Classifying the Field: Using Logistic Regression and Neural Networks to Model the Finishing Order of Formula One Races

Nate Everett Downer

Abstract

This report explores the extent to which it is possible to model the finishing order of a Formula One race by using two different classification models. Due to the amount of randomness inherent to motor racing, modeling a race's outcome based on the drivers' past performance is incredibly difficult. Ultimately, this report will show that both Logistic Regression, and Neural Network models are not able to reliably outperform baseline predictors when projecting the full finishing order for an individual race. Instead, these machine learning methods are better suited to modeling a driver's performance over the course of a season. While these models cannot accurately predict who will have a bad race on a given day, they are able to predict who will have more bad days across a number of races.

Additionally this report evaluates how well the predictions of the trained models stack up against actual Formula One fans. By simulating all of the models' predictions for the 2021 season, and comparing them to the predictions submitted to the "Back of the Grid" podcast's predictions league, this report further demonstrates these models' weakness at predicting the outcomes of specific races.

Table of Contents

Project Files

All of the data and code used in this analysis can be viewed on GitHub. Links to specific sources and notebooks can be found in the Endnotes.

I. Introduction

Formula One is perhaps the most data driven sporting competition in the world. Teams routinely use complex models to predict tire wear, plot ideal race strategies, and design the aerodynamic features of their cars. The analysis in this paper, however, seeks to investigate the question that fans are perhaps most interested in: at the end of the race, what order will the drivers cross the finish line in?

Beyond simply predicting a winner, these models will each generate a full running order for every race with as much accuracy as possible. Because of the amount of randomness inherent in motor racing, this is an incredibly difficult task. Ultimately, this paper will conclude that reliable predictions are not possible using classification methods, and will explore the reasons why both Logistic Regression, and Neural Networks fail to accurately predict the outcomes of Formula One races.

II. Data

Formula One races generate enormous amounts of data, and fortunately, quite a lot of it is publicly available. This project is built largely on the back of the incredible Kaggel dataset "Formula 1 World Championship (1950 - 2021)" which is curated and updated by Vopani[1]. While this provided all of the race and qualifying data that was needed, it does not include information about performance in free practice. Before every race, teams are given three practice sessions during which they can familiarize themselves with the track, and refine their car's setup. Data from these sessions is very useful because it can give an early indication of which teams and drivers are struggling at a particular circuit. This data was acquired by scraping Formula One's website.[2]

While this dataset contains the results for races going back to 1950, there are several reasons why using data from that long ago is not a good idea. The scoring system, number of teams, practice format, and nature of the technical regulations have changed so dramatically over the years, that including older data would only add noise to the dataset. Because of this, all data from before 2014 (the start of the so-called "turbo-hybrid" era) was discarded, and not considered for this analysis. What remained was 160 races worth of data. Each race included data from between 20 and 22 drivers, for which 15 factors were analyzed and used to train the model.[3]

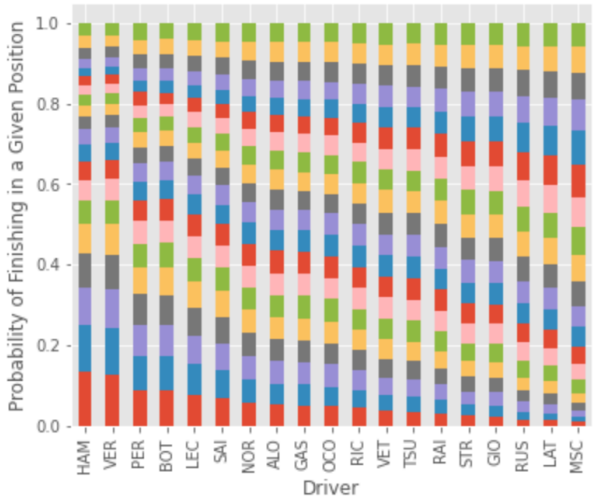
III. Model

Before a model could be created, it was first necessary to determine if the problem was better approached with a regression model, or a classification model. Initially, the fact that we are trying to predict each driver's finishing position (a number within a given range) seems to indicate that a regression model would be the best option, but upon further examination it becomes clear that this method would not work. The main problem is that there is a lot of randomness in car racing, and regression does not do an adequate good of accounting for this.

