F1 Module

Shane Hauck

2022-11-15

**Formula One** The recent Netflix Docuseries Formula One: Drive to Survive has gripped audiences into becoming enthralled in the once pedestrian sport of Formula One. The growing popularity has caused for the viewers of the sport to become more interested in the specifics of the complexities of the cars. With the advancements of technology over time, it is easy to assume that cars should have gotten faster overtime. With the vast budget differential across teams, the access to better technology varies. Given a wide variety of data, lets see if we can come to a reasonable conclusion about whether cars have gotten equally faster over time.

Load tidyverse package

library(tidyverse)

Load the necessary data for this exercise

qualifying <- read\_csv("data/qualifying.csv")  
head(qualifying)

## # A tibble: 6 × 9  
## qualifyId raceId driverId constructorId number position q1 q2 q3   
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>  
## 1 1 18 1 1 22 1 1:26.572 1:25.1… 1:26…  
## 2 2 18 9 2 4 2 1:26.103 1:25.3… 1:26…  
## 3 3 18 5 1 23 3 1:25.664 1:25.4… 1:27…  
## 4 4 18 13 6 2 4 1:25.994 1:25.6… 1:27…  
## 5 5 18 2 2 3 5 1:25.960 1:25.5… 1:27…  
## 6 6 18 15 7 11 6 1:26.427 1:26.1… 1:28…

drivers <- read\_csv("data/drivers.csv")  
head(drivers)

## # A tibble: 6 × 9  
## driverId driverRef number code forename surname dob nation…¹ url   
## <dbl> <chr> <chr> <chr> <chr> <chr> <date> <chr> <chr>  
## 1 1 hamilton "44" HAM Lewis Hamilton 1985-01-07 British http…  
## 2 2 heidfeld "\\N" HEI Nick Heidfeld 1977-05-10 German http…  
## 3 3 rosberg "6" ROS Nico Rosberg 1985-06-27 German http…  
## 4 4 alonso "14" ALO Fernando Alonso 1981-07-29 Spanish http…  
## 5 5 kovalainen "\\N" KOV Heikki Kovalainen 1981-10-19 Finnish http…  
## 6 6 nakajima "\\N" NAK Kazuki Nakajima 1985-01-11 Japanese http…  
## # … with abbreviated variable name ¹​nationality

races <- read\_csv("data/races.csv")  
head(races)

## # A tibble: 6 × 18  
## raceId year round circuitId name date time url fp1\_d…¹ fp1\_t…²  
## <dbl> <dbl> <dbl> <dbl> <chr> <date> <chr> <chr> <chr> <chr>   
## 1 1 2009 1 1 Australia… 2009-03-29 06:0… http… "\\N" "\\N"   
## 2 2 2009 2 2 Malaysian… 2009-04-05 09:0… http… "\\N" "\\N"   
## 3 3 2009 3 17 Chinese G… 2009-04-19 07:0… http… "\\N" "\\N"   
## 4 4 2009 4 3 Bahrain G… 2009-04-26 12:0… http… "\\N" "\\N"   
## 5 5 2009 5 4 Spanish G… 2009-05-10 12:0… http… "\\N" "\\N"   
## 6 6 2009 6 6 Monaco Gr… 2009-05-24 12:0… http… "\\N" "\\N"   
## # … with 8 more variables: fp2\_date <chr>, fp2\_time <chr>, fp3\_date <chr>,  
## # fp3\_time <chr>, quali\_date <chr>, quali\_time <chr>, sprint\_date <chr>,  
## # sprint\_time <chr>, and abbreviated variable names ¹​fp1\_date, ²​fp1\_time

circuits <- read\_csv("data/circuits.csv")  
head(circuits)

## # A tibble: 6 × 9  
## circuitId circuitRef name locat…¹ country lat lng alt url   
## <dbl> <chr> <chr> <chr> <chr> <dbl> <dbl> <chr> <chr>  
## 1 1 albert\_park Albert Park G… Melbou… Austra… -37.8 145. 10 http…  
## 2 2 sepang Sepang Intern… Kuala … Malays… 2.76 102. 18 http…  
## 3 3 bahrain Bahrain Inter… Sakhir Bahrain 26.0 50.5 7 http…  
## 4 4 catalunya Circuit de Ba… Montme… Spain 41.6 2.26 109 http…  
## 5 5 istanbul Istanbul Park Istanb… Turkey 41.0 29.4 130 http…  
## 6 6 monaco Circuit de Mo… Monte-… Monaco 43.7 7.42 7 http…  
## # … with abbreviated variable name ¹​location

constructors <- read\_csv("data/constructors.csv")  
head(constructors)

