

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

## **Week 4: Lecture 8**

Prof. Weldzius

Villanova University

Slides Updated: 2025-02-12

# Effect of assassination 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
## 5    -9.000000             0
## 6    -9.000000             0
```

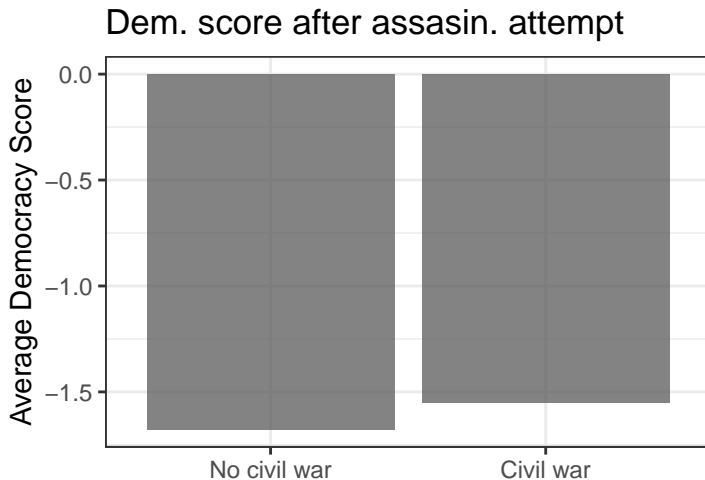
## Before we begin with lesson, Pset 2 help

```
PolityAfter <- leaders %>%  
  group_by(civilwarbefore) %>%  
  summarize(polityafter_mean = mean(polityafter))
```

```
PolityAfterPlot <- PolityAfter %>%  
  ggplot(aes(x=as.factor(civilwarbefore), y = polityafter_mean)) +  
  geom_bar(stat = "identity", alpha=0.75) +  
  scale_x_discrete(labels = c("No civil war", "Civil war")) +  
  labs(title = "Dem. score after assassin. attempt",  
       y = "Average Democracy Score", x = "") +  
  theme_bw()
```

# Before we begin with lesson, Pset 2 help

PolityAfterPlot



# More Pset 2 help!

Question 5 update:

- Given that the number of children might be a confounder for the relationship between number of girls and voting, let's estimate the effects using statistical control for the number of children using the following steps:
  - Create one subset of the data called `girls_123` that restricts to judges with one, two or three children and have at least one girl.
  - Create another subset of the data called `nogirls_123` that restricts to judges with one, two or three children and have no girls.
  - Calculate the mean of `progressive_vote` within levels of the numbers of kids (`num_kids`) for each of these subsets and save these vectors as `girls_vote_by_nkids` and `nogirls_vote_by_nkids`.
  - Use `inner_join` to combine the two data subsets then estimate the average treatment effect within levels, saving this vector as `ate_nkids`.

# More Pset 2 help!

- Use `inner_join` to combine the two data subsets...

```
PolityAfter <- leaders %>%  
  group_by(civilwarbefore) %>%  
  summarize(polityafter_mean = mean(polityafter))  
  
PolityBefore <- leaders %>%  
  group_by(civilwarbefore) %>%  
  summarize(politybefore_mean = mean(politybefore))  
  
PolityCombine <- inner_join(PolityAfter, PolityBefore)  
PolityCombine
```

```
## # A tibble: 2 x 3  
##   civilwarbefore polityafter_mean politybefore_mean  
##           <int>           <dbl>           <dbl>  
## 1             0           -1.68           -1.52  
## 2             1           -1.55           -1.53
```

# 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
```