As an example, imagine we have two drivers. Driver A is a consistent but mediocre driver who usually finishes around the middle of the pack. Driver B is very fast, but they are prone to accidents and do not have a reliable car, meaning that they will either finish very close to the front, or very close to the back depending on how lucky they are on a given day. A model built on regression to the mean would give both of these drivers a similar average finishing position, even though their actual results are wildly different. A classification model works best because it is able to help overcome this problem.

My initial approach was to train a set of logistic regression models to calculate the probability of each driver finishing in each position, and store those probabilities in a table. The algorithm then looks over the table, and finds the prediction that it is most certain about (i.e. the maximum value of the probability table). It records that combination of driver and position in the predictions table, and then drops both that driver, and that position from the probability table. This process is repeated until each driver has been assigned to a position. Figure 1 visualizes what the predictions table for a sample race looks like. In the graph, each slice of the bar represents the probability that the driver will finish in a given position. Positions are arranged with 1st Position at the bottom of the chart, and 20th Position at the top.

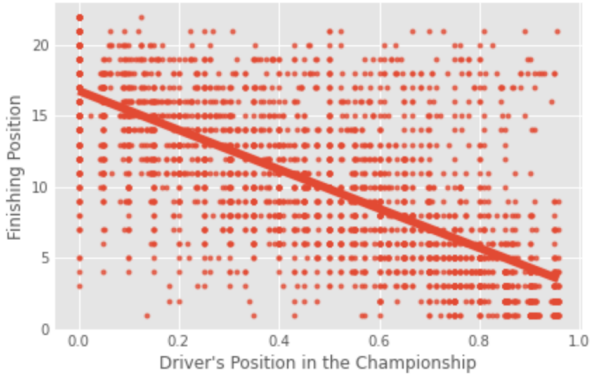
Fig 1 - *Probability Matrix for 2021 Abu Dhabi GP*



While this approach is computationally intensive, it has several major benefits. First, this algorithm can tell the difference between a constant midfield driver, and a fast but inconsistent driver because it evaluates the probability of a driver finishing in each position separately. Second, it starts with the predictions that it is most sure of, and then moves on to more difficult predictions, allowing it to use the process of elimination to make the more difficult predictions easier.

The data used for this model can be broadly broken up into two types of indicators, those reflecting recent performance, and those reflecting overall performance. Indicators for recent performance include the drivers' positions in free practice, as well as their qualifying and finishing positions from the previous three races. For reasons that will be explained in the discussion, data from qualifying, and the third free practice session was not used. The indicators of overall performance include both the driver's current standing in the World Championship and their team's current standing in the Constructors Championship. They also include synthesized factors that represent the drivers' overall experience level, and their average performance at a particular track. Note that their performance from previous years was not included as a factor because the balance of performance between teams changes dramatically from year to year. All variables are scaled to a range between 0 and 1 in order to work efficiently with the logistic regression model. As Figures 2 shows, the data is rather chaotic, but there are clear trends.[4]

Fig 2 - *Finishing Position v. WDC Position*



This chart also illustrates the baselines that the model will be evaluated alongside: the current standings in the World Drivers Championship (WDC). Even though this is a classification algorithm, I found that using root mean squared error -- where the error is the difference between the predicted finishing position and the actual finishing position for each driver -- to score the

results worked well. It provides a measure of how far off the predictions were in general, while penalizing predictions that were further from the mark more severely. I also evaluated what percent of the baseline's predictions were actually correct, but this number is so small that it is not especially useful as an indicator.

I also evaluated how well the finishing order from the last race worked as a baseline predictor; but, as Table 1 shows, both of these baselines do a relatively poor job of accurately predicting the finishing order. They are certainly better than guessing randomly, but not by that much. This poor accuracy is a sign of just how difficult this problem is, and highlights just how much chance plays a factor in the outcome of Formula One races.

