CSSS/POLS 510 Maximum Likelihood Estimation: Lab 1

Logistics and R Review

Kenya Amano

2020-10-3

About TA

Logistics

1. **Lab Sessions**: Fri, 3:30 - 5:20pm via Zoom

Logistics

1. **Lab Sessions**: Fri, 3:30 - 5:20pm via Zoom

- Covers application of material from lecture using examples; clarification and extention of lecture material; Q & A for homeworks and lectures

Logistics

1. **Lab Sessions**: Fri, 3:30 - 5:20pm via Zoom

- Covers application of material from lecture using examples; clarification and extention of lecture material; Q & A for homeworks and lectures

- Materials will be available on the **course website**

Logistics

1. **Lab Sessions**: Fri, 3:30 - 5:20pm via Zoom

- Covers application of material from lecture using examples; clarification and extention of lecture material; Q & A for homeworks and lectures

- Materials will be available on the **course website**

2. **Office Hours**: Thursdays, 1:00 - 2:15 pm, or by appointment via Zoom

Logistics

1. **Lab Sessions**: Fri, 3:30 - 5:20pm via Zoom

- Covers application of material from lecture using examples; clarification and extention of lecture material; Q & A for homeworks and lectures

- Materials will be available on the **course website**

2. **Office Hours**: Thursdays, 1:00 - 2:15 pm, or by appointment via Zoom

- Available for trouble shooting and specific questions about homework and lecture materials

Logistics

1. **Lab Sessions**: Fri, 3:30 - 5:20pm via Zoom

- Covers application of material from lecture using examples; clarification and extention of lecture material; Q & A for homeworks and lectures

- Materials will be available on the **course website**

2. **Office Hours**: Thursdays, 1:00 - 2:15 pm, or by appointment via Zoom

- Available for trouble shooting and specific questions about homework and lecture materials
- Time is subject to change and, if it does, I will e-mail the list

Logistics (Cont.)

3. **Homework**: 5-6 due every 2 weeks or so

Logistics (Cont.)

3. **Homework**: 5-6 due every 2 weeks or so

- Must be typed up

Logistics (Cont.)

3. **Homework**: 5-6 due every 2 weeks or so

- Must be typed up
- Ideally, done using R or R Studio with write up in LaTeX

Logistics (Cont.)

3. **Homework**: 5-6 due every 2 weeks or so

- Must be typed up
- Ideally, done using R or R Studio with write up in LaTeX
- Using R Studio with R Markdown is an easy way to do this (Will work on this next week)

Logistics (Cont.)

3. **Homework**: 5-6 due every 2 weeks or so

- Must be typed up
- Ideally, done using R or R Studio with write up in LaTeX
- Using R Studio with R Markdown is an easy way to do this (Will work on this next week)
- We will use two of Chris's packages extensively: simcf and tile

Logistics - Goals

1. **Be-Able-Tos**: When this course is over, you should be able to do the following (and much more):

Logistics - Goals

1. **Be-Able-Tos**: When this course is over, you should be able to do the following (and much more):
   - Identify the proper distribution and model for your data (logistic, ordered, multinomial, count)

Logistics - Goals

1. **Be-Able-Tos**: When this course is over, you should be able to do the following (and much more):
   - Identify the proper distribution and model for your data (logistic, ordered, multinomial, count)
   - Run the model using both the glm function and "by hand" using optim, extract parameters of interest, and interpret these in probabilities

Logistics - Goals

1. **Be-Able-Tos**: When this course is over, you should be able to do the following (and much more):
   - Identify the proper distribution and model for your data (logistic, ordered, multinomial, count)
   - Run the model using both the glm function and "by hand" using optim, extract parameters of interest, and interpret these in probabilities
   - Compute predicted probabilities and use simulation to find the confidence intervals of $\hat{\pi}_i$ across counterfactuals values of covariates $\boldsymbol{x_i}$

Logistics - Goals

1. **Be-Able-Tos**: When this course is over, you should be able to do the following (and much more):
   - Identify the proper distribution and model for your data (logistic, ordered, multinomial, count)
   - Run the model using both the glm function and "by hand" using optim, extract parameters of interest, and interpret these in probabilities
   - Compute predicted probabilities and use simulation to find the confidence intervals of $\hat{\pi}_i$ across counterfactuals values of covariates $\boldsymbol{x_i}$
   - Use cross-validation to assess the predictive accuracy of several models and also compare these models across a variety of in-sample goodness of fit tests

Logistics - Goals

1. **Be-Able-Tos**: When this course is over, you should be able to do the following (and much more):
   - Identify the proper distribution and model for your data (logistic, ordered, multinomial, count)
   - Run the model using both the glm function and "by hand" using optim, extract parameters of interest, and interpret these in probabilities
   - Compute predicted probabilities and use simulation to find the confidence intervals of $\hat{\pi}_i$ across counterfactuals values of covariates $\boldsymbol{x_i}$
   - Use cross-validation to assess the predictive accuracy of several models and also compare these models across a variety of in-sample goodness of fit tests
   - Use one of several algorithms to impute missing data

