

CSSS 510: Lab 1

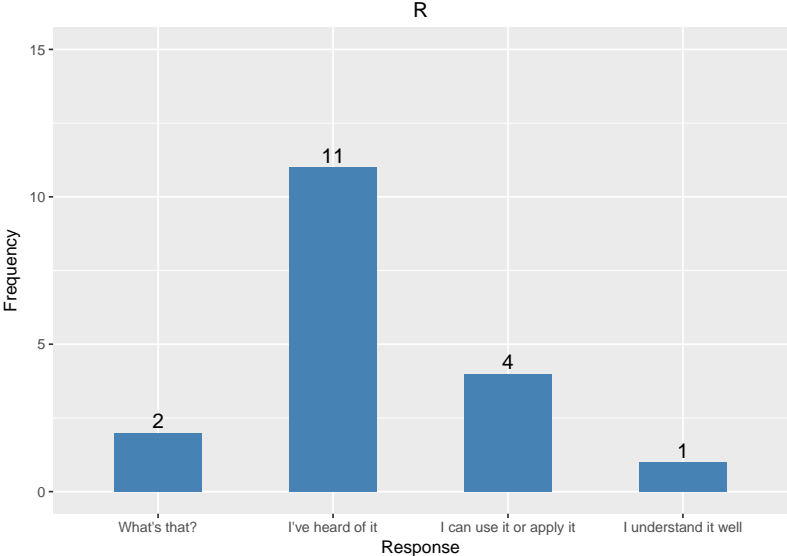
Logistics & R Refresher

2017-9-29

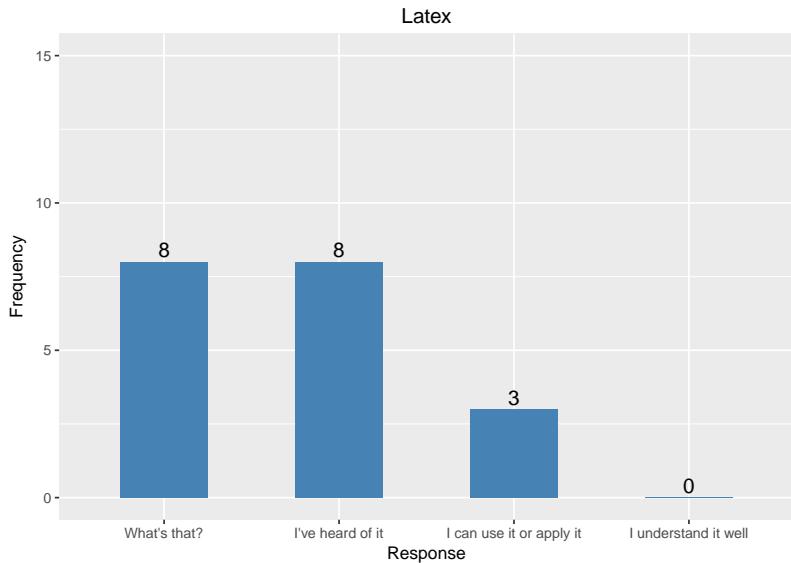
Logistics

- 1. Lab Sessions:** Fri, 4:00-5:20pm in Savery 117
 - ▶ Covers application of material from lecture using examples; clarification and extension of lecture material; Q & A for homeworks and lectures
 - ▶ Materials will be available on the **course website**
- 2. Office Hours:** Tues, 3:00-4:20pm in Smith 220
 - ▶ Available for trouble shooting and specific questions about homework and lecture materials
- 3. Homeworks:** 5-6 due every 2 weeks or so
 - ▶ 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
 - ▶ We will use two of Chris's packages extensively: `simcf` and `tile`