Table 1 - *Evaluation of Baseline Measures*

Baseline	RMSE	Pct. Correct
WDC Order	5.50	10.3%
Last Finishing Order	6.41	10.2%
Random Order	11.38	4.9%

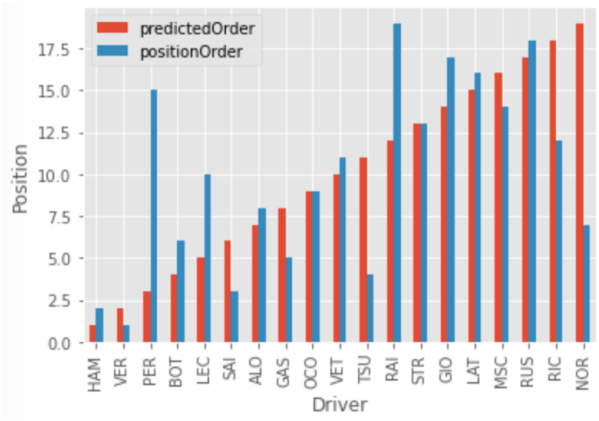
IV. Algorithms

The core of the algorithm used to create this model is the array of logistic regression models used to predict the odds that each driver will finish in each position. A logistic regression model was chosen because they can be trained relatively quickly, and are capable of outputting not just a binary classification, but also the probability that an unknown data point will fall into a particular category. Because of the way the model generates predictions, this is absolutely essential for the model to function.

Tuning the hyper-parameters of this model proved to be a challenge because the result that ultimately needs to be evaluated is aggregated from the outputs of many different models. This makes it very hard to evaluate each of the models in the array separately. In order to overcome this, I defined a function within Python that not only trained the models, but also aggregated the results. The function was defined in such a way that values defining all of the parameters for the logistic regression function could be passed as arguments. This made it possible to run a grid search using various solvers and values for C in order to find the parameters that created the most accurate predictions.

In order to do this, the data was split into training and test sets with a training set of 128 races, and a test set of 32 races. Several different random seeds were used to test the algorithm using different sections of the data, and the final score was created from the average of the scores from the various seeds. Through this tuning process, it was determined that the ideal C value is 0.01, and a Stochastic Average Gradient solver worked best.[5]

Fig 3 - *Predicted Order v. Actual Finishing Order for the 2021 Abu Dhabi Grand Prix*



V. Results

The model based on logistic regression was able to generate finishing orders that seem reasonable, but it ultimately did not perform significantly better than the best baseline predictor (WDC Order). Figure 3, which shows both the model's predicted finishing order for the 2021 Abu Dhabi Grand Prix in red, and the actual finishing order in blue, is a

good example of this. None of the predictions are especially strange, but there is still a lot wrong. Most notably, the model failed to predict that Kimi Räikkönen (RAI) would crash out and subsequently finish in last place. Instead it has the two Mclaren drivers, Lando Noris (NOR) and Daniel Riccardo (RIC), finishing in 19th and 20th. One can only imagine what sort of racing incident it would take to cause this result.

After performing in-sample evaluation on the model (taking the average error from 10 different testing sets, each containing 32 races), the root mean squared error of the logistic regression model was slightly higher than the baseline, but it also managed to make marginally more correct predictions than the baseline.

Table 2 - *Evaluation of The Model*

Predictor	RMSE	Pct. Correct
LogReg Model	5.82	13.3%
WDC Order	5.50	10.3%
Last Finishing Order	6.41	10.2%
Random Order	11.38	4.9%

What is more interesting is that the baseline and the model performed quite differently depending on which finishing position they were modeling. As Figures 4 and 5 show, the logistic regression model is most accurate when predicting the first few finishing positions, and the lower end of the midfield (positions 11-16). However it has a hard time predicting the upper end of the midfield (positions 6-10), and its predictions for positions 19-22 are essentially random. By contrast, the baseline predictor has roughly the same accuracy across all of the midfield positions, but it is even worse at predicting who will finish last.

VI. Analysis

This difference in accuracy depending on the position is a result of the structure of the logistic regression model. Because the algorithm starts by assigning the positions that it is most certain about and then works towards the positions it is least sure of, the performance is not consistent. What this specific pattern or error suggests is that the model has the easiest time identifying which drivers will finish towards the front, and which drivers will constantly finish towards the back.

