Chapter 3 and 4 - Modern Dive

Allyson Cameron

2022-09-06

Chapter 3

First, lets load `tidyverse`, `readr`, `dplyr`, and `knitr` and load the data.

```
library(tidyverse)
```

```
## -- 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 1.4.0
## 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()
```

```
library(readr)
library(dplyr)
library(knitr)
mario_kart <-
  read_csv(
    paste0(
      "/Users/allysoncameron/Documents/soc_722_stats/",
      "Data/world_records.csv"
    )
  )
```

```
## 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~
```

Question 1

Now let's filter out only the races with `"Three Lap"` and take out laps from `"Rainbow Road"`.

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

glimpse(three_laps)
```

```
## Rows: 1,112
## 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~
```

Now, let's save a dataset that only contains the records achieved at Rainbow Road.

```
rainbow_road <- mario_kart %>%
  filter(type == "Three Lap" & track == "Rainbow Road")

glimpse(rainbow_road)
```

```
## Rows: 99
## Columns: 9
## $ track           <chr> "Rainbow Road", "Rainbow Road", "Rainbow Road", "Rainb~
## $ type            <chr> "Three Lap", "Three Lap", "Three Lap", "Three Lap", "T~
## $ shortcut        <chr> "No", "No", "No", "No", "No", "No", "No", "No", "No", ~
## $ player          <chr> "Booth", "Jonathan", "Zwartjes", "Jonathan", "Penev", ~
## $ system_played   <chr> "NTSC", "NTSC", "PAL", "NTSC", "PAL", "PAL", "PAL", "P~
## $ date            <date> 1997-05-27, 1997-08-27, 1998-01-14, 1998-03-13, 1998-~
## $ time_period     <chr> "6M 15.83S", "6M 9.67S", "6M 8.69S", "6M 5.51S", "6M 4~
## $ time            <dbl> 375.83, 369.67, 368.69, 365.51, 364.15, 363.86, 362.15~
## $ record_duration <dbl> 92, 140, 58, 173, 9, 2, 9, 8, 9, 1, 14, 113, 65, 8, 35~
```

Question 2

Now, lets get the average `time` at Rainbow Road and the standard deviations.

```
summary_rr <- rainbow_road %>%

  summarize(mean_time = mean(time, na.rm = TRUE),
            sd_record_rr = sd(time, na.rm = TRUE))

summary_rr
```

```
## # A tibble: 1 x 2
##   mean_time sd_record_rr
##       <dbl>        <dbl>
## 1      276.         91.8
```

Let's do the same things for the other dataset with all of the other tracks.

```
summary_three_laps <- three_laps %>%
  summarize(mean_time_3 = mean(time, na.rm = TRUE),
            sd_record__3 = sd(time, na.rm = TRUE))

summary_three_laps
```

```
## # A tibble: 1 x 2
##   mean_time_3 sd_record__3
##         <dbl>        <dbl>
## 1        114.         53.0
```

Notice any differences? The average time for Rainbow Road was significantly longer (275.63) than the average for all other tracks doing three-laps (113.80). Additionally, there is more variation in the times of the records at Rainbow Road (91.82) than at the other tracks with three-laps (52.98).

Question 3

Next we are going to create `three_laps_by_track` which will first look in `three_laps`, then (`%>%`), `group_by` tracks, then (`%>%`), `filter` to only count cases of individuals who actually currently hold a record, then (`%>%`), `summarize` to count how many different records have been established on each track. After this, I will arrange the counts in descending order so that I can see which track has the most records.

```
three_laps_by_track <- three_laps %>%
  group_by(track) %>%
  filter(record_duration != 0) %>%
  summarize(num_three_laps_records = n()) %>%
  arrange(desc(num_three_laps_records))

glimpse(three_laps_by_track)
```

```
## Rows: 15
## Columns: 2
## $ track                  <chr> "Toad's Turnpike", "Frappe Snowland", "D.K.'s J~
## $ num_three_laps_records <int> 86, 82, 80, 80, 77, 70, 70, 70, 65, 64, 55, 53,~
```

Toad's Turnpike has the most, with 86 current records.

Question 4

Now we want to investigate if there are drivers who have multiple records at each track, and how many records they have.

For this, we will be grouping by both driver and track.

```
by_player_each_track <- three_laps %>%
  group_by(player, track) %>%
  filter(record_duration != 0) %>%
  summarize(num_by_player_track = n()) %>%
  arrange(desc(num_by_player_track))
```

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

```
glimpse(by_player_each_track)
```

```
## Rows: 277
## Columns: 3
## Groups: player [52]
## $ player              <chr> "Penev", "Lacey", "MR", "abney317", "MR", "abney31~
## $ track               <chr> "Choco Mountain", "D.K.'s Jungle Parkway", "Frappe~
## $ num_by_player_track <int> 24, 23, 17, 16, 16, 15, 14, 13, 13, 13, 12, 12, 12~
```

Who is the player that has recorded the most records at any one track and what track was it? Player Penev is the player who holds the most records overall (24) and this is on a track called Choco Mountain.

Question 5

Now, I will show you the best time recorded on each track by using `group_by`, `arrange`, and `slice` to see the first (best) time for each. I will also limit which columns show using `select`.

```
best_time_by_track <- three_laps %>%
  group_by(track) %>%
  arrange(time) %>%
  slice(1) %>%
  select(track, time)

glimpse(best_time_by_track)
```

```
## Rows: 15
## Columns: 2
## Groups: track [15]
## $ track <chr> "Banshee Boardwalk", "Bowser's Castle", "Choco Mountain", "D.K.'~
## $ time  <dbl> 124.09, 132.00, 17.29, 21.35, 23.61, 121.92, 95.25, 25.30, 58.49~
```

Question 6

Let's create a new variable that is a 1 if record_duration is higher than 100 or 0 otherwise.

```
three_laps <- three_laps %>%
  mutate(rec_duration_mod = as.numeric(three_laps$record_duration >= 100))

glimpse(three_laps)
```

```
## Rows: 1,112
## Columns: 10
## $ track            <chr> "Luigi Raceway", "Luigi Raceway", "Luigi Raceway", "L~
## $ type             <chr> "Three Lap", "Three Lap", "Three Lap", "Three Lap", "~
## $ 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 ~
## $ time             <dbl> 132.99, 129.99, 128.99, 126.99, 124.51, 122.89, 122.8~
## $ record_duration  <dbl> 1, 0, 12, 7, 54, 0, 0, 27, 0, 64, 3, 0, 90, 132, 1, 7~
## $ rec_duration_mod <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,~
```

Now, let's look at the total amount of long-duration records each player holds.

```
long_duration_by_player <- three_laps %>%
  group_by(player) %>%
  summarize(sum_rec_duration = sum(rec_duration_mod, na.rm = TRUE)) %>%
  arrange(desc(sum_rec_duration))

glimpse(long_duration_by_player)
```

```
## Rows: 57
## Columns: 2
## $ player           <chr> "MR", "MJ", "Penev", "Zwartjes", "Lacey", "VAJ", "abn~
## $ sum_rec_duration <dbl> 76, 47, 24, 24, 23, 23, 21, 20, 16, 16, 12, 10, 10, 9~
```

What player has the most long-duration records? Player MR has the most long-duration records (76).

Question 7

Now, lets import the a data set to join it with our `three_laps` dataset.

```
drivers <-
  read_csv(paste0(
    "/Users/allysoncameron/Documents/soc_722_stats/",
    "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.
```

```
glimpse(drivers)
```

```
## Rows: 2,250
## Columns: 6
## $ position <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ player   <chr> "Penev", "Penev", "Penev", "Penev", "Penev", "Penev", "Penev"~
## $ total    <dbl> 344, 344, 344, 344, 344, 344, 344, 344, 344, 344, 344, 344, 3~
## $ year     <dbl> 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2~
## $ records  <dbl> NA, 181, 126, 14, 5, 11, 2, 5, NA, NA, NA, NA, NA, NA, NA, NA~
## $ nation   <chr> "Australia", "Australia", "Australia", "Australia", "Australi~
```

Let's complete the join using `left_join`.

```
# Let's take out year as it is information we already have.
three_laps_drivers <- three_laps %>%
  left_join(drivers, by = "player") %>%
  select(-year)


glimpse(three_laps_drivers)
```

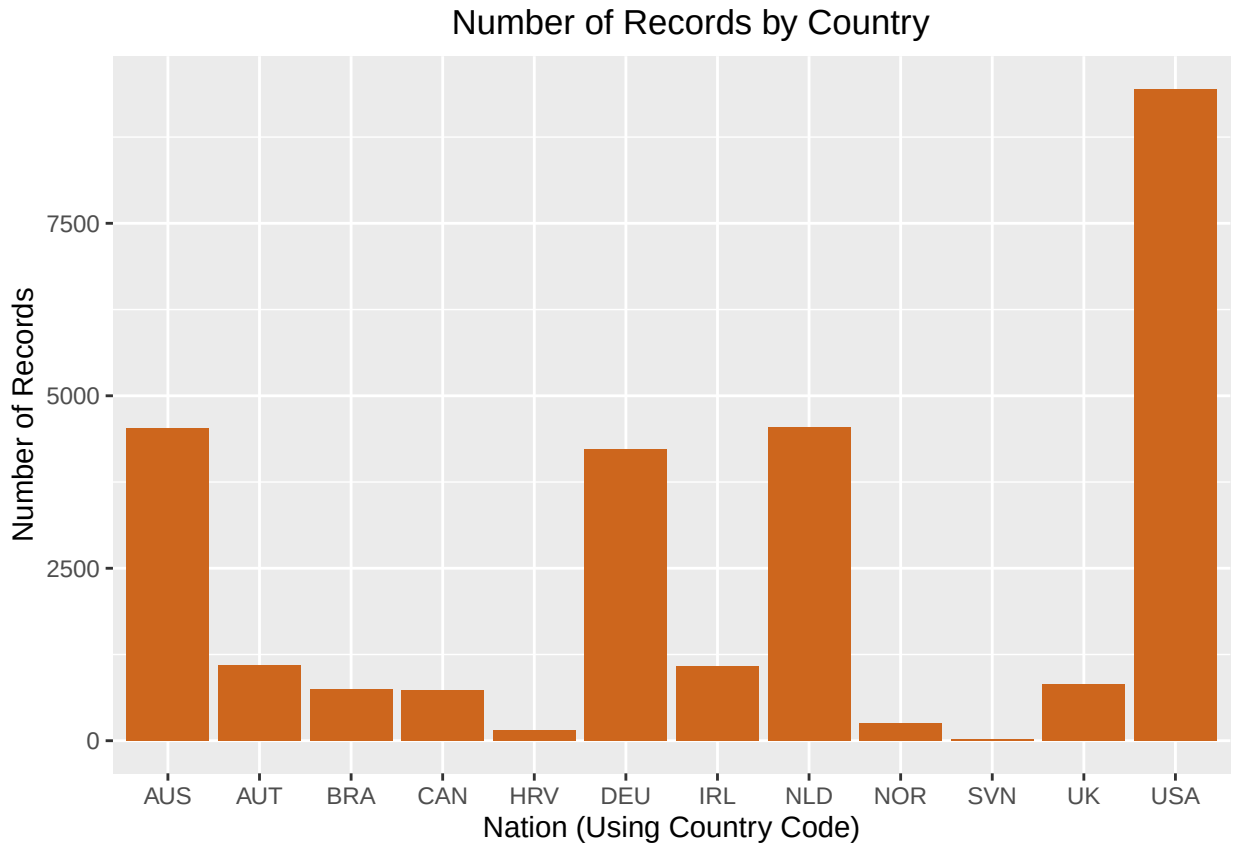
```
## Rows: 27,800
## Columns: 14
## $ track           <chr> "Luigi Raceway", "Luigi Raceway", "Luigi Raceway", "L~
## $ type            <chr> "Three Lap", "Three Lap", "Three Lap", "Three Lap", "~
## $ shortcut        <chr> "No", "No", "No", "No", "No", "No", "No", "No", "No",~
## $ player          <chr> "Salam", "Salam", "Salam", "Salam", "Salam", "Salam",~
## $ system_played   <chr> "NTSC", "NTSC", "NTSC", "NTSC", "NTSC", "NTSC", "NTSC~
## $ date            <date> 1997-02-15, 1997-02-15, 1997-02-15, 1997-02-15, 1997~
## $ time_period     <chr> "2M 12.99S", "2M 12.99S", "2M 12.99S", "2M 12.99S", "~
## $ time            <dbl> 132.99, 132.99, 132.99, 132.99, 132.99, 132.99, 132.9~
## $ record_duration  <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ rec_duration_mod <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ position        <dbl> 42, 42, 42, 42, 42, 42, 42, 42, 42, 42, 42, 42, 42, 4~
## $ total           <dbl> 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,~
## $ records         <dbl> 4, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ nation          <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N~
```

Lastly, let's try to plot a bar chart of number of records by country. Here are the countries and there corresponding codes for your reference.

Country	Country Code
Australia	AUS
Austria	AUT
Brazil	CAN
Canada	HRV
Germany	DEU
Ireland	IRL
Netherlands	NLD
Norway	NOR
Slovenia	SVN

Country	Country Code
United Kingdom	UK
United States	USA