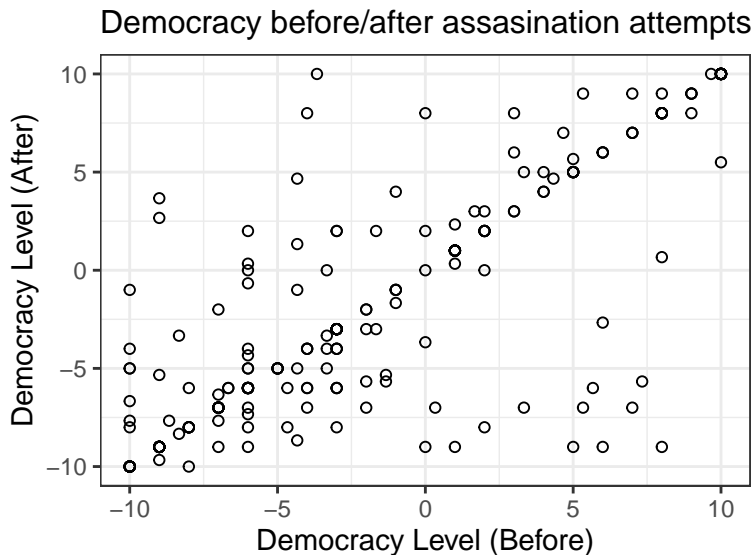


# Cross-tabs with proportions (by row)

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

```
## # A tibble: 2 x 3  
## # Groups:   civilwarbefore [2]  
##   civilwarbefore '0'    '1'  
##           <int> <dbl> <dbl>  
## 1             0 0.903 0.0969  
## 2             1 0.5   0.5
```

# Scatterplot



# Scatterplot

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

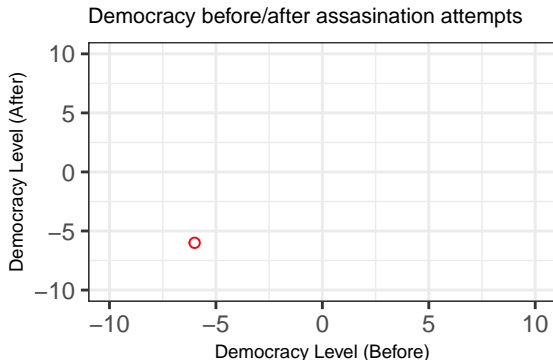
```
leaders %>%  
  ggplot(aes(x = politybefore, y = polityafter)) +  
  geom_point(shape = 21) +  
  labs(title = "Democracy before/after assassination attempts",  
        x = "Democracy Level (Before)",  
        y = "Democracy Level (After)") +  
  theme_bw() +  
  theme(plot.title = element_text(size=12))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

# Scatterplot

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

```
##      politybefore polityafter
```

```
## 1             -6             -6
```

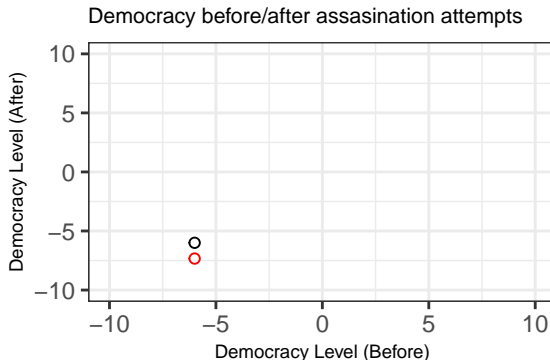


# Scatterplot

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

```
##      politybefore polityafter
```

```
## 2                -6      -7.333333
```

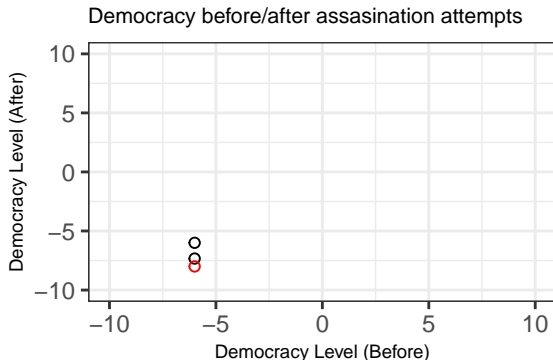


# Scatterplot

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

```
##      politybefore polityafter
```

```
## 3              -6           -8
```

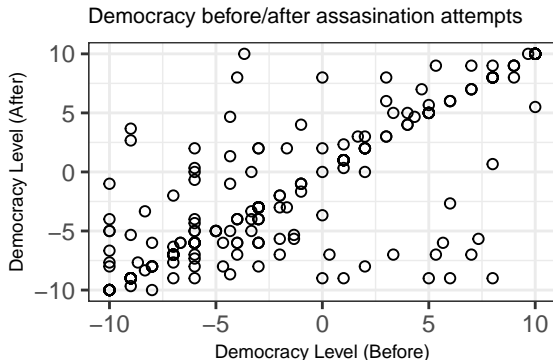


# Scatterplot

```
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 variables 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

