HW MD 3 and 4

Emilio Horner

2022-09-08

R Markdown

1.

```
library(tidyverse)
```

```
## Warning in system("timedatectl", intern = TRUE): running command 'timedatectl'
## had status 1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
mario_kart <- read_csv("http://raw.githubusercontent.com/NicolasRestrep/223_course/main/Data/world_reco
```

```
## Rows: 2334 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (6): track, type, shortcut, player, system_played, time_period
## dbl  (2): time, record_duration
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse (mario_kart)
```

```
## Rows: 2,334
## Columns: 9
## $ track           <chr> "Luigi Raceway", "Luigi Raceway", "Luigi Raceway", "Lu~
## $ type            <chr> "Three Lap", "Three Lap", "Three Lap", "Three Lap", "T~
## $ shortcut        <chr> "No", "No", "No", "No", "No", "No", "No", "No", "No", ~
## $ player          <chr> "Salam", "Booth", "Salam", "Salam", "Gregg G", "Rocky ~
## $ system_played   <chr> "NTSC", "NTSC", "NTSC", "NTSC", "NTSC", "NTSC", "NTSC"~
## $ date            <date> 1997-02-15, 1997-02-16, 1997-02-16, 1997-02-28, 1997-~
## $ time_period     <chr> "2M 12.99S", "2M 9.99S", "2M 8.99S", "2M 6.99S", "2M 4~
## $ time            <dbl> 132.99, 129.99, 128.99, 126.99, 124.51, 122.89, 122.87~
## $ record_duration <dbl> 1, 0, 12, 7, 54, 0, 0, 27, 0, 64, 3, 0, 90, 132, 1, 74~
```

```
three_laps <- mario_kart %>% filter(type == "Three Lap")
NoRainbow <- three_laps |>
```

```
    filter((track != "Rainbow Road"))
```

```
three_laps <- mario_kart %>% filter(type == "Three Lap")
OnlyRainbow <- three_laps |>
  filter(track == "Rainbow Road")
```

2.

```
OnlyRainbow |>
  summarize(mean = mean(time), std_dev = sd(time))
```

```
## # A tibble: 1 x 2
##    mean std_dev
##   <dbl>   <dbl>
## 1  276.    91.8
```

```
NoRainbow |>
  summarise(mean=mean(time), std_dev = sd (time))
```

```
## # A tibble: 1 x 2
##    mean std_dev
##   <dbl>   <dbl>
## 1  114.    53.0
```

The time it takes to complete the Rainbow Road track is longer than the other tracks on average. Additionally it has a higher standard deviation

3.

```
three_laps |>
  group_by(track) |>
  summarize(count = n()) |>
  arrange(desc (count))
```

```
## # A tibble: 16 x 2
##    track                count
##    <chr>                <int>
##  1 Toad's Turnpike        124
##  2 Rainbow Road            99
##  3 Frappe Snowland         92
##  4 D.K.'s Jungle Parkway   86
##  5 Choco Mountain          84
##  6 Mario Raceway           82
##  7 Luigi Raceway           81
##  8 Royal Raceway           77
##  9 Yoshi Valley            74
## 10 Kalimari Desert         73
## 11 Sherbet Land            73
## 12 Wario Stadium           71
## 13 Koopa Troopa Beach      56
## 14 Banshee Boardwalk       55
## 15 Moo Moo Farm            44
## 16 Bowser's Castle         40
```

Toad's Turnpike has the highest number of records at 124.

```
three_laps |>
  group_by(player, track) |>
```

```
  summarise(count= n()) |>
  arrange(desc (count))
```

```
## `summarise()` has grouped output by 'player'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 306 x 3
## # Groups:   player [60]
##    player   track                 count
##    <chr>    <chr>                 <int>
##  1 Penev    Choco Mountain           26
##  2 Lacey    D.K.'s Jungle Parkway    24
##  3 abney317 Rainbow Road             21
##  4 MR       Toad's Turnpike          20
##  5 MR       Frappe Snowland          18
##  6 Penev    Toad's Turnpike          18
##  7 abney317 Kalimari Desert          16
##  8 MR       Sherbet Land             16
##  9 abney317 Choco Mountain           15
## 10 abney317 Toad's Turnpike          15
## # ... with 296 more rows
```