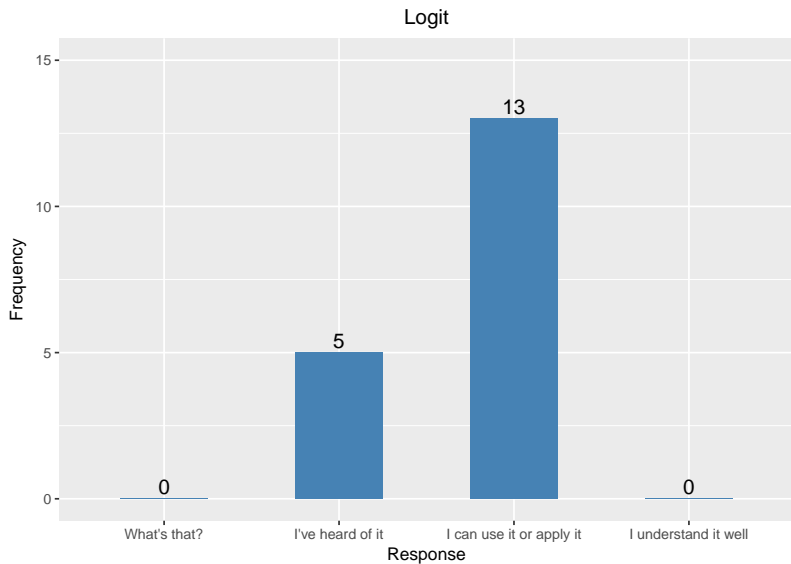
Survey Results



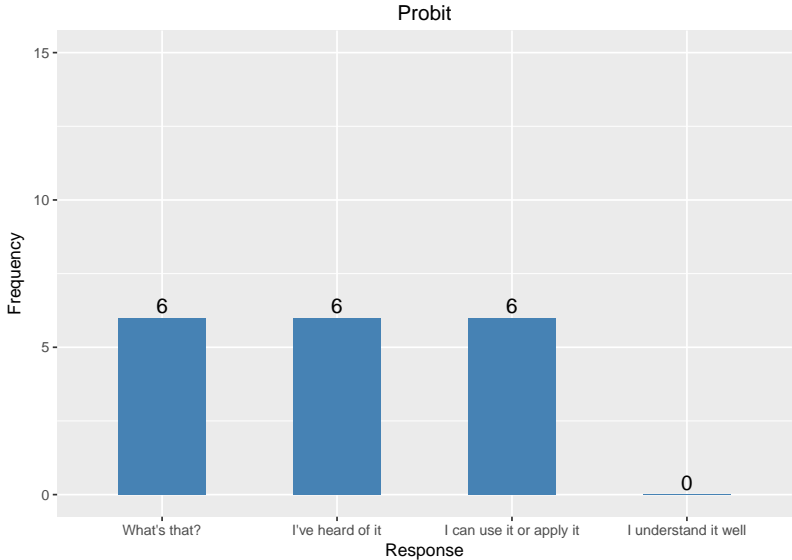
Survey Results



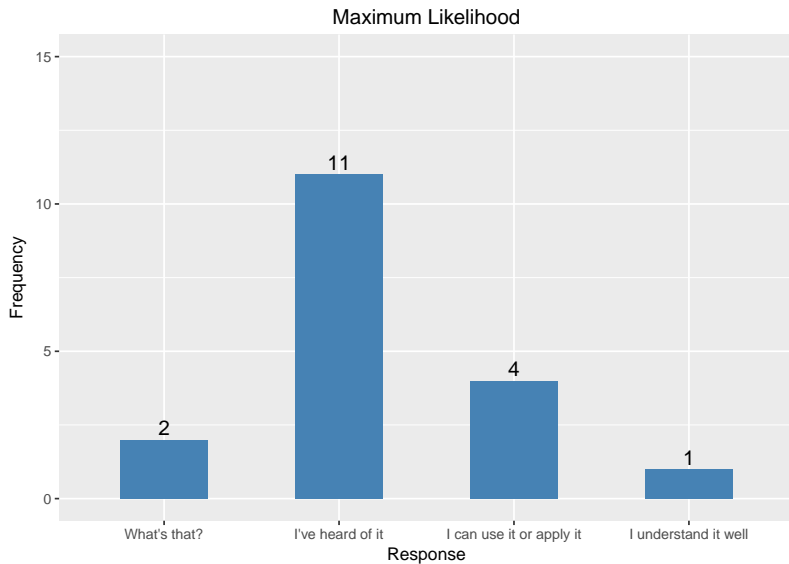
Survey Results



Survey Results



Survey Results



Logistics

1. When this course is over, you should be able to do the following (and much more):
 - ▶ Fit a logistic regression 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 $\hat{\pi}$ across counterfactual values of \mathbf{x}
 - ▶ 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
 - ▶ Fit a variety of bounded and unbounded count models that address overdispersion
 - ▶ Use one of several algorithms to impute missing data

Logistics

2. The course moves fast: you should comfortable doing the following for the homework assignments and project
 - ▶ tidying and transforming data
 - ▶ importing and exporting data sets
 - ▶ generating plots of your data and results
 - ▶ writing basic functions and loops for repeated procedures

- ▶ Fortunately, for those of you new to R, there are many resources to get you up to speed
 - ▶ Zuur et al. (2009), Chapter 1-5
 - ▶ Wickham and Groleman (2017)

R Refresher

Data Objects

Create the following vectors

1. vector.1 : 1,2,3,4,5,6,6,6,6,6
2. vector.2: 10 randomly drawn numbers from a normal distribution with a mean 10 and a s.d. of 1
3. vector.3: Results of 10 single binomial trials with a probability of 0.4
4. vector.4: Sample 100 observations from a 5-trial binomial distribution with a probability of success of 0.4

Vectors

```
#Clear memory  
rm(list=ls())  
  
vector.1 <- c(seq(1,5,1), rep(6,5))  
  