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

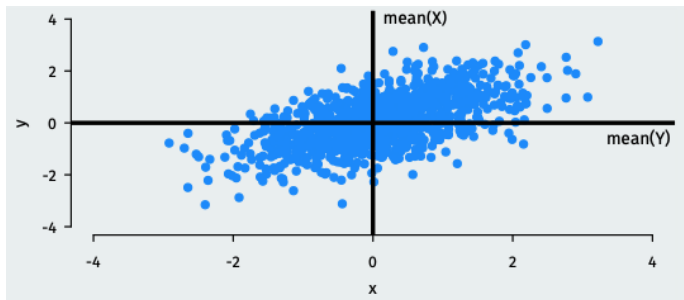


# Correlation

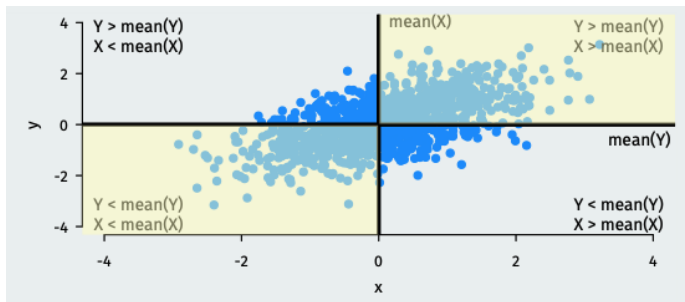
- How do variables move together on average?
- When  $x_i$  is big, what is  $y_i$  likely to be?
  - Positive correlation: when  $x_i$  is big,  $y_i$  is also big
  - Negative correlation: when  $x_i$  is big,  $y_i$  is small
  - High magnitude of correlation: data cluster tightly around a line
-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

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

# Correlation intuition:

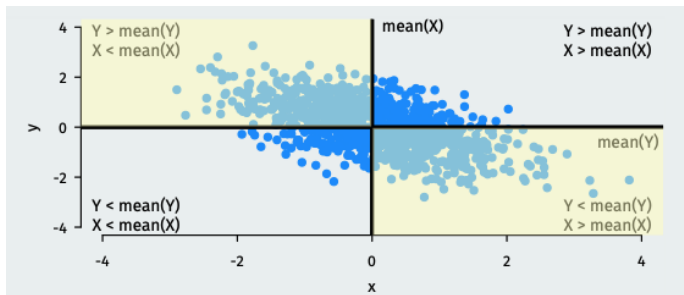


# Correlation intuition:



- Large values of  $X$  tend to occur with large values of  $Y$ 
  - $(\text{z-score for } x_i) \times (\text{z-score for } y_1) = (\text{pos. num.}) \times (\text{pos. num.}) = +$
- Small values of  $X$  tend to occur with small values of  $Y$ 
  - $(\text{z-score for } x_i) \times (\text{z-score for } y_1) = (\text{neg. num.}) \times (\text{neg. num.}) = +$
- If these dominate  $\rightsquigarrow$  positive correlation

# Correlation intuition:



- Large values of  $X$  tend to occur with small values of  $Y$ 
  - $(\text{z-score for } x_i) \times (\text{z-score for } y_1) = (\text{pos. num.}) \times (\text{neg. num.}) = -$
- Small values of  $X$  tend to occur with large values of  $Y$ 
  - $(\text{z-score for } x_i) \times (\text{z-score for } y_1) = (\text{neg. num.}) \times (\text{pos. num.}) = -$
- If these dominate  $\rightsquigarrow$  negative correlation

# Properties of correlation coefficient

- Correlation measures **linear** association.
- Interpretation:
  - Correlation is between -1 and 1
  - Correlation of 0 means no linear association
  - Positive correlations  $\rightsquigarrow$  positive associations
  - Negative correlations  $\rightsquigarrow$  negative associations
  - Closer to -1 or 1 means stronger association
- Order doesn't matter:  $\text{cor}(x,y) = \text{cor}(y,x)$
- Not affected by changes of scale:
  - $\text{cor}(x,y) = \text{cor}(ax+b, cy+d)$
  - Celsius vs. Fahrenheit; dollars vs. pesos; cm vs. in.

# Correlation in R

- Use the `cor()` function

```
leaders %>%  
  select(politybefore, polityafter) %>%  
  cor()
```

```
##               politybefore polityafter  
## politybefore    1.0000000    0.8283237  
## polityafter     0.8283237    1.0000000
```

-Very highly correlated!