Fig 4 - *RMSE v. Finishing Position (LogReg Model)*

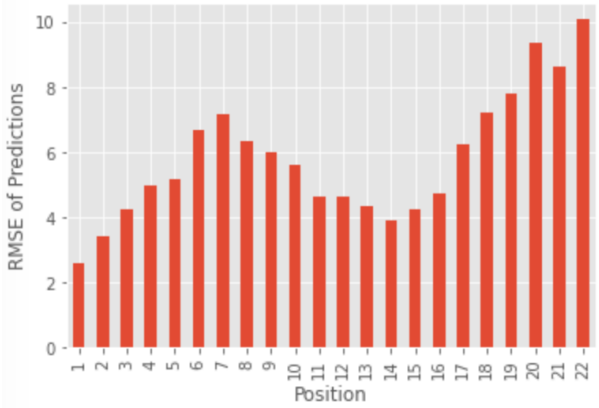
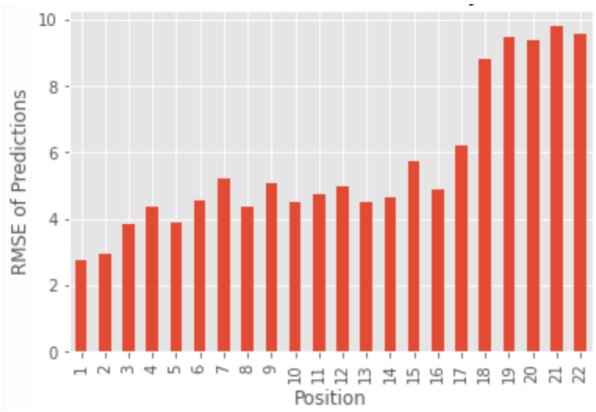


Fig 5 - *RMSE v. Finishing Position (Best Baseline)*



Moreover, both the logistic regression model and the baseline have a very hard time predicting which drivers will finish in the lowest positions. This is because there are many different factors that can cause a driver to finish at the back of the order, and most of them are hard to predict. No matter how fast a driver is, a tire puncture, a collision, a slow pit stop, or a mechanical issue can cause them to plummet to the back of the field or even fail to finish a race.

In Formula 1, drivers who don't complete the full race distance still receive a "finishing" position

with the first driver to retire given the lowest position. Since 2014, the average number of finishers in a race has hovered around 17, meaning that positions below 18th are most often given to drivers who don't finish for one reason or another. This is why we see a corresponding decrease in the accuracy of the predictions for these places. Collisions and mechanical failures are simply very hard to predict based on a driver's past performance.

However, just because the model does a poor job of predicting individual instances of poor performance, that does not mean that it will do a bad job of predicting who will tend to have more collisions or mechanical failures over time. To get a better picture of how well the model could predict aggregate performance, I modeled the results for every race in the 2021 season, and computed how many points each driver would have had at the end of the season, if all of the predictions came true.

Fig 6 - *Predicted Points v. Actual Points for The 2021 Season*

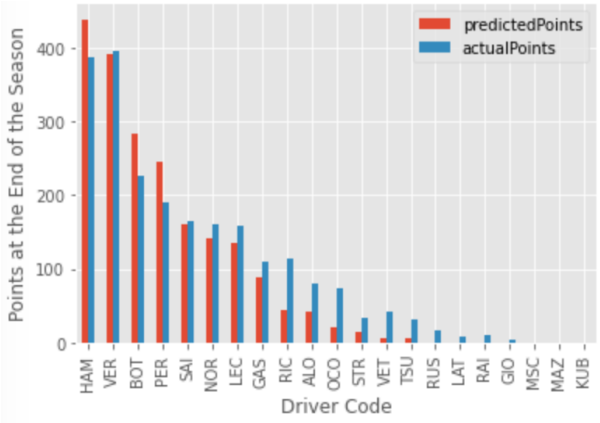


Figure 6 shows how these predicted points totals compare to the actual results, and on the whole they are very close. Every driver is within one place of their actual finishing position. Moveover, the RMSE for the whole data set is 32.7, which is significantly lower than the baseline RMSE of 38.7. This indicates that over time, the model is significantly better than the baseline indicator at identifying which drivers will consistently experience more unforeseen issues.

However, after completing this analysis I was still not satisfied, and was curious to see if it was possible to find more complex relationships between the factors by using a different model. In order to investigate this, I rebuilt the algorithm using an array of neural networks, instead of logistic regression models. The algorithm is structurally the same, except that it uses neural networks to predict the probability that a driver will finish in a given position instead of logistic regression models.