vector.2 <- rnorm(10, 10, 1)  
  
#help?  
?rnorm  
  
vector.3 <- rbinom(10, 1, 0.4)  
  
vector.4 <- rbinom(100, 5, 0.4)
```

Vectors

5. Check what type of data vector.2 is
6. Round up vector.2 to two decimal place

Vectors

```
is.character(vector.2)
```

```
## [1] FALSE
```

```
mode(vector.2)
```

```
## [1] "numeric"
```

```
round(vector.2, 2)
```

```
## [1] 10.06  9.62 10.42 10.32  9.62  8.97  8.99  9.01 11.88 10.41
```

Matrices

7. matrix.1: Create 5 by 5 matrix containing all NAs
8. Assign matrix.1 the row names (a,b,c,d,e) and the column names (1,2,3,4,5)
9. Replace the NAs in the first column of matrix.1 with Inf

Matrices

```
matrix.1<-matrix(NA, nrow=5, ncol=5)

rownames(matrix.1)<-c("a","b","c","d","e")
colnames(matrix.1)<-c(1,2,3,4,5)

matrix.1[,1]<-Inf
```

Lists

10. Create a list that contains vector.1, vector.2, and matrix.1
11. Locate vector.2 from the list

Lists

```
list.1 <- list(vector.1, vector.2, vector.3, matrix.1)
names(list.1) <-
  c("vector.1", "vector.2", "vector.3", "matrix.1")

list.1[[2]]
```

```
## [1] 10.058525  9.615536 10.423318 10.324244  9.620970  8.973722  8.990473
## [8]  9.013487 11.880644 10.414685
```

```
list.1$vector.2
```

```
## [1] 10.058525  9.615536 10.423318 10.324244  9.620970  8.973722  8.990473
## [8]  9.013487 11.880644 10.414685
```

Data Frames

Data frames are a special type of list in which each row has same length. It is also a matrix like object, yet its elements - unlike elements in a matrix - doesn't have to be of same type. Most of the data we use are in data frames.