Logistics - R

1. **The stuff in R**: For the homework assignments and project you will need to feel comfortable

Logistics - R

1. **The stuff in R**: For the homework assignments and project you will need to feel comfortable
   - importing (and exporting) data sets

Logistics - R

1. **The stuff in R**: For the homework assignments and project you will need to feel comfortable
   - importing (and exporting) data sets
   - tidying and transforming data

Logistics - R

1. **The stuff in R**: For the homework assignments and project you will need to feel comfortable
   - importing (and exporting) data sets
   - tidying and transforming data
   - analyzing data (conceptual part of the course)

Logistics - R

1. **The stuff in R**: For the homework assignments and project you will need to feel comfortable
   - importing (and exporting) data sets
   - tidying and transforming data
   - analyzing data (conceptual part of the course)
   - generating plots of your data and results

Logistics - R

1. **The stuff in R**: For the homework assignments and project you will need to feel comfortable
   - importing (and exporting) data sets
   - tidying and transforming data
   - analyzing data (conceptual part of the course)
   - generating plots of your data and results
   - writing basic functions and loops for repeated procedures

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Do not print unnecessary code and output. Learn how to use `results = "hide"` and `echo = TRUE` in R Markdown.

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Do not print unnecessary code and output. Learn how to use `results = "hide"` and `echo = TRUE` in R Markdown.
   - Name well

Logistics - `R`

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Do not print unnecessary code and output. Learn how to use `results = "hide"` and `echo = TRUE` in R Markdown.
   - Name well
     - functions vs. all other objects

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Do not print unnecessary code and output. Learn how to use `results = "hide"` and `echo = TRUE` in R Markdown.
   - Name well
     - functions vs. all other objects
     - readability is about consistency (dot.naming, CamelCaseNaming, pothole_naming)

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Do not print unnecessary code and output. Learn how to use `results = "hide"` and `echo = TRUE` in R Markdown.
   - Name well
     - functions vs. all other objects
     - readability is about consistency (dot.naming, CamelCaseNaming, pothole_naming)
     - short, clear, consistent – help future you (and present me)

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.

```r
rbinom(n = 1000, size = 30, prob = 0.49) # GOOD!

rbinom(1000, 30, 0.49) # LESS GOOD!
```

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Specify arguments fully, e.g.

```r
rbinom(n = 1000, size = 30, prob = 0.49) # GOOD!

rbinom(1000, 30, 0.49) # LESS GOOD!
```

Logistics - R

2. I have to read lots of your code. Please be considerate when writing code and submitting assignments.
   - Specify arguments fully, e.g.

```r
rbinom(n = 1000, size = 30, prob = 0.49) # GOOD!

rbinom(1000, 30, 0.49) # LESS GOOD!
```

   - See the Google R styleguide for an example

Logistics - R Useful resources

- R

Logistics - R Useful resources

- R
  - *R for Data Science* (Grolemund and Wickham 2016)

Logistics - R Useful resources

- R
  - *R for Data Science* (Grolemund and Wickham 2016)
  - *Quantitative Social Science : An Introduction* (Imai 2017)

Logistics - R Useful resources

- R
  - *R for Data Science* (Grolemund and Wickham 2016)
  - *Quantitative Social Science : An Introduction* (Imai 2017)
  - DataCamp: https://www.datacamp.com

Logistics - R Useful resources

- R
  - *R for Data Science* (Grolemund and Wickham 2016)
  - *Quantitative Social Science : An Introduction* (Imai 2017)
  - DataCamp: https://www.datacamp.com
  - R cheat sheets: https://rstudio.com/resources/cheatsheets/

Logistics - R Useful resources

- R
  - *R for Data Science* (Grolemund and Wickham 2016)
  - *Quantitative Social Science : An Introduction* (Imai 2017)
  - DataCamp: https://www.datacamp.com
  - R cheat sheets: https://rstudio.com/resources/cheatsheets/
- R Markdown

Logistics - R Useful resources

- R
  - *R for Data Science* (Grolemund and Wickham 2016)
  - *Quantitative Social Science : An Introduction* (Imai 2017)
  - DataCamp: https://www.datacamp.com
  - R cheat sheets: https://rstudio.com/resources/cheatsheets/
- R Markdown
  - *R Markdown: The Definitive Guide* (Xie, Allaire, and Grolemund 2019)

Logistics - R Useful resources

- Data visualization

Logistics - R Useful resources

- Data visualization
  - *Data Visualization: A Practical Introduction* (Healy 2018)

Logistics - R Useful resources

- Data visualization
  - *Data Visualization: A Practical Introduction* (Healy 2018)
  - *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures* (Wilke 2019)

Logistics - R Useful resources

- Data visualization
  - *Data Visualization: A Practical Introduction* (Healy 2018)
  - *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures* (Wilke 2019)
- Others

Logistics - R Useful resources

- Data visualization
  - *Data Visualization: A Practical Introduction* (Healy 2018)
  - *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures* (Wilke 2019)
- Others
  - Stack Overflow: https://stackoverflow.com

Logistics - R Useful resources

- Data visualization
  - *Data Visualization: A Practical Introduction* (Healy 2018)
  - *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures* (Wilke 2019)
- Others
  - Stack Overflow: https://stackoverflow.com
  - TidyTuesday Project: https://github.com/rfordatascience/tidytuesday

Logistics - Social Sciences & Computing

1. There are best practices for computing in the social sciences. You should aim for transparency and replicability in your work in general, and clarity and consistency in your code.