```
# Let's take out the NA values from nations
three_laps_drivers %>%
    filter(!is.na(nation)) %>%
  ggplot(aes(x = nation)) +
  geom_bar(fill = "chocolate3") + labs(x = "Nation (Using Country Code)",
                                       y = "Number of Records",
                                     title = "Number of Records by Country") +
  scale_x_discrete(labels = c("AUS", "AUT", "BRA", "CAN", "HRV", "DEU", "IRL",
                              "NLD", "NOR", "SVN", "UK", "USA")) +
  theme(plot.title = element_text(hjust = 0.53))
```



Chapter 4

Before we begin, let's load `tidyverse`, `dplyr`, and `scales`.

```
library(tidyverse)
library(dplyr)
library(scales)
```

```
##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor
```

Question 1

First, let's import a raw data file from a Git Hub link.

```
nfl_salaries <- read.csv(
  paste0(
    "https://raw.githubusercontent.com/",
    "NicolasRestrep/223_course/main/Data/",
    "nfl_salaries.csv"
  )
)
glimpse(nfl_salaries)
```

```
## Rows: 800
## Columns: 11
## $ year              <int> 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011~
## $ Cornerback        <int> 11265916, 11000000, 10000000, 10000000, 10000000, 92~
## $ Defensive.Lineman <int> 17818000, 16200000, 12476000, 11904706, 11762782, 11~
## $ Linebacker        <int> 16420000, 15623000, 11825000, 10083333, 10020000, 81~
## $ Offensive.Lineman <int> 15960000, 12800000, 11767500, 10358200, 10000000, 98~
## $ Quarterback       <int> 17228125, 16000000, 14400000, 14100000, 13510000, 13~
## $ Running.Back      <int> 12955000, 10873833, 9479000, 7700000, 7500000, 70330~
## $ Safety            <int> 8871428, 8787500, 8282500, 8000000, 7804333, 7652700~
## $ Special.Teamer    <int> 4300000, 3725000, 3556176, 3500000, 3250000, 3225000~
## $ Tight.End         <int> 8734375, 8591000, 8290000, 7723333, 6974666, 6133333~
## $ Wide.Receiver     <int> 16250000, 14175000, 11424000, 11415000, 10800000, 99~
```

Question 2

Now let's tidy up the data and combine the different columns/positions into one column called `positions` and add their values into a separate column called `salaries`.

```
nfl_salaries_tidy <- nfl_salaries %>%
  pivot_longer(names_to = "position",
               values_to = "salaries",
               cols = -year)
glimpse(nfl_salaries_tidy)
```