12. Open Lab1data.csv in R
13. Is it a data frame? Is it a matrix?
14. Check the names and summary statistics of the data
15. Remove observations with missing values
16. Plot GDP per capita (on the x-axis) and polity2 (on the y-axis)
17. Create a new variable called "democracy". Assign 0 to countries with negative value or zero polity2 score, and assign 1 to countries with positive score.
18. Use a loop to do the same recoding

Data Frames

```
library(foreign)
library(tidyverse)

## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr

## Conflicts with tidy packages -----

## filter(): dplyr, stats
## lag():    dplyr, stats

setwd("/Users/danielyoo/CSSS-POLS-510-MLE/Lab1Slides")

data<-read.csv("Lab1data.csv", header=T)
```

Data Frames

```
is.data.frame(data) #Yes!
```

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

```
is.matrix(data) #No
```

```
## [1] FALSE
```

```
is.character(data$Year)
```

```
## [1] FALSE
```

```
data$Year<-as.character(data$Year)
```

Data Frames

```
names(data)
```

```
## [1] "country"
```

```
## [2] "Year"
```

```
## [3] "GDP.per.capita.PPP.current.international"
```

```
## [4] "polity2"
```

Data Frames

```
summary(data)
```

```
##              country              Year
## Afghanistan      : 11  Length:1914
## Albania           : 11  Class  :character
## Algeria           : 11  Mode   :character
## Andorra           : 11
## Angola            : 11
## Antigua and Barbuda: 11
## (Other)           :1848
## GDP.per.capita.PPP.current.international  polity2
## Min.      : 219.2      Min.      :-10.000
## 1st Qu.: 1625.0      1st Qu.: -4.000
## Median   : 4299.2    Median   :  5.000
## Mean     : 7874.9    Mean     :  2.431
## 3rd Qu.: 9818.6    3rd Qu.:  8.000
## Max.     :91712.3    Max.     : 10.000
## NA's     :373       NA's     :542
```

Data Frames

```
head(unique(data$country)) # observations on 174 countries
```

```
## [1] Antigua and Barbuda Afghanistan Albania  
## [4] Algeria Andorra Angola  
## 174 Levels: Afghanistan Albania Algeria Andorra ... Zimbabwe
```

```
head(tapply(data$country, data$Year, length))
```

```
## 2000 2001 2002 2003 2004 2005  
## 174 174 174 174 174 174
```

```
head(tapply(data$Year, data$country, length))
```

```
## Afghanistan Albania Algeria  
## 11 11 11  
## Andorra Angola Antigua and Barbuda  
## 11 11 11
```

Data Frames

```
data<-na.omit(data) # listwise deletion!!
```

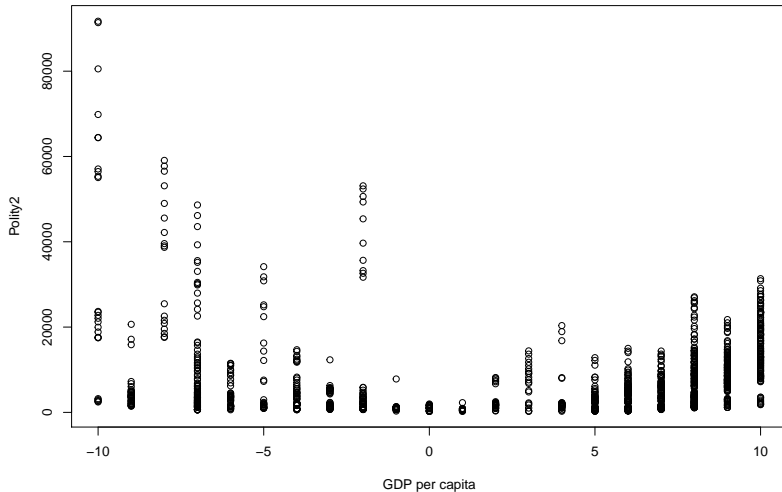
```
dim(data)
```

```
## [1] 1305    4
```

```
attach(data)
```


Data Frames

```
plot(polity2, GDP.per.capita.PPP.current.international, ylab="Polity2", xlab="GDP per capita")
```