PSC4375: Summarizing bivariate relationships: cross-tabs, scatterplots, and correlation

Week 4: Lecture 8

Prof. Weldzius

Villanova University

Slides Updated: 2025-01-25

Effect of assasination attempts

```
library(tidyverse)
data(leaders, package = "qss")
head(leaders[,1:7])
```

```
##   year     country       leadername age politybefore
## 1 1929 Afghanistan Habibullah Ghazi  39            -6
## 2 1933 Afghanistan       Nadir Shah  53            -6
## 3 1934 Afghanistan      Hashim Khan  50            -6
## 4 1924     Albania             Zogu  29             0
## 5 1931     Albania             Zogu  36            -9
## 6 1968     Algeria      Boumedienne  41            -9
##   polityafter interwarbefore
## 1   -6.000000              0
## 2   -7.333333              0
## 3   -8.000000              0
## 4   -9.000000              0
```

Contingency tables

- With two categorical variables, we can create **contingency tables**
  - Also known as **cross-tabs**
  - Rows are the values of one variable, columns the other

```
leaders %>%
  group_by(civilwarbefore,civilwarafter) %>%
  count() %>%
  spread(civilwarafter, n)
```

```
## # A tibble: 2 x 3
## # Groups:   civilwarbefore [2]
##   civilwarbefore   `0`   `1`
##            <int> <int> <int>
## 1              0   177    19
## 2              1    27    27
```

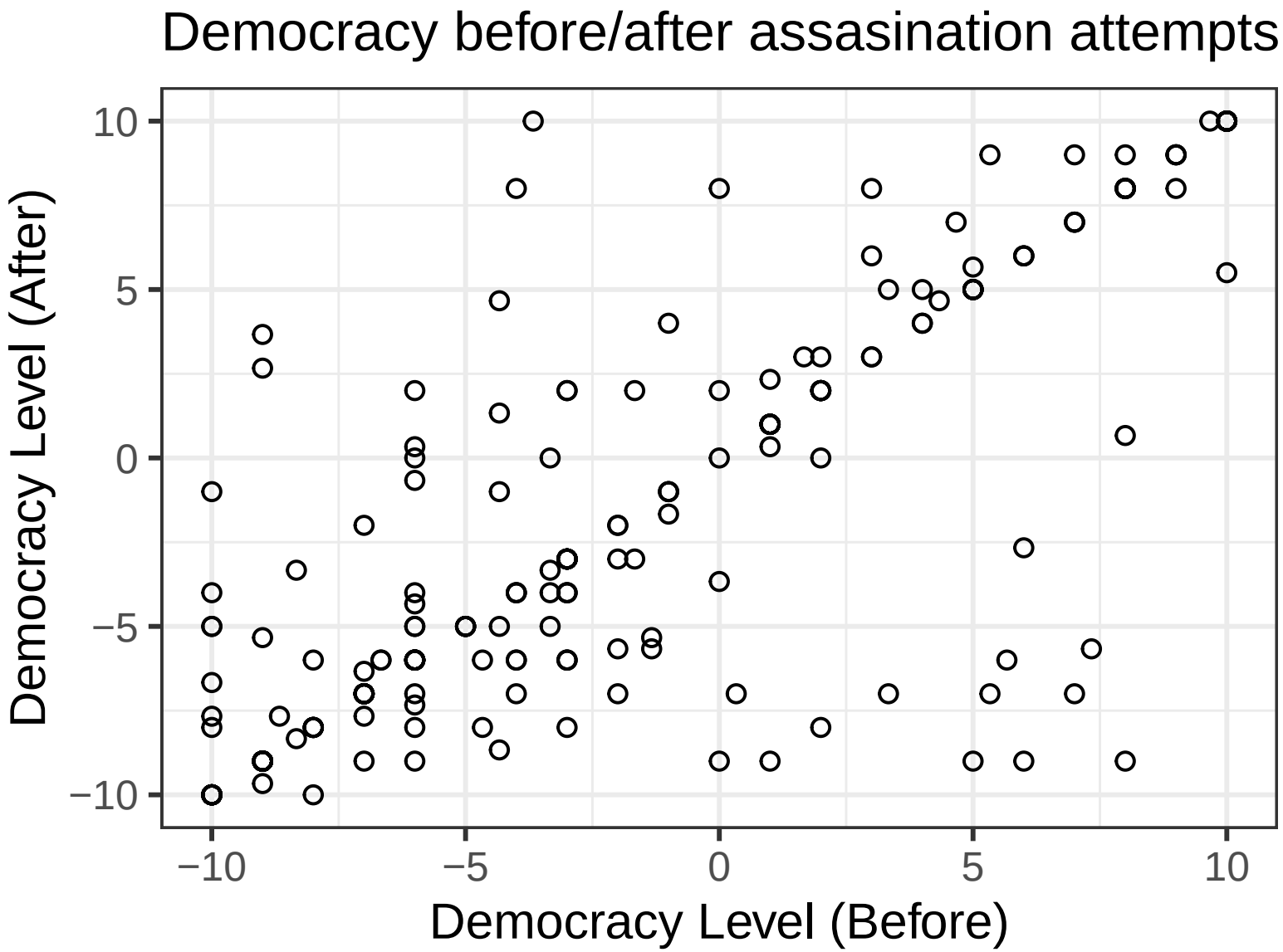
- Quick summary how the two variables "go together"