Penev has the most records at Choco Mountain. He has 26 records

5.

```
three_laps |>
  group_by(track) |>
  summarise(mean(time))
```

```
## # A tibble: 16 x 2
##    track                `mean(time)`
##    <chr>                       <dbl>
##  1 Banshee Boardwalk           126.
##  2 Bowser's Castle             134.
##  3 Choco Mountain               95.2
##  4 D.K.'s Jungle Parkway       101.
##  5 Frappe Snowland              77.1
##  6 Kalimari Desert             126.
##  7 Koopa Troopa Beach           96.6
##  8 Luigi Raceway               104.
##  9 Mario Raceway                79.1
## 10 Moo Moo Farm                 88.4
## 11 Rainbow Road                276.
## 12 Royal Raceway               158.
## 13 Sherbet Land                116.
## 14 Toad's Turnpike             122.
## 15 Wario Stadium               214.
## 16 Yoshi Valley                 82.7
```

Rainbow Road has the highest average time.

```
three_laps %>%
  group_by(player) %>%
  arrange(time) %>%
  slice(1) %>%
  head()
```

```
## # A tibble: 6 x 9
## # Groups:   player [6]
##   track           type   short~1 player syste~2 date       time_~3  time recor~4
##   <chr>           <chr>  <chr>   <chr>  <chr>   <date>     <chr>   <dbl>   <dbl>
## 1 Choco Mountain  Three~ Yes     ABE    NTSC    1997-06-01 1M 39.~  99.8      23
## 2 Choco Mountain  Three~ Yes     abney~ NTSC    2021-02-03 17.29S   17.3      23
## 3 Yoshi Valley    Three~ Yes     Alex G PAL     2010-11-27 33.39S   33.4    3659
## 4 Frappe Snowland Three~ Yes     Allen~ NTSC    1997-10-31 28.22S   28.2       0
## 5 Wario Stadium   Three~ Yes     Ben M  PAL     2002-08-22 15.48S   15.5    1370
## 6 Frappe Snowland Three~ Yes     Booth  NTSC    1998-09-18 27.11S   27.1      74
## # ... with abbreviated variable names 1: shortcut, 2: system_played,
## #   3: time_period, 4: record_duration
```

```
three_laps |>
group_by(track) |>
summarise(min(time))
```

```
## # A tibble: 16 x 2
##    track                 `min(time)`
##    <chr>                       <dbl>
##  1 Banshee Boardwalk           124.
##  2 Bowser's Castle             132
##  3 Choco Mountain               17.3
##  4 D.K.'s Jungle Parkway        21.4
##  5 Frappe Snowland              23.6
##  6 Kalimari Desert             122.
##  7 Koopa Troopa Beach           95.2
##  8 Luigi Raceway                25.3
##  9 Mario Raceway                58.5
## 10 Moo Moo Farm                 85.9
## 11 Rainbow Road                 50.4
## 12 Royal Raceway               119.
## 13 Sherbet Land                 91.6
## 14 Toad's Turnpike              30.3
## 15 Wario Stadium                14.6
## 16 Yoshi Valley                 33.4
```

6.

```
three_laps <- three_laps |>
  mutate(onehundred = ifelse(record_duration > 100, 1, 0))
player_hundred <- three_laps |>
  filter(onehundred == 1) |>
  group_by(player, onehundred) |>
  summarize(count = n()) |>
  arrange(desc (count))
```

```
## `summarise()` has grouped output by 'player'. You can override using the
## `.groups` argument.
```

```
player_hundred
```

```
## # A tibble: 42 x 3
## # Groups:   player [42]
##    player   onehundred count
##    <chr>         <dbl> <int>
##  1 MR                1    81
```

```
##  2 MJ                1    50
##  3 Penev             1    27
##  4 abney317          1    26
##  5 VAJ               1    26
##  6 Zwartjes          1    24
##  7 Lacey             1    23
##  8 Dan               1    21
##  9 Karlo             1    18
## 10 Booth             1    17
## # ... with 32 more rows
```