## # A tibble: 6 × 5  
## constructorId constructorRef name nationality url   
## <dbl> <chr> <chr> <chr> <chr>   
## 1 1 mclaren McLaren British http://en.wikipedia.org/w…  
## 2 2 bmw\_sauber BMW Sauber German http://en.wikipedia.org/w…  
## 3 3 williams Williams British http://en.wikipedia.org/w…  
## 4 4 renault Renault French http://en.wikipedia.org/w…  
## 5 5 toro\_rosso Toro Rosso Italian http://en.wikipedia.org/w…  
## 6 6 ferrari Ferrari Italian http://en.wikipedia.org/w…

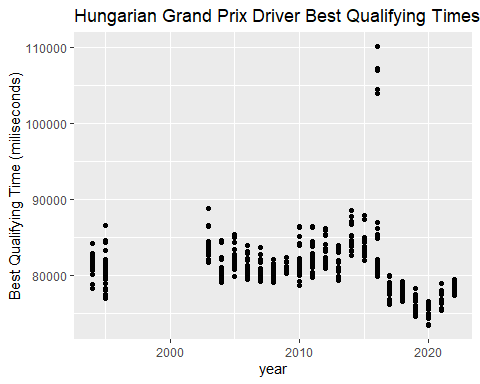
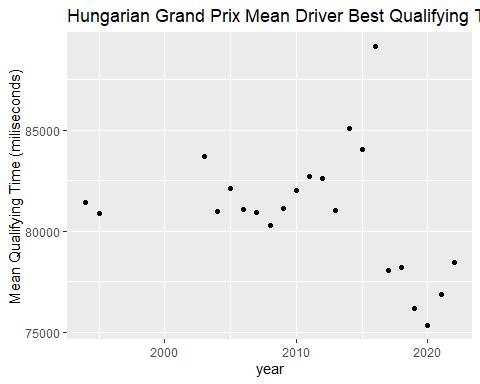
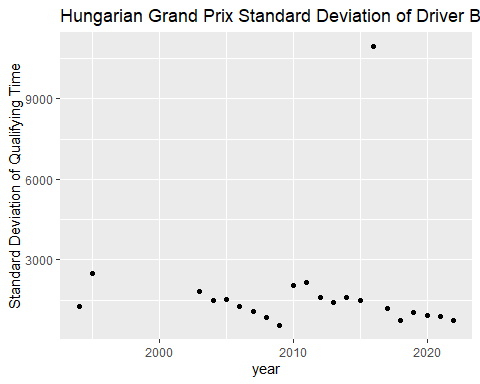
1. Given the datasets above, list all of the foreign keys and what tables they are linked to (hint: there are 4)
2. Using the data and the foreign keys you just determined, create a dataset by joining that looks like. (You are given more than you need. Focus on selecting only the columns required for the end product.)

head(f1\_all)

## # A tibble: 6 × 17  
## ...1 qualifyId raceId drive…¹ const…² posit…³ q1 q2 q3 year circu…⁴  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr> <dbl> <dbl>  
## 1 1 1 18 1 1 1 1:26… 1:25… 1:26… 2008 1  
## 2 2 2 18 9 2 2 1:26… 1:25… 1:26… 2008 1  
## 3 3 3 18 5 1 3 1:25… 1:25… 1:27… 2008 1  
## 4 4 4 18 13 6 4 1:25… 1:25… 1:27… 2008 1  
## 5 5 5 18 2 2 5 1:25… 1:25… 1:27… 2008 1  
## 6 6 6 18 15 7 6 1:26… 1:26… 1:28… 2008 1  
## # … with 6 more variables: name <chr>, driverRef <chr>, forename <chr>,  
## # surname <chr>, constructorRef <chr>, constructerName <chr>, and abbreviated  
## # variable names ¹​driverId, ²​constructorId, ³​position, ⁴​circuitId

1. The columns q1, q2, and q3 represent lap times that we are about to analyze. Describe what is wrong with the way we are currently representing the data.
2. To best fix this issue we want to convert the columns to represent the times as milliseconds. The lubridate package will be useful here. Google “how to convert factor of minutes:seconds:miliseconds to miliseconds using lubridate”.

Click on the first result(stackoverflow) use the example to figure out how to convert q1, q2 and q3 to miliseconds.

1. Now that q1, q2 and q3 can be compared, create a new variable that has the fastest time for each driver for each qualifying.
2. From this point forward lets only focus on the Hungarian Grand Prix. Create a dataset that only features only the best qualifying times from the Hungarian Grand Prix.
3. Create these 4 plots
4. All best qualifying times over year 
5. Mean qualifying time for each year over year 
6. Standard deviation for each year over year 
7. Mean divided by standard deviation over year 