

Poli 5D Social Science Data Analytics

R: More on Matrix, and Visualization

Shane Xinyang Xuan
ShaneXuan.com

February 26, 2017

Contact Information

Shane Xinyang Xuan

xxuan@ucsd.edu

The teaching staff is a team!

Professor Roberts	M	1600-1800 (SSB 299)
Jason Bigenho	Th	1000-1200 (Econ 116)
Shane Xuan	M	1100-1150 (SSB 332)
	Th	1200-1250 (SSB 332)

Supplemental Materials

UCLA STATA starter kit

<http://www.ats.ucla.edu/stat/stata/sk/>

Princeton data analysis

<http://dss.princeton.edu/training/>

Some quick notes before we start today's section:

- ▶ Make sure that you pass around the attendance sheet

Some quick notes before we start today's section:

- ▶ Make sure that you pass around the attendance sheet
- ▶ We will talk about **matrix** (Imai 2.2) and **visualization** (Imai 3.3, 3.6) in **R** today

Some quick notes before we start today's section:

- ▶ Make sure that you pass around the attendance sheet
- ▶ We will talk about **matrix** (Imai 2.2) and **visualization** (Imai 3.3, 3.6) in **R** today
- ▶ I will walk around answering questions if we have time left

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)
- ▶ We can subset vectors and data frames using **indexing** and **logical values**

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)
- ▶ We can subset vectors and data frames using **indexing** and **logical values**
- ▶ Suppose we are working with a data frame called *myData*

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)
- ▶ We can subset vectors and data frames using **indexing** and **logical values**
- ▶ Suppose we are working with a data frame called *myData*
- ▶ **Scenario**: I want to calculate the mean of variable **X** given variable **Y** is taking a value of 3

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)
- ▶ We can subset vectors and data frames using **indexing** and **logical values**
- ▶ Suppose we are working with a data frame called *myData*
- ▶ **Scenario**: I want to calculate the mean of variable **X** given variable **Y** is taking a value of 3
`mean(myData$X[myData$Y == 3])`

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)
- ▶ We can subset vectors and data frames using **indexing** and **logical values**
- ▶ Suppose we are working with a data frame called *myData*
- ▶ **Scenario**: I want to calculate the mean of variable **X** given variable **Y** is taking a value of 3
`mean(myData$X[myData$Y == 3])`
- ▶ **Scenario**: I want to calculate the mean of variable **X** given variable **Z** is taking a value of “apple”

Subsetting Data

- ▶ Takes time to get intuition, and make sure you read [Imai 2.2](#)
- ▶ We can subset vectors and data frames using **indexing** and **logical values**
- ▶ Suppose we are working with a data frame called *myData*
- ▶ **Scenario**: I want to calculate the mean of variable **X** given variable **Y** is taking a value of 3
`mean(myData$X[myData$Y == 3])`
- ▶ **Scenario**: I want to calculate the mean of variable **X** given variable **Z** is taking a value of "apple"
`mean(myData$X[myData$Z == "apple"])`

Subsetting Data: More on Logics

- ▶ We can use `ifelse()` function to construct variables that might be helpful for subsetting data

Subsetting Data: More on Logics

- ▶ We can use `ifelse()` function to construct variables that might be helpful for subsetting data
- ▶ **Scenario:** I want to generate a variable `T` that takes a value of 1 if variable `Q` takes a value of 3 and variable `P` takes a value of "apple", and 0 otherwise

Subsetting Data: More on Logics

- ▶ We can use `ifelse()` function to construct variables that might be helpful for subsetting data
- ▶ **Scenario:** I want to generate a variable `T` that takes a value of 1 if variable `Q` takes a value of 3 and variable `P` takes a value of "apple", and 0 otherwise

```
myData$T ← ifelse (myData$Q==3 & myData$P=="apple", 1, 0)
```

Subsetting Data: More on Logics

- ▶ We can use `ifelse()` function to construct variables that might be helpful for subsetting data
- ▶ **Scenario:** I want to generate a variable `T` that takes a value of 1 if variable `Q` takes a value of 3 and variable `P` takes a value of "apple", and 0 otherwise

```
myData$T ← ifelse (myData$Q==3 & myData$P=="apple", 1, 0)
```

- ▶ This might not look intuitive to you right now – any questions that I can answer?

Visualization in R

Good graphs **take time**, and make sure you read the book

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot

- 1.) Use `table()` to generate a table

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot

- 1.) Use `table()` to generate a table
- 2.) Plot this table using `barplot()`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

- ▶ `abline(h=x)`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

- ▶ `abline(h=x)`
- ▶ `abline(v=x)`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

- ▶ `abline(h=x)`
- ▶ `abline(v=x)`
- ▶ To add a **line/point/text**: `lines()`; `points()`; `text()`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

- ▶ `abline(h=x)`
- ▶ `abline(v=x)`
- ▶ To add a **line/point/text**: `lines()`; `points()`; `text()`
- ▶ `col`; `cex`; `lty`; `lwd`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

- ▶ `abline(h=x)`
- ▶ `abline(v=x)`
- ▶ To add a **line/point/text**: `lines()`; `points()`; `text()`
- ▶ `col`; `cex`; `lty`; `lwd`
- ▶ `main`; `ylab`; `xlab`; `ylim`; `xlim`

Visualization in R

Good graphs **take time**, and make sure you read the book

- ▶ Barplot
 - 1.) Use `table()` to generate a table
 - 2.) Plot this table using `barplot()`
- ▶ Histogram: `hist(x, ...)`
- ▶ Boxplot: `boxplot(x, ...)`
- ▶ Scatterplot: `plot(x, y, ...)`

Common arguments:

- ▶ `abline(h=x)`
- ▶ `abline(v=x)`
- ▶ To add a **line/point/text**: `lines()`; `points()`; `text()`
- ▶ `col`; `cex`; `lty`; `lwd`
- ▶ `main`; `ylab`; `xlab`; `ylim`; `xlim`
- ▶ **Scenario**: I want x rows and y columns of plots
`par(mfrow=c(x,y))`

- ▶ I will be walking around answering questions
- ▶ Office hours: M (1100-1150) and Th (1200-1250) in [SSB 332](#)