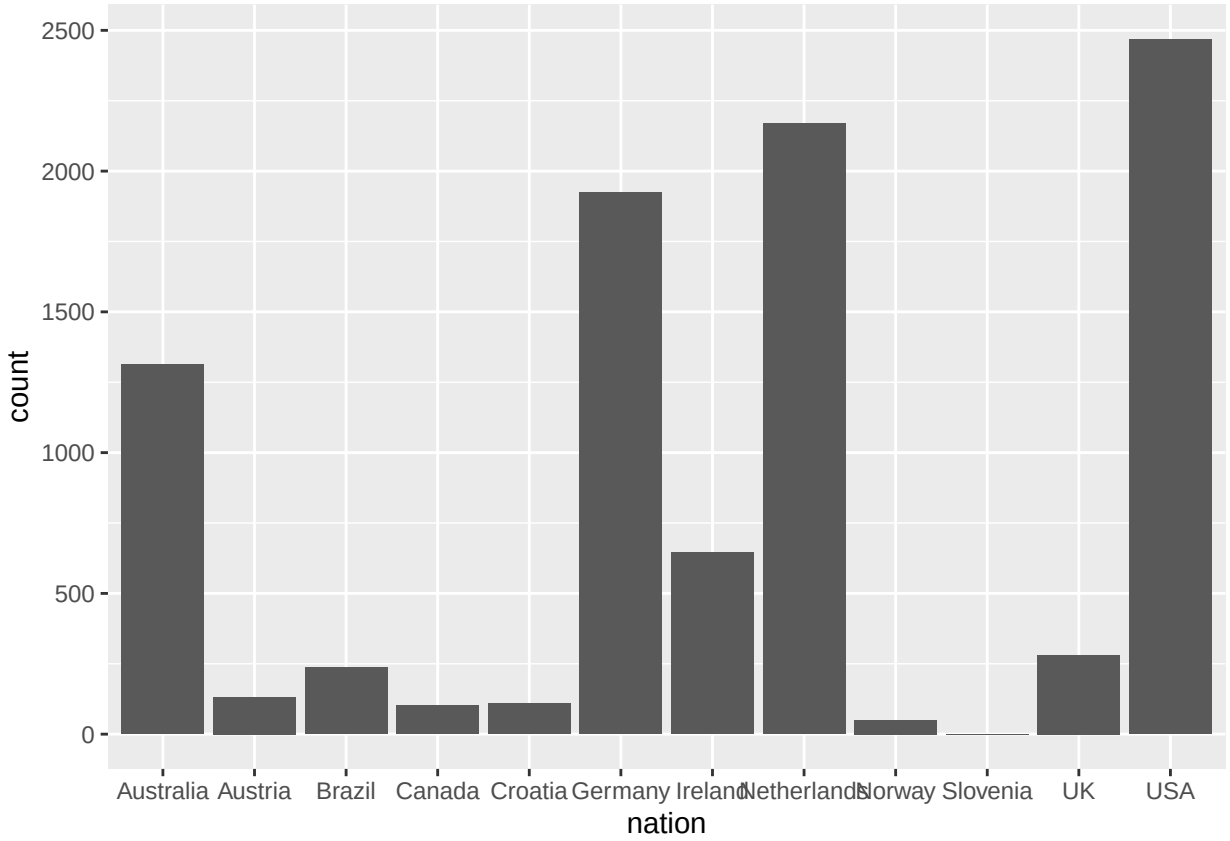
Player MR holds the most long duration records with 81.

7.

```
drivers <- read_csv("https://raw.githubusercontent.com/NicolasRestrep/223_course/main/Data/drivers.csv")
```

```
## Rows: 2250 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): player, nation
## dbl (4): position, total, year, records
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
drivers_joined <- three_laps |>
  left_join (drivers, by = "player") |>
  drop_na()
ggplot(data = drivers_joined, mapping = aes(x = nation)) +
  geom_bar()
```



MD Chapter 4

1.

```
NFL_Salaries <- read_csv("https://raw.githubusercontent.com/NicolasRestrep/223_course/main/Data/nfl_sala
```

```
## Rows: 800 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (11): year, Cornerback, Defensive Lineman, Linebacker, Offensive Lineman...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2.

```
NFL_Salaries_Tidy <- NFL_Salaries |>
  pivot_longer(names_to = "Position",
               values_to = "Salary",
               cols = -year)
NFL_Salaries_Tidy
```

```
## # A tibble: 8,000 x 3
##     year Position            Salary
##    <dbl> <chr>                <dbl>
##  1  2011 Cornerback        11265916
##  2  2011 Defensive Lineman 17818000
##  3  2011 Linebacker        16420000
##  4  2011 Offensive Lineman 15960000
```

