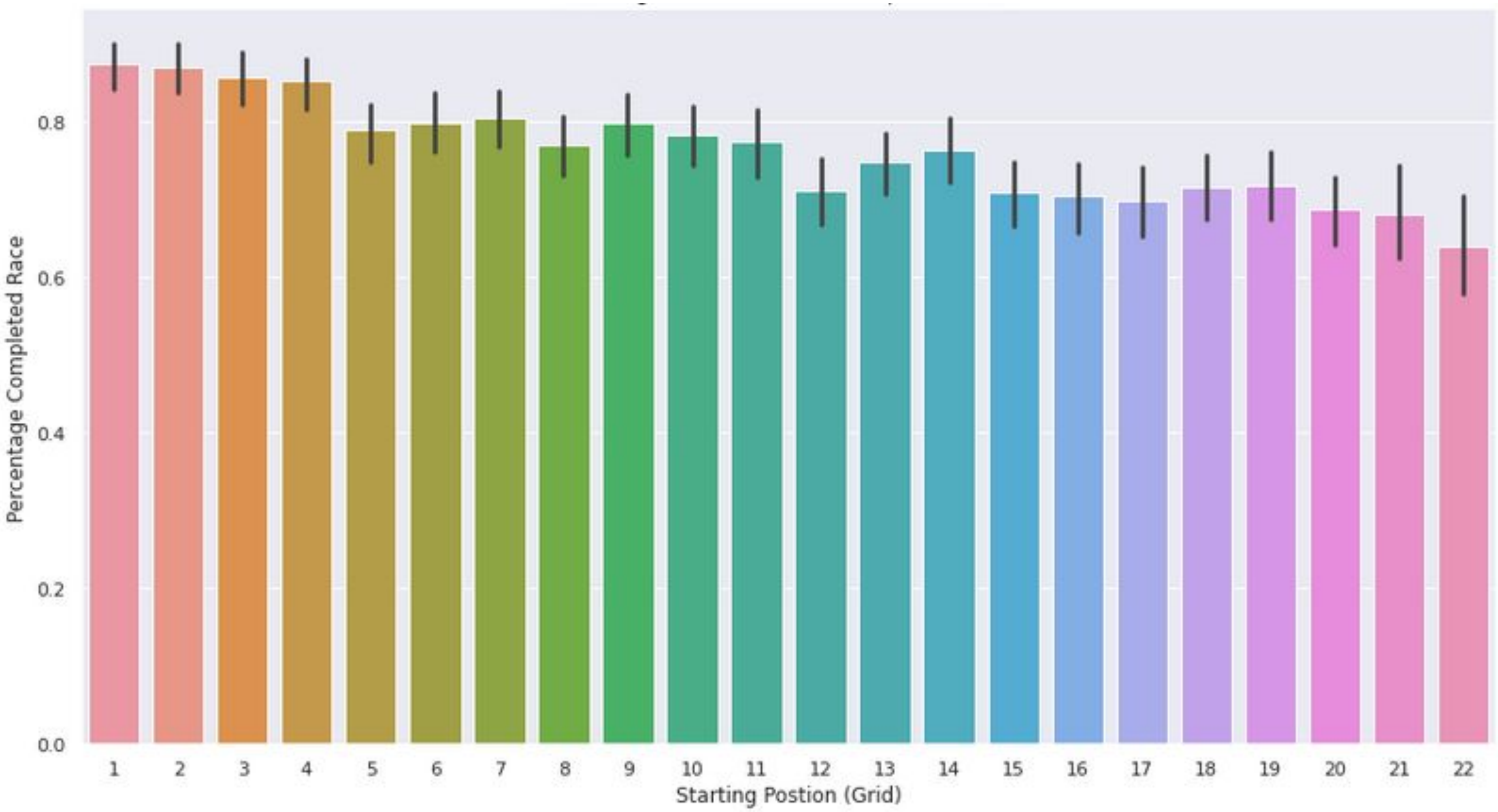
Results	
resultID	int
raceID	int
driverID	int
constructorID	int
number	int
grid	int
position	int
positionText	string
positionOrder	int
points	int
laps	int
time	HH:MM:SS
milliseconds	int
fastestLap	int
rank	int
fastestLapTime	MM:SS:S
FastestLapSpeed	float
statusID	int

DATA SUMMARY:
SELECTION OF TOOLS & STORAGE

DATA SUMMARY:
FEATURE SELECTION & ENGINEERING

- Various features were researched and visualized
- Findings are in Data Visualization Workbook on GitHub
- Example below showing relationship of Starting Position and Race Completion

Percentage of Cars Finishing Race by Starting Position (Grid)


Percentage Completed Race
Starting Postion (Grid)