Cross-tabs with proportions

```
leaders %>%
  group_by(civilwarbefore,civilwarafter) %>%
  count() %>%
  ungroup() %>%
  mutate(prop = n/ sum(n)) %>%
  select(-n) %>%
  spread(civilwarafter, prop, drop = T)
```

```
## # A tibble: 2 x 3
##   civilwarbefore   `0`   `1`
##            <int> <dbl> <dbl>
## 1              0 0.708 0.076
## 2              1 0.108 0.108
```

Scatterplot



Scatterplot

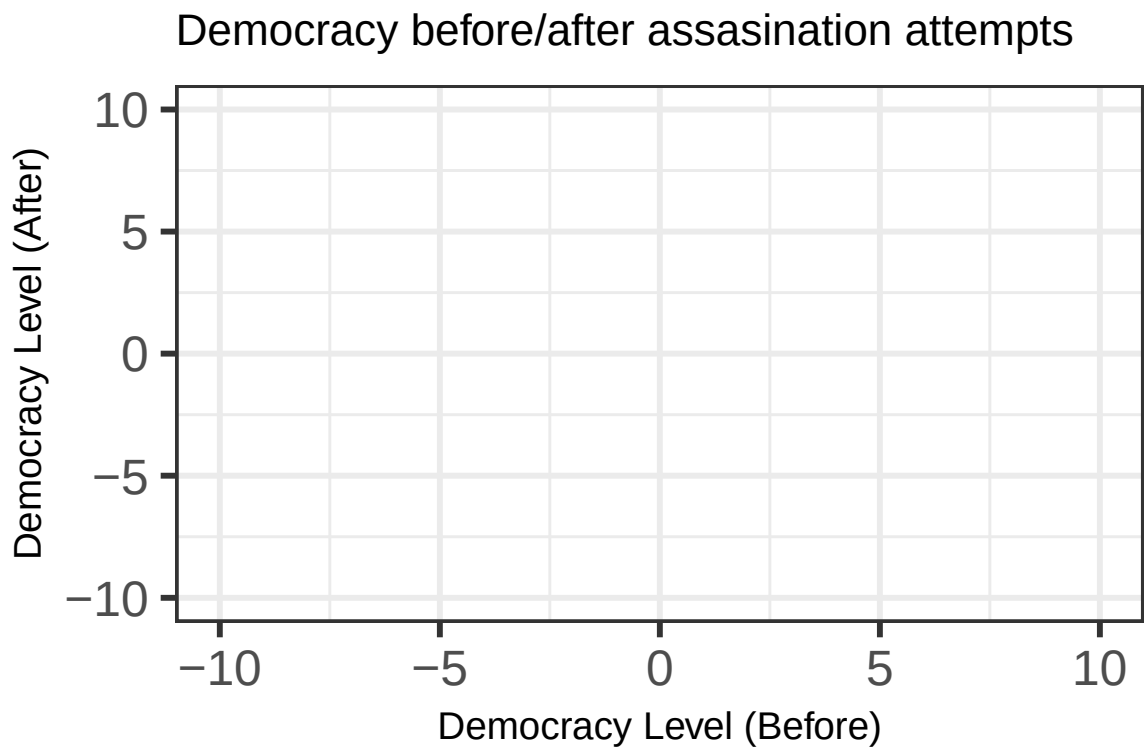
- Each point on the scatterplot (x_i, y_i)
- Use `geom_point()` function in ggplot

```
leaders %>%
  ggplot(aes(x = politybefore, y = polityafter)) +
  geom_point(shape = 21) +
  labs(title = "Democracy before/after assasination attempts",
       x = "Democracy Level (Before)",
       y = "Democracy Level (After)") +
  theme_bw() +
  theme(plot.title = element_text(size=12))
```