```
##  5  2011 Quarterback       17228125
##  6  2011 Running Back      12955000
##  7  2011 Safety             8871428
##  8  2011 Special Teamer     4300000
##  9  2011 Tight End          8734375
## 10  2011 Wide Receiver     16250000
## # ... with 7,990 more rows
```

3.

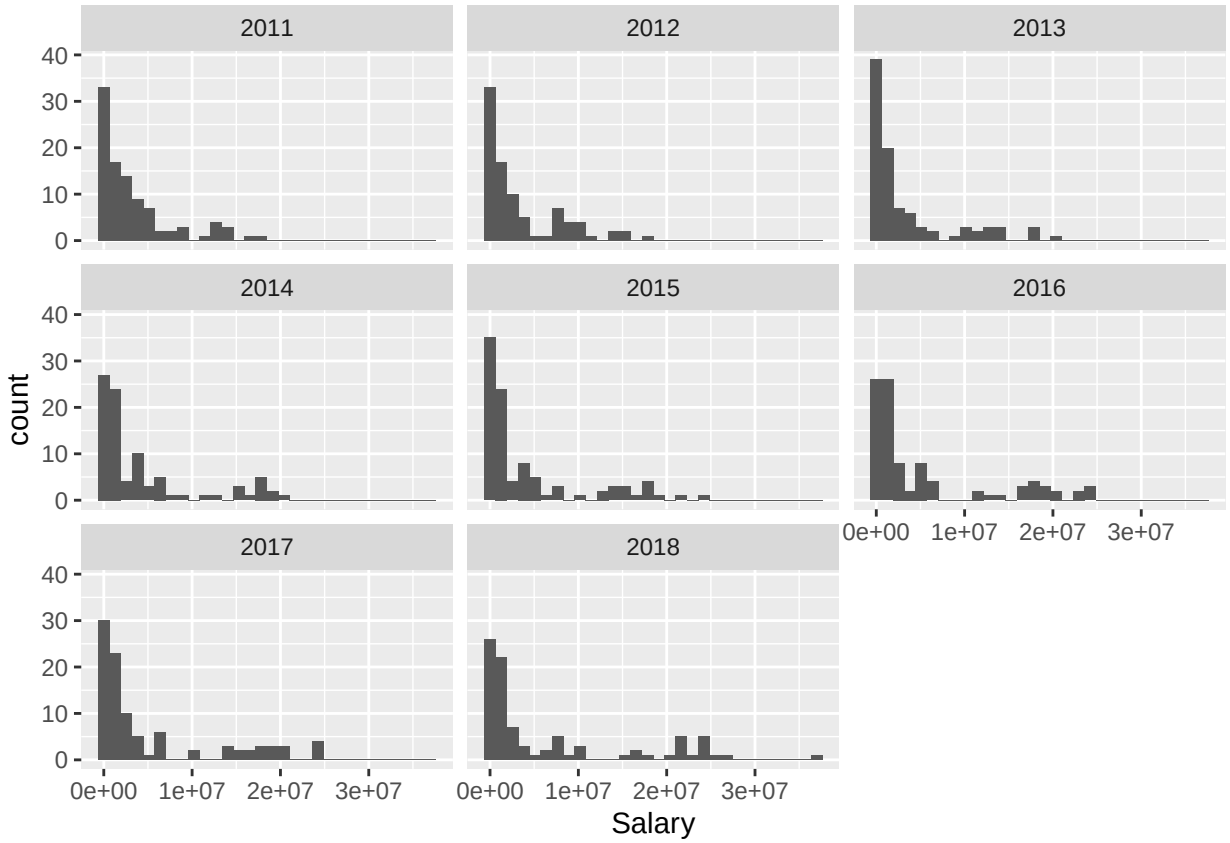
```
Quarterback_Salary <- NFL_Salaries_Tidy |>
  filter(Position== "Quarterback")
  Quarterback_Salary
```

```
## # A tibble: 800 x 3
##     year Position      Salary
##    <dbl> <chr>          <dbl>
##  1  2011 Quarterback 17228125
##  2  2011 Quarterback 16000000
##  3  2011 Quarterback 14400000
##  4  2011 Quarterback 14100000
##  5  2011 Quarterback 13510000
##  6  2011 Quarterback 13250000
##  7  2011 Quarterback 12950000
##  8  2011 Quarterback 12574700
##  9  2011 Quarterback 12465000
## 10  2011 Quarterback 11320000
## # ... with 790 more rows
```

```
ggplot(data = Quarterback_Salary, mapping = aes(x = Salary)) +
  geom_histogram()+ facet_wrap(~ year)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 55 rows containing non-finite values (stat_bin).
```



4.