As with the first model, tuning the hyper-parameters proved difficult. It became immediately apparent that the model would be very prone to overfitting, and at first this resulted in incredibly large error values. With repeated testing, I determined that using two hidden layers of neurons with six neurons in layer one, and three neurons in layer two struck the best balance between complexity, and avoiding overfitting.[6]

Table 3 - *Evaluation of The Model*

Predictor	RMSE	Pct. Correct
NN Model	5.88	12.1%
LogReg Model	5.82	13.3%
WDC Order	5.50	10.3%

Even with extensive tuning, the neural network algorithm performed even worse than the logistic regression algorithm. Evaluation of the error broken down by position showed that the neural network model was also subject to the same weaknesses as the logistic regression model. Ultimately there is so much random noise in the data that any attempt to look for "deeper trends" very quickly results in the model fitting itself to the noise, leading to a model that is ultimately weaker.

VII. Deployment

After all of this analysis and development, I was curious to see how well the baseline predictors and the models I created stacked up against real people trying to guess the outcome of Formula One races. To do this, I needed a large set of publicly available predictions made by dedicated Formula One fans. Fortunately, the "Back of the Grid" Formula One podcast hosts a predictions league each season, and they publish all of the predictions made for each race on their website.[7] Each week participants guess (among other things) which driver will finish first, which driver will be the first to retire from the race, and what position a randomly chosen driver will finish in. One point is awarded for each correct prediction.

In order to compare my model's predictions to those made by the actual people competing in the competition, I scrubbed all of the relevant data from the "Back of the Grid" website, and recalculated the scores for each participant using only the three criteria that my models are capable of predicting. I came up with a data set made up of 63 participants, who entered predictions for each race of the season.[8] I then trained both the logistic regression and neural network models on all of the data up through the end of 2020, and used those models to make predictions for each race in the 2021 season. I also evaluated how well the best baseline predictor (Championship Order) would have performed in this simulated predictions league.

Table 4 - *Comparison to Human Predictions*

Predictor	Points	Percentile
Best Human	21	100.0%
Mean	15	50.0%
NN Model	9	3.2%
LogReg Model	8	1.5%
WDC Order	13	27.0%

As it turns out both models performed terribly when faced with this real world test. Both models finished in the bottom 5% of all participants, and failed to get even half the number of correct predictions that the best human managed. This is certainly not what I was expecting. Based on the models performance from in-sample evaluations I projected that both models would score somewhere between 14 and 15 points. In reality they scored 30% fewer points. This further reinforces the conclusion that these models perform well when identifying trends over time, but perform poorly when predicting specific results for specific races.

The relatively poor performance also suggests that there are factors real people are able to account for that the current model is not trained to consider. One such factor is the age of the driver's engine. There was a lot of discussion towards the end of the 2021 season about the relative performance of the drivers' power units based on their age. Many observers credited Lewis Hamilton's wins in Brazil and Qatar to the fact that he had recently installed a fresh power unit. Having no way to factor this in, both algorithms instead predicted that Max Verstappen would win in Brazil in Qatar. Other factors that could be included to refine the model will be discussed in part IX.

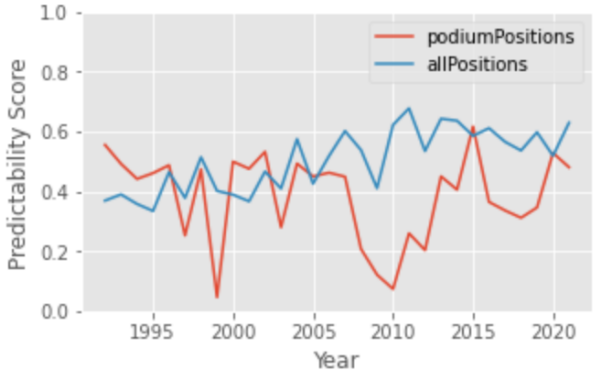
VIII. Discussion

So what does this analysis say about the sport of Formula One? Like all sports, Formula One is thrilling because the outcomes are determined by a mix of skill and luck. As fans, we don't want the outcome of a race to be entirely determined by the skill of the teams and drivers, lest the competition become too predictable, while at the same time we don't want the result to be completely random because then the competition would lose all of its meaning. This analysis often makes reference to the amount of randomness inherent to the results of Formula One races, but can that randomness be quantified? To investigate this, I have created a

metric that I call “Predictability Score”, and charted it for every season since 1990.