Scatterplot

```
leaders[1, c("politybefore","polityafter")]
```

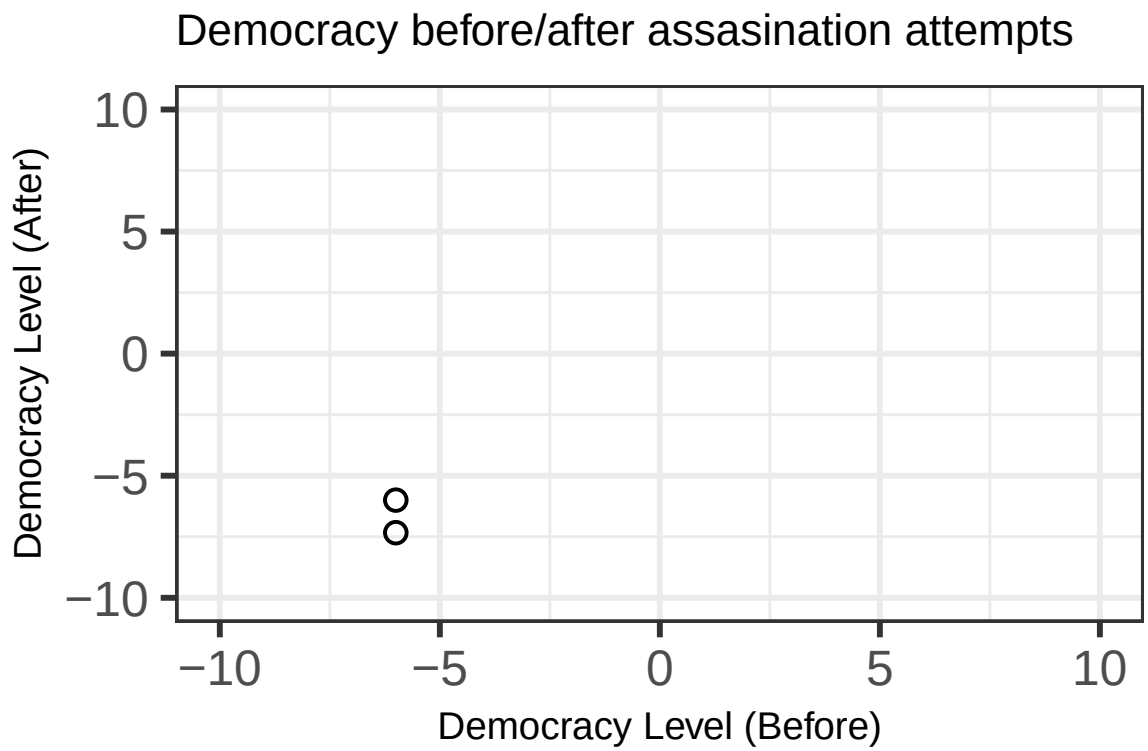
```
##   politybefore polityafter
## 1           -6          -6
```



Scatterplot

```
leaders[2, c("politybefore","polityafter")]
```

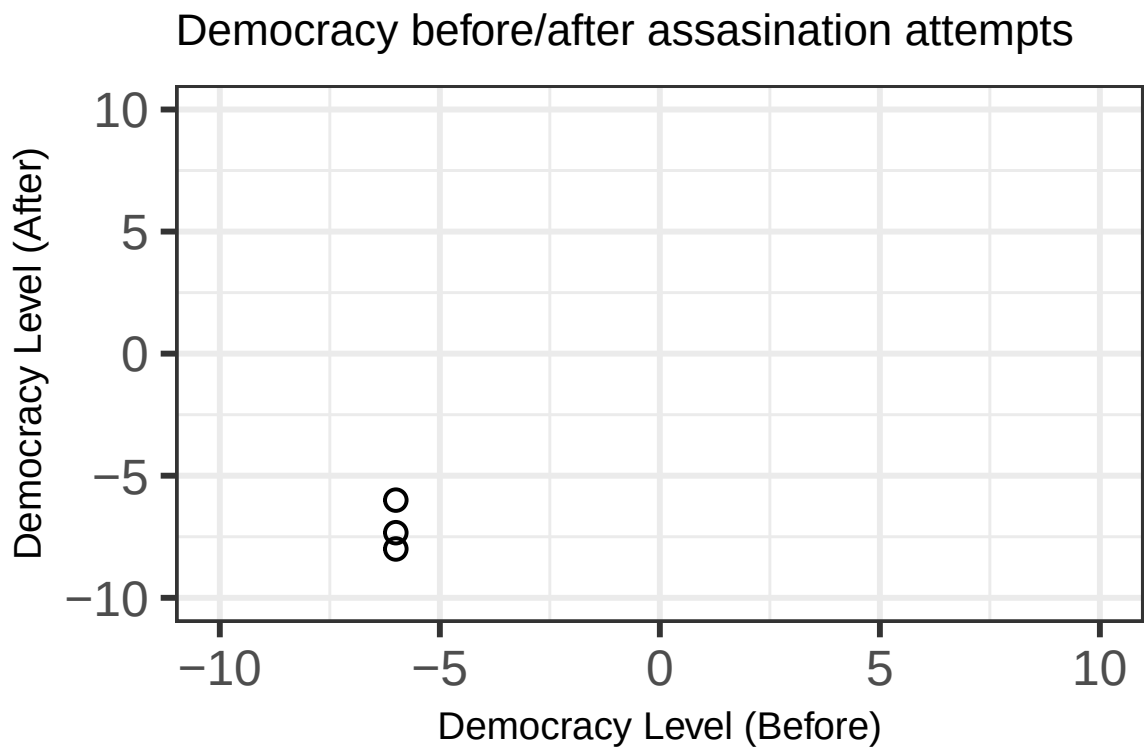
```
##   politybefore polityafter
## 2           -6   -7.333333
```