```
Average_Salaries <- NFL_Salaries_Tidy |>
  group_by(Position, year) |>
  summarise(mean = mean (Salary))
```

```
## `summarise()` has grouped output by 'Position'. You can override using the
## `.groups` argument.
```

```
  Average_Salaries
```

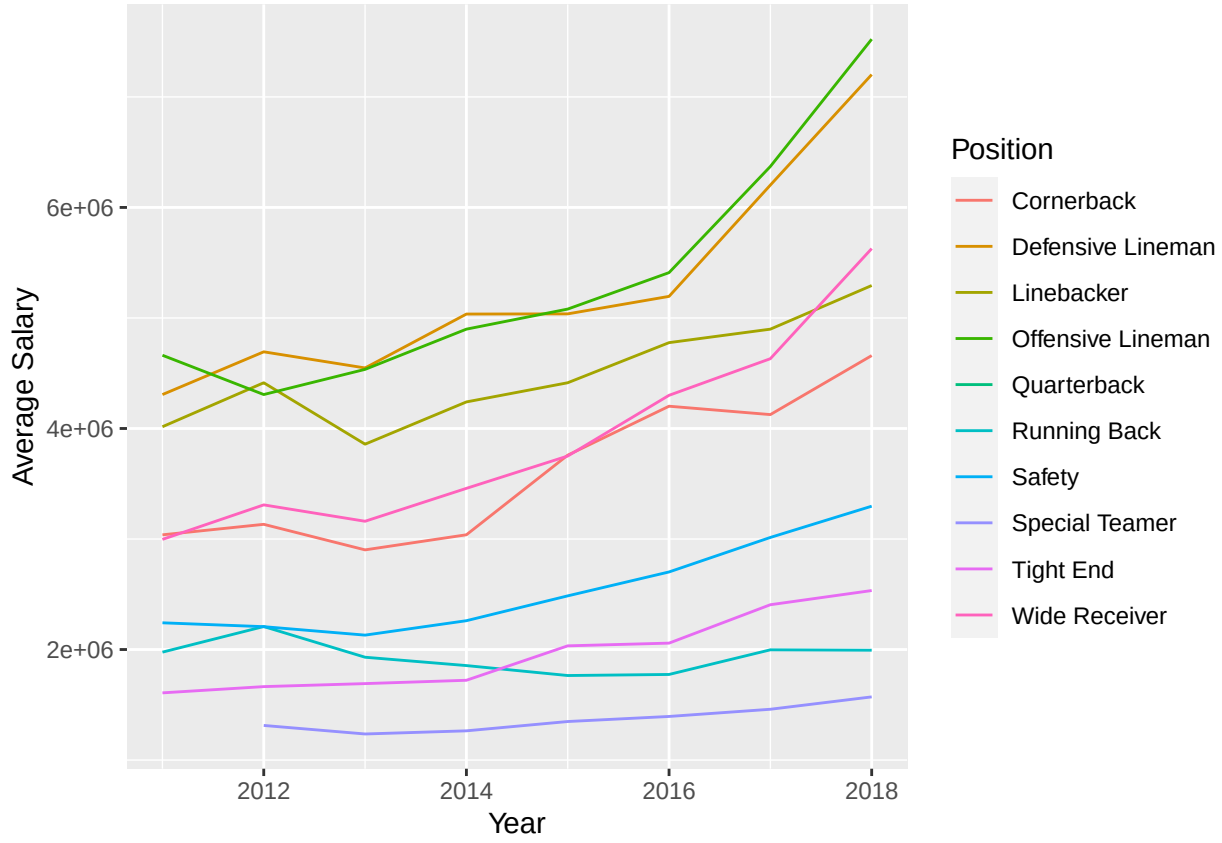
```
## # A tibble: 80 x 3
## # Groups:   Position [10]
##    Position           year     mean
##    <chr>             <dbl>    <dbl>
##  1 Cornerback         2011 3037766.
##  2 Cornerback         2012 3132916.
##  3 Cornerback         2013 2901798.
##  4 Cornerback         2014 3038278.
##  5 Cornerback         2015 3758543.
##  6 Cornerback         2016 4201470.
##  7 Cornerback         2017 4125692.
##  8 Cornerback         2018 4659704.
##  9 Defensive Lineman  2011 4306995.
## 10 Defensive Lineman  2012 4693730.
## # ... with 70 more rows
```

5.

```
ggplot(Average_Salaries, aes(x = year, y = mean, col = Position)) +
  geom_line(position = "dodge") +
  labs(x = "Year", y = "Average Salary")
```

```
## Warning: Width not defined. Set with `position_dodge(width = ?)`
```

```
## Warning: Removed 9 row(s) containing missing values (geom_path).
```



There has been an in pay for Offensive lineman in the last 5 years

Additionally, the salaries for Special Teams has stayed relatively constant over the last 10 years