Logistics - Social Sciences & Computing

1. There are best practices for computing in the social sciences. You should aim for transparency and replicability in your work in general, and clarity and consistency in your code.
   - Best Practices (Wilson et al. 2014)

Logistics - Social Sciences & Computing

1. There are best practices for computing in the social sciences. You should aim for transparency and replicability in your work in general, and clarity and consistency in your code.
   - Best Practices (Wilson et al. 2014)
   - Good Enough (Wilson et al. 2017)

R refresher

1. Overview

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming
  - System-supplied or user-defined functionality as *functions*

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming
  - System-supplied or user-defined functionality as *functions*
  - Extended via *packages*

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming
  - System-supplied or user-defined functionality as *functions*
  - Extended via *packages*
- RStudio is an integrated development environment for R, which includes:

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming
  - System-supplied or user-defined functionality as *functions*
  - Extended via *packages*
- RStudio is an integrated development environment for R, which includes:
  - a console to run R code

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming
  - System-supplied or user-defined functionality as *functions*
  - Extended via *packages*
- RStudio is an integrated development environment for R, which includes:
  - a console to run R code
  - an editor to write code and text

R refresher

1. Overview

- R is a language and environment for statistical computing and graphics
  - *Object-oriented* style of programming
  - System-supplied or user-defined functionality as *functions*
  - Extended via *packages*
- RStudio is an integrated development environment for R, which includes:
  - a console to run R code
  - an editor to write code and text
  - tools for plotting, history, debugging and workspace management

R refresher

2. Data Types

R refresher

2. Data Types
   - character, numeric (integer or double), logical, complex

R refresher

2. Data Types
   - character, numeric (integer or double), logical, complex
   - data can also be missing

R refresher

2. Data Types
   - character, numeric (integer or double), logical, complex
   - data can also be missing
3. Data Structures

R refresher

2. Data Types
   - character, numeric (integer or double), logical, complex
   - data can also be missing
3. Data Structures

- Matrices vs. data frames

R refresher

2. Data Types
   - character, numeric (integer or double), logical, complex
   - data can also be missing
3. Data Structures

- Matrices vs. data frames
  - Matrices can only contain one **homogenous** type of vectors

R refresher

2. Data Types
   - character, numeric (integer or double), logical, complex
   - data can also be missing
3. Data Structures

- Matrices vs. data frames
  - Matrices can only contain one **homogenous** type of vectors
  - Data frames can contain **heterogeneous** types of vectors, and thus are more flexible

R refresher

3. Data Structure - Summary

	Homogeneous	Heterogeneous
1d	Atomic vector	List
2d	Matrix	Data frame
nd	Array	

R refresher

3. Data Structure - Summary

	Homogeneous	Heterogeneous
1d	Atomic vector	List
2d	Matrix	Data frame
nd	Array	

- For much more see here or here

R refresher

4. R as calculator

R refresher

4. R as calculator

- Standard mathematical operators (e.g. `+ - * / ^` etc.)

R refresher

4. R as calculator

- Standard mathematical operators (e.g. `+ - * / ^` etc.)
- Functions (e.g., `mean()`) take arguments (inputs)

R refresher

4. R as calculator

- Standard mathematical operators (e.g. `+ - * / ^` etc.)
- Functions (e.g., `mean()`) take arguments (inputs)
- Logical operators (e.g. `==, >, <, >=, <=, !=`) return `TRUE FALSE` or `NA`

R refresher

4. R as calculator

```
1 + 7
```

```
## [1] 8
```

```
(1 + 7)>= 4
```

```
## [1] TRUE
```

```
mean(c(1,7))
```

```
## [1] 4
```

R refresher

4. R as calculator

```
1 + 7
```

```
## [1] 8
```

```
(1 + 7)>= 4
```

```
## [1] TRUE
```

```
mean(c(1,7))
```

```
## [1] 4
```

5. Create objects with assignment operator <-

R refresher

4. R as calculator

```
1 + 7
```

```
## [1] 8
```

```
(1 + 7)>= 4
```

```
## [1] TRUE
```

```
mean(c(1,7))
```

```
## [1] 4
```

5. Create objects with assignment operator <-
    - Don't use = for assignment (even thought it works)

R refresher

Let's open RStudio and [**Lab 1 practice code**]