Scatterplot

```
leaders[3, c("politybefore","polityafter")]
```

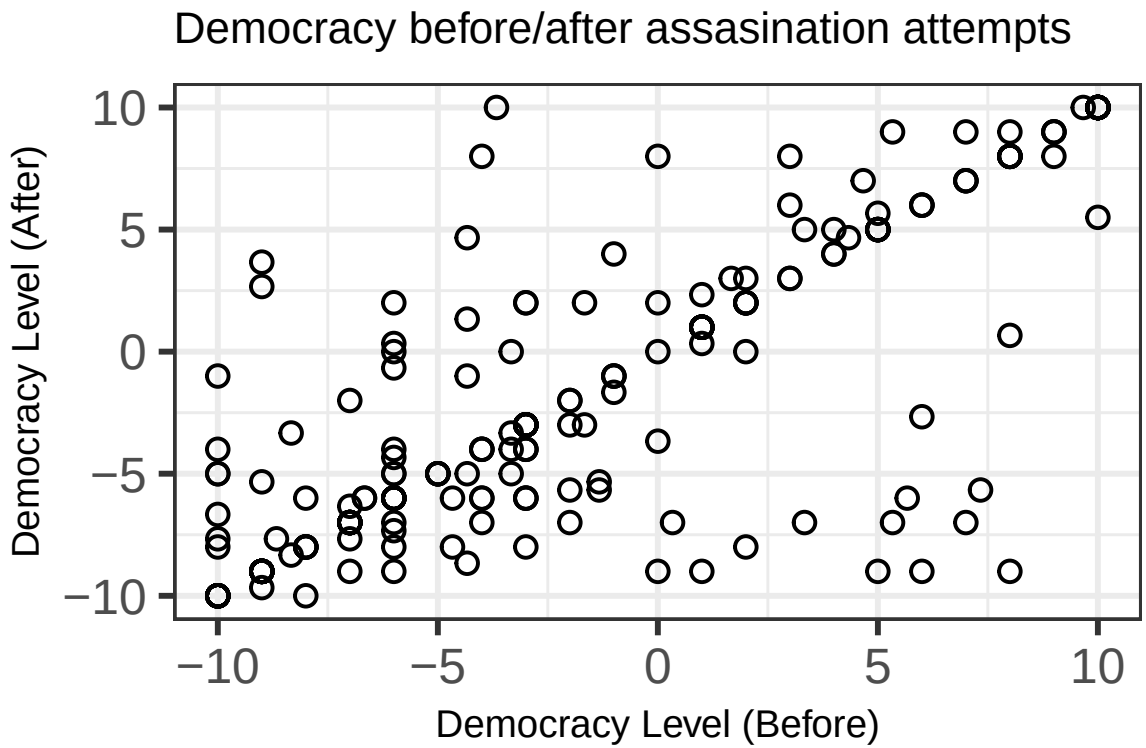
```
##   politybefore polityafter
## 3           -6          -8
```



Scatterplot

```
leaders[3, c("politybefore","polityafter")]
```

```
##   politybefore polityafter
## 3           -6          -8
```



How big is big?

- Would be nice to have a standard summary of how similar variabls are
  - Problem: variables on different scales!
  - Needs a way to put any variable on common units
- **z-score** to the rescue!

$$\text{z-score of } x_i = \frac{x_i - \text{mean of } x}{\text{standard deviation of } x}$$

- Crucial property: z-scores don't depend on units

$$\text{z-score of } (ax_i + b) = \text{z-score of } x_i$$

Correlation

- How do variables move together on average?
- When x_i is big, what is y_i likely to be?
  - Positive correlation: when x_i is big, y_i is also big
  - Negative correlation: when x_i is big, y_i is small
  - High magnitude of correlation: data cluster tightly around a line
- The technical definition of the **correlation coefficient**:

$$\frac{1}{n-1}\sum_{i=1}^{n}[(\text{z-score for } x_i) * (\text{z-score for } y_i)]$$

Correlation intuition: