F1 Driver Retention Prediction

# Project Objective

## Background

* Formula 1 is the highest class of open wheel single seater racing and there is a large
* The World Driver’s Championship has been contested every year since 1950 and while the sport in this era began as a hobby for wealthy socialites, it has evolved into a commercial juggernaut with billions being paid to the most successful drivers and teams, known as constructors
* The sport has altered dramatically over the last 20 years, the target of my project, changing hands from Bernie Ecclestone’s F1 Group to Chase Carey’s Liberty Media
* Thanks to the dramatic 2021 title fight between Lewis Hamilton and Max Verstappen viewers are at their highest ever levels globally since 2008, with Sky recently attracting 2.3 million viewers in the UK alone for a race, the highest ever audience for a pay per view broadcaster

## Driver prediction importance

* Predicting the result of a race consistently, thanks to random variables such as reliability, accidents and multiple potential winners, is almost impossible without diving into very specific details only available to manufacturers
* However, there is still merit in seeing whether a driver leaves a team or not
* Being able to make driver predictions could be beneficial for teams to know if a driver maybe considering retirement, and also if results are mirroring those of a driver that leaves, to consider whether they should be making a better financial decision as well
* It is worth considering that a driver is sometimes kept for more than just driver merit, for example Lance Stroll is know as a ‘pay driver’ as his father invests a lot of money in the team he drives for and is unlikely to do this if he doesn’t drive
* This level of data is not readily available and so this will not be studied, although I will attempt to see of there is correlation between constructor and driver nationality as sometimes this can affect keeping a driver on for team popularity in a home country

## Task and Deliverables

* The task is to deliver a logistic model, that is able to determine whether a driver will leave a team (1) or stay at the team (0) for the following season
* The past 20 seasons will be looked at, starting with the 2002 season to the most recent 2021 season
* As drivers are known for the next season, this will be a supervised learning project
* This will be based on the team they begin the new season with, occasionally drivers are swapped mid-season or are replaced by a test driver for injury, and recently for Covid protocols

# Project Features

## Approach

* I will first begin by identifying a data set with many potential attributes, with an effective composite index of year and driver name (although this will have an ID index for ease)
* I will try identifiers to churning

## Metrics

* Using accuracy, precision, recall and f1 score to see whether the correct predictions from the regression method are matched to the binary
* I am not aware of any particular projects that study this however generally good scores for this project type will likely be around an accuracy of 0.75 or higher, this is hopefully what I can achieve
* High accuracies are probably the best metric, sometimes drivers are promoted as well as demoted so it would be hard to say we want to be able to identify truer positive than other for example

## Human Expertise

* My domain expertise is that I am a real diehard fan and have been for many years
* This gives me an understanding of which metrics are likely to give the best models and investigate these further
* I am also aware that F1 is one of the most data intensive industries on the planet and while I am looking at simple predictions for a basic problem
* So much more data is out there privately that could help me predict this problem and potentially work on a fluid unsupervised model, which I’m sure teams are out there using to calculate salaries for their drivers

## Assumptions

* A ‘new’ team signing a ‘new’ driver is treated as such, although I am aware team rebranding will most likely affect my model
* Driver retirements and drivers with year gaps, are treated as leaving teams
* Driver politics, such as paid drivers and racer-team fallout does not occur
* The modal class for a driver’s constructerId will be used in each year to assume what team they were racing for (sometimes drivers switch teams mid-season)

# Data

## Data sources

* The following data source I will be using comes from:
  + Kaggle – Formula 1 World Championships (1950 – 2021)
    - <https://www.kaggle.com/rohanrao/formula-1-world-championship-1950-2020?select=driver_standings.csv>
    - These zipped csv files are available with a public domain license and take up combined data space of 20MB, so a relatively small total file size
  + The credits from Kaggle also use:
    - The ergast.com API
    - Wikipedia: {x} Formula One World Championship pages, URLs listed in the .csv files
* The data contains no sensitive information that needs to be redacted
* The test set will be identified by using a train-test-split method during modelling

## Understanding CSV files

* Drivers – this lists driver information
  + Primary key: driverId
  + Key columns: forename, surname, dob, nationality
* Races – this list information about each individual race that has occurred
  + Primary key: raceId
  + Key columns: year, round
* Results – this lists each driver’s result in each raceId, and who they were racing for
  + Primary key: resultId
  + Key columns: raceId, driverId, constructorId, grid, positionOrder, points
* Driver\_standings – the cumulative results after each race over the course of a season for all drivers
  + Primary key: driverStandingId
  + Key columns: raceId, driverId, points, position, wins
* ConstructorId – constructor information
  + Primary key: constructorId
  + Key columns: name, nationality
* Constructor\_results – individual constructors results for each race
  + Primary key: constructorResultsId
  + Key columns: raceId, constructorId, points
* Constructor\_standings – cumulative results and standing after each race across a season
  + Primary key: constructorStandingsId
  + Key columns: raceId, constructorId, points (cumulative), position, wins, cumulative

# Noticed CSV Observations

* Driver\_standings: raceId 1074 is erroneous, 223
* Constructor standings: raceId 1074 is erroneous, also 36-52
  + These are identified in the 2007 season, McLaren were awarded no constructors points as a result of a spying scandal
  + The drivers would’ve effectively scored 218 between them
* However, race 1074 is a future race yet to exist in 2022 so we will end up blanking this
* Results errors: 732, 741, 728
* These other erroneous results will be from unused years
* In certain seasons there are a large number of drivers caused by weird season structures in early F1 championships
  + From 1996 to present, there were no more than 28 drivers for a season, this should hopefully prevent random skew from one time drivers who never returned to F1

# Feature information

Based on the disparity, or lack of disparity shown

## Final Columns

* Numerical\_columns = ['points\_driver', 'median\_position', 'podiums', 'wins', 'median\_start\_position', 'num\_races', 'percentage\_races', 'full\_season', 'position\_driver', 'points\_constructor', 'position\_constructor', 'driver\_age', 'percentage\_of\_constructor', 'position\_gain', 'nationality\_match', 'out\_perform\_constructor', 'podium\_scored']
* Y\_column = ['driver\_move']

## Good features

* Out\_perform\_constructor: less than -5, high churn, more than 5 high churn, -1, to 1 low churn
* Median\_position: low churn less than 6, high churn less than
* Podium\_scored
* Full\_season
* Percentage\_of\_constructor: high churn <0.2 and >0.8

## Bad features

* Nationality match

# Modelling improvements

* 0.65 to 0.69 with just improving a logistic regression model
* 0.69 to 0.74 utilising better and different models

# Model limitations

* Makes predictions based on a whole season’s results
  + Often we would need to see
* New team branding recorded as driver leaving
* Limited data to model on
* Driver performance and schedule consistency too low pre 2000

# Potential Project Improvements

* Studying additional

# Formatting

* Digital Future Blue
  + RGB = 0, 100, 220
  + HEX = #0064dc
* Title = 3.16, 13.75: 2.34, 1.39
* Main = x, 13.75: 2.34, 4.55