Communicating Variation Part 2

Dr Austin R Brown

Kennesaw State University

Introduction

- In the previous lecture, we learned how to use and modify time series plots in order to communicate variation over time.
  - Strikeouts per 9 innings pitched per season, in our example.
- But what is a primary limitation of this plot?
- With this plot, we can see the variation over time. But what we cannot observe is the within year variation.
  - And consequently, we can't see how the within year variation changes over time.
- This may be useful to know! As previously mentioned, this type of information is very commonly used in visualizations of financial or stock data.
- For us to include this information in this given plot, we have a few different strategies we can employ.

Communicating Variation: Time Series Plot with Standard Errors

- First, we can plot not just the mean value, but also the standard errors of the means, which is method by which variation can be quantified.

- To do this, we will reaggregate our data now using the `rstatix` package in order to obtain the standard errors of the means.

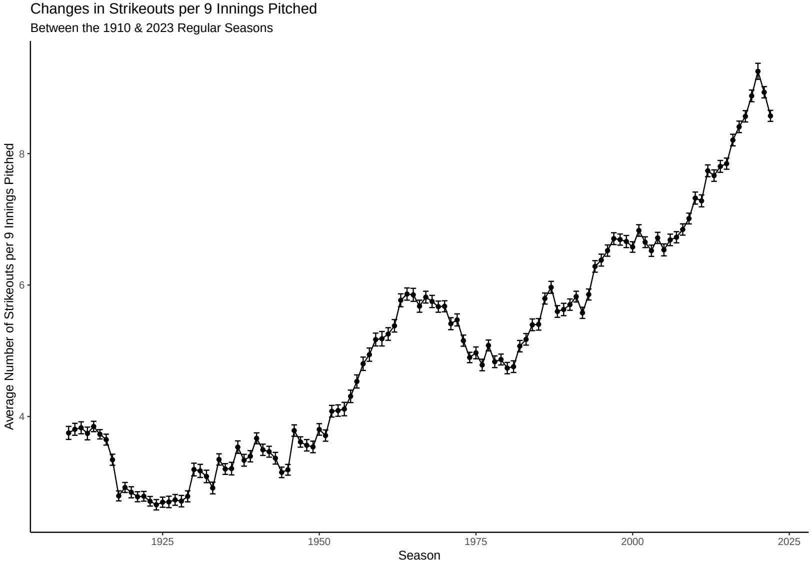
Communicating Variation: Time Series Plot with Standard Errors

```
library(rstatix)
so91 <- Lahman::Pitching |>
  filter(between(yearID,1910,2023) & (IPouts/3) >= 15) |>
  mutate(SO9 = SO/(IPouts/3)*9) |>
  group_by(yearID) |>
  get_summary_stats(SO9,type='full')
```

Communicating Variation: Time Series Plot with Standard Errors

```r
so91 |>
  ggplot(aes(x=yearID,y=mean)) +
  geom_point() +
  geom_errorbar(aes(ymin=mean-se,ymax=mean+se)) +
  geom_line() +
  labs(x="Season",
       y="Average Number of Strikeouts per 9 Innings Pitched",
       title="Changes in Strikeouts per 9 Innings Pitched",
       subtitle="Between the 1910 & 2023 Regular Seasons") +
  theme_classic()
```

Communicating Variation: Time Series Plot with Standard Errors



Communicating Variation: Time Series Plot with Standard Errors

- What's the problem with this graph?

- Primarily, because we have so many seasons, and so many pitchers within a given season, the standard errors are small. Recall, the standard error of the sample mean is:

$$SE[\bar{x}] = \frac{s}{\sqrt{n}}$$

- So as the sample size, n, grows, the standard errors get small. And as you can see in the `so91` dataframe, every season, hundreds of pitchers pitch.

Communicating Variation: Time Series Plot with Boxplots

- To get around this issue, it may be preferable to use boxplots rather than standard errors to represent the within year variation.