The Predictability Score is found by looking at the Pearson Correlation between a driver’s finishing position, and their current position in the Driver’s Championship. This means that seasons with a predictability score closer to one tended to have more races where drivers finished close to their current position in the championship. Seasons with a low predictability score featured more upsets, and more surprise performances. Figure 7 shows the predictability score for all positions in blue, as well as the predictability score for the podium positions in red. This shows that while the sport has generally become more predictable over time (likely as a result of increased reliability, and a decrease in the number of teams competing), the predictability of the podium places still swings dramatically from year to year.

Fig 7 - *Predictability of Race Results Over Time*



As one might expect, the two seasons with the least predictable podiums from the last 30 years featured two of the most exciting title battles of all time. The 2010 season featured five winners in the first seven races, and a record ten different changes in the lead of the Championship. It went down to the wire with Sebastian Vettel claiming the title by a four point margin at the final race. The 1999 season, meanwhile, featured a four way title battle which Mika Häkkinen won at the last race by two points. Further disorder was added to the season by the fact that Michale Schumacher missed five races due to an injury. By contrast, the most predictable season was 2015, which was dominated by Lewis Hamilton and Mercedes. Over the course of the season, Mercedes only failed to get both of their cars on the podium three times.

Ultimately, even though it was not possible to create a model that accurately predicts the finishing order of a Formula One race, this is positive indication for the sport, and it means that fans can look forward to more upsets and surprise podiums in the future.

IX. Future Research

The one significant way in which the current model could be improved would be to change the way that it factors in the identity of the team and driver. As designed, the model is blind to who the specific driver it is making predictions for is, and only evaluates them based on their recent performance. Perhaps adding dummy variables with the identity of each team and driver to the data set would improve the model’s performance. This approach would likely be especially effective if used with the Neural Network model, as the trained model could identify patterns particular to each driver and team.

Moveover, it may be possible to create more accurate models by tapping into new and different types of data. Information about the weather, and drivers' relative performance in different types of weather could be used to refine the model further, but given that most races take place in dry conditions, this would not affect the overall performance of the model very much. As mentioned before, the age of each car’s power unit would also be useful to know, but I have not been able to find a reliable source for that data.

Endnotes and Appendices

1. The Formula 1 World Championship (1950 - 2021) dataset can be found at:
   https://www.kaggle.com/rohanrao/formula-1-world-championship-1950-2020

2. The Free Practice dataset was scrubbed from:
   https://www.formula1.com/en/results.html
   The notebook used to compile and clean the data can be viewed here:
   https://github.com/nate-downer/classifying-the-field/blob/main/web-scrapping-programs/practice-data-scrubber.ipynb

3. The following factors were used as inputs to the model:
   1. The driver's position from first Free Practice session
   2. The driver's position from second Free Practice session
   3. The driver's finishing positions from the previous six races
   4. The driver's qualifying positions from the previous four races
   5. The driver's current position in the World Drivers Championship
   6. The driver's team's current position in the World Constructors Championship
   7. The driver's relative performance at the track
   8. The driver's level of experience, relative to the other drivers on the grid

4. The notebook used to perform exploratory data analysis can be viewed here:
   https://github.com/nate-downer/classifying-the-field/blob/main/Exploratory%20Data%20Analysis.ipynb

5. The notebook used to train, tune, and evaluate the logistic regression model can be found here:
   https://github.com/nate-downer/classifying-the-field/blob/main/Logistic%20Regression%20Model.ipynb

6. The notebook used to train, tune, and evaluate the neural network model can be found here:
   https://github.com/nate-downer/classifying-the-field/blob/main/Neural%20Network%20Model.ipynb

7. The "Back of the Grid" podcast is hosted by Tom King, Chris Evans, and Stu Greenwood. The results from their predictions league can be found here:
   https://backofthegrid.com/prediction-results#
   The notebook used to compile and clean the predictions data can be viewed here:
   https://github.com/nate-downer/classifying-the-field/blob/main/web-scrapping-programs/BOTG-results-scrubber.ipynb