```
## Rows: 8,000
## Columns: 3
## $ year     <int> 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2~
## $ position <chr> "Cornerback", "Defensive.Lineman", "Linebacker", "Offensive.L~
## $ salaries <int> 11265916, 17818000, 16420000, 15960000, 17228125, 12955000, 8~
```

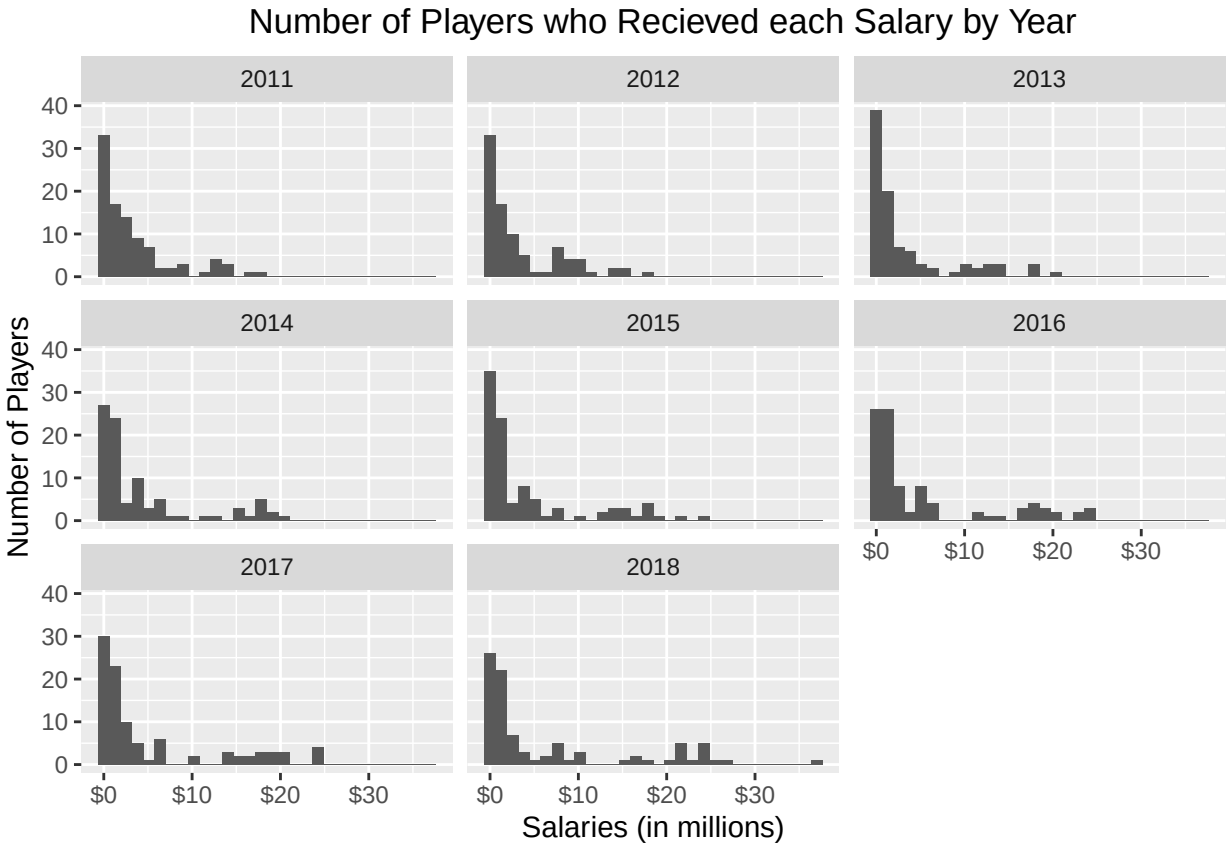
Question 3

Let's make histograms for each year for quarterbacks.

```
# Let's filter out quarter backs first and convert salaries to "in thousands"
qb_only <- nfl_salaries_tidy %>%
  filter(position == "Quarterback")

qb_only  <- qb_only %>%
  mutate(sal_in_millions = qb_only$salaries / 1000000)


#First lets convert our salaries to "in thousands"

# Now let's create our histogram
ggplot(qb_only, aes(x = sal_in_millions)) +
  geom_histogram() +  facet_wrap(~ year) + labs(
    x = "Salaries (in millions)",
    y = "Number of Players",
    title = paste0("Number of",
                   " Players who",
                   " Recieved each",
                   " Salary by Year")
  ) +
  scale_x_continuous(labels = dollar) +
  theme(plot.title = element_text(hjust = 0.53))
```

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

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



What patterns do you notice? I notice that the the distribution is positively (right) skewed meaning that while a majority of quarterbacks make under 10 million, there are some who make well above this. This pattern of skewed-ness is consistent across every year.

Question 4

Now, let's create a new dataset that contains the average salary for each position each year.

```
avg_pos_sal <- nfl_salaries_tidy %>%
  group_by(position, year) %>%
  summarize(avg_salaries = mean(salaries, na.rm = TRUE))
```

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

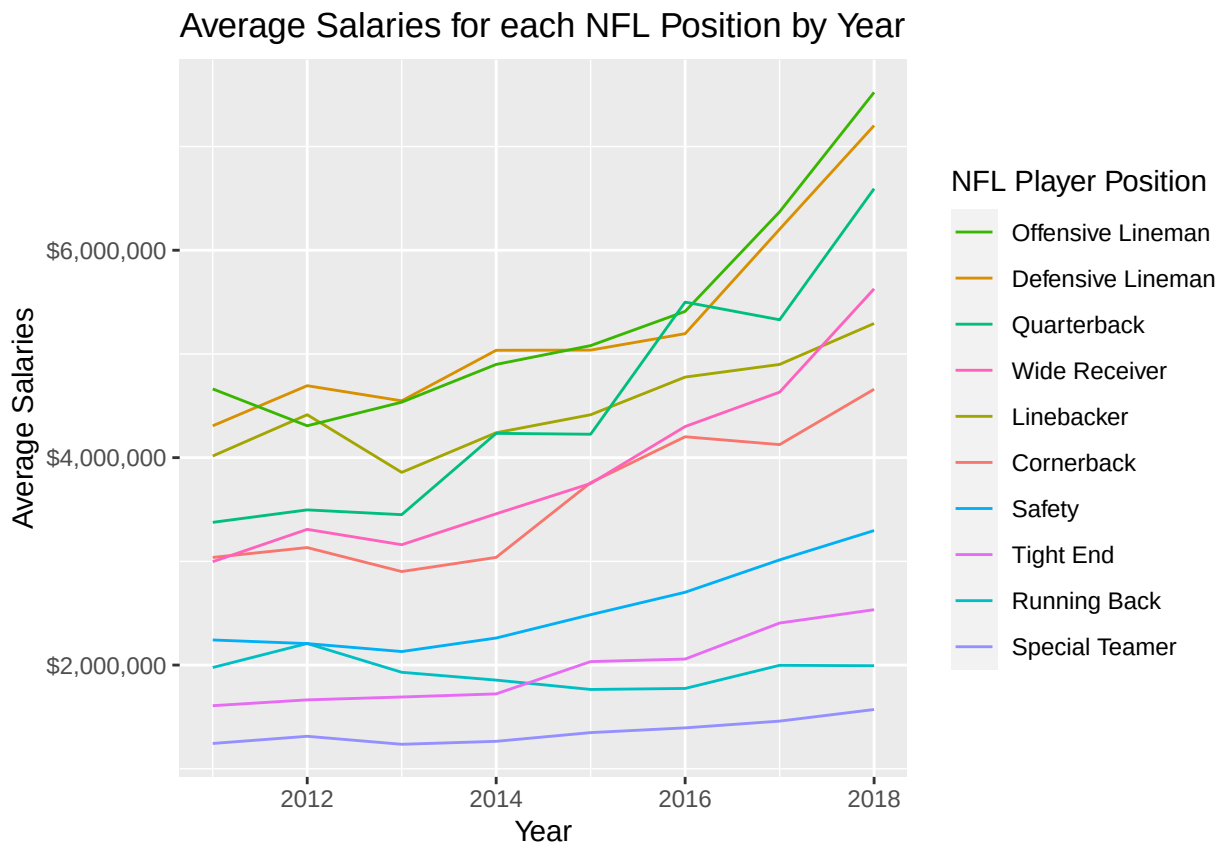
```
glimpse(avg_pos_sal)
```

```
## Rows: 80
## Columns: 3
## Groups: position [10]
## $ position     <chr> "Cornerback", "Cornerback", "Cornerback", "Cornerback", "~
## $ year         <int> 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2011, 201~
## $ avg_salaries <dbl> 3037766, 3132916, 2901798, 3038278, 3758543, 4201470, 412~
```

Question 5

Lastly, let's make a linegraph that traces the evolution of each position's average salary across the years.

```
ggplot(avg_pos_sal, aes(x = year, y = avg_salaries,
                        col = position)) +
  geom_line() +
  scale_y_continuous(name = "Average Salaries", labels = dollar) +
  labs(x = "Year", title = "Average Salaries for each NFL Position by Year",
       color = "NFL Player Position") +
  scale_color_discrete(
    breaks = c("Offensive.Lineman",
      "Defensive.Lineman",
      "Quarterback",
      "Wide.Receiver",
      "Linebacker",
      "Cornerback",
      "Safety",
      "Tight.End",
      "Running.Back",
      "Special.Teamer"),
    labels = c(
      "Offensive Lineman",
      "Defensive Lineman",
      "Quarterback",
      "Wide Receiver",
      "Linebacker",
      "Cornerback",
      "Safety",

      "Tight End",
      "Running Back",
      "Special Teamer"))
```



Describe at least two trends that are apparent to you.

1. Linemen positions have consistently made the most each year.
2. Overtime, all positions have had a salary increase, however, some (Safety, Running Back, Tight End, Special Teamer) have had smaller increases than others (Offensive Lineman, Defensive Lineman, Quarterback, Wide Reciever, Linebacker, Cornerback).