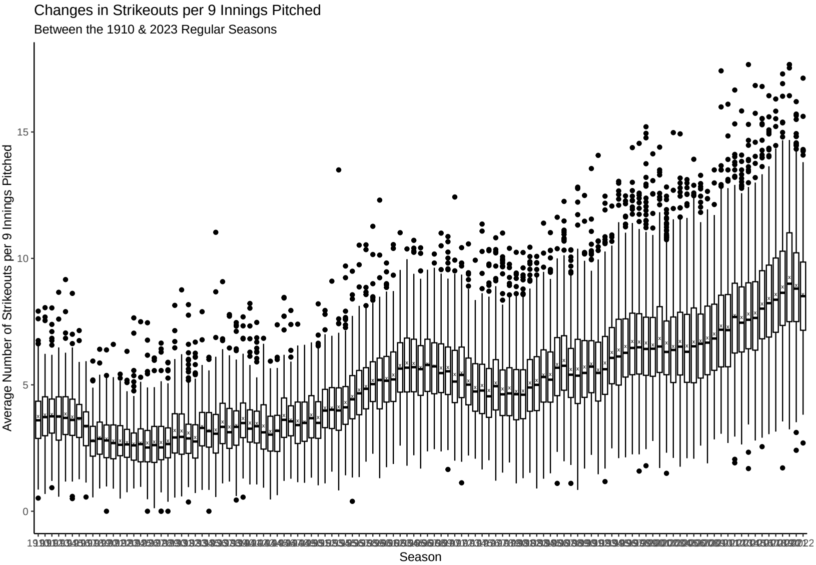
- However, remember that to visualize boxplots, the data must be in its raw form. Both of our datasets are aggregated data:

```
so92 <- Lahman::Pitching |>
  filter(between(yearID,1910,2023) & (IPouts/3) >= 15) |>
  mutate(SO9 = SO/(IPouts/3)*9)
```

Communicating Variation: Time Series Plot with Boxplots

```r
so92 |>
  ggplot(aes(x=factor(yearID),y=SO9)) +
  geom_boxplot(fill='white',color='black') +
  geom_point(data=so91,
             aes(x=factor(yearID),
                 y=mean),shape='x') +
  labs(x="Season",
       y="Average Number of Strikeouts per 9 Innings Pitched",
       title="Changes in Strikeouts per 9 Innings Pitched",
       subtitle="Between the 1910 & 2023 Regular Seasons") +
  theme_classic()
```

Communicating Variation: Time Series Plot with Boxplots



Communicating Variation: Time Series Plot with Boxplots

- Okay we're getting closer! Here, we can see that not only is the mean SO/9 increasing over time, but so too is the variability from year to year.

- We can see this since the difference between the top of each box and the bottom of each box, the 75th and 25th percentiles, respectively, tends to widen as we move from left to right across the graphic.

- But notice what's going on the x-axis. The tick mark labels are all overlapping to the point where they are completely illegible.

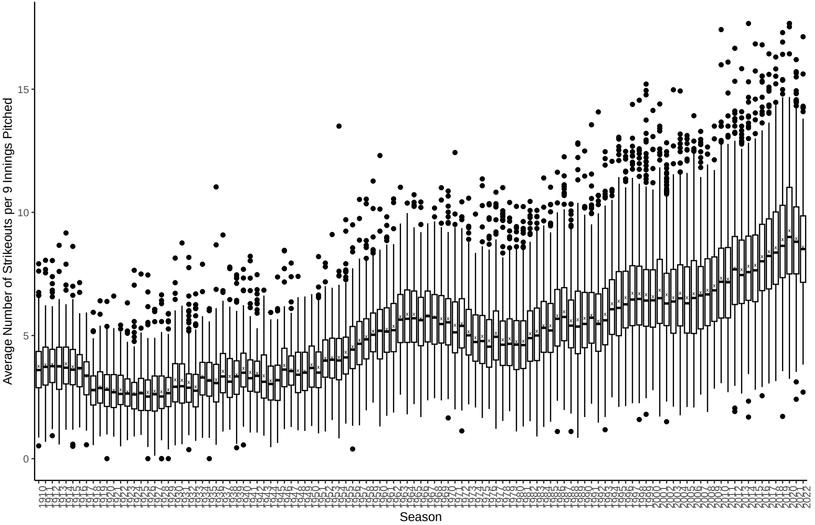
- Fortunately, we have a few tricks to modify the tick marks on either the x or y axis.

Communicating Variation: Time Series Plot with Boxplots

- One trick people sometimes use is changing the orientation or angle of the tick marks:

```
so92 |>
  ggplot(aes(x=factor(yearID),y=SO9)) +
  geom_boxplot(fill='white',color='black') +
  geom_point(data=so91,
             aes(x=factor(yearID),
                 y=mean),shape='x') +
  labs(x="Season",
       y="Average Number of Strikeouts per 9 Innings Pitched",
       title="Changes in Strikeouts per 9 Innings Pitched",
       subtitle="Between the 1910 & 2023 Regular Seasons") +
  theme_classic() +
  theme(axis.text.x=element_text(angle=90))
```

Communicating Variation: Time Series Plot with Boxplots



Changes in Strikeouts per 9 Innings Pitched

Between the 1910 & 2023 Regular Seasons

Communicating Variation: Time Series Plot with Boxplots

- In this case, that was helpful to eliminate the complete overlap of the labels, but now the problem is that there are too many labels.
    - It is still difficult to make out the individual digits for each year.

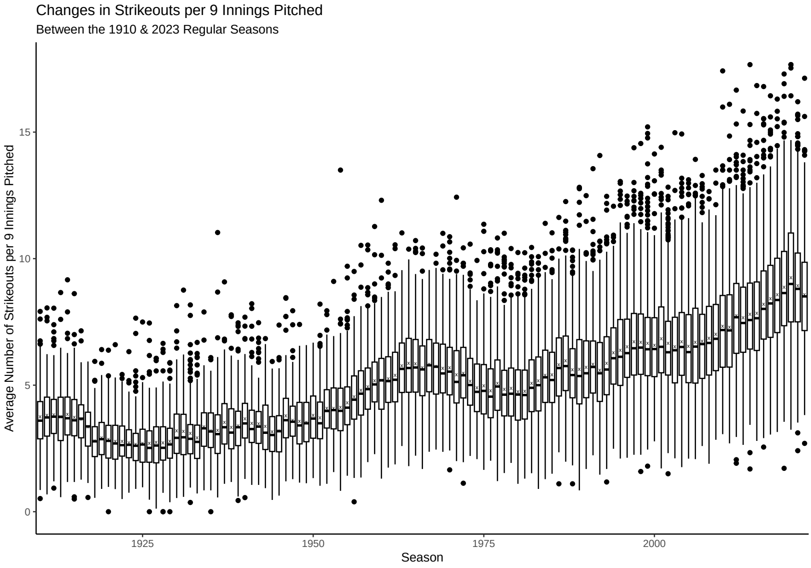
- What if we instead removed some of the labels to have the same number as we had with the regular time series plot (i.e., 1925, 1950, etc)

- Let's see how that can be achieved!

Communicating Variation: Time Series Plot with Boxplots

```r
so92 |>
  ggplot(aes(x=factor(yearID),y=SO9)) +
  geom_boxplot(fill='white',color='black') +
  geom_point(data=so91,
             aes(x=factor(yearID),
                 y=mean),shape='x') +
  labs(x="Season",
       y="Average Number of Strikeouts per 9 Innings Pitched",
       title="Changes in Strikeouts per 9 Innings Pitched",
       subtitle="Between the 1910 & 2023 Regular Seasons") +
  theme_classic() +
  scale_x_discrete(breaks=seq(1925,2020,by=25))
```

Communicating Variation: Time Series Plot with Boxplots


Changes in Strikeouts per 9 Innings Pitched
Between the 1910 & 2023 Regular Seasons
Average Number of Strikeouts per 9 Innings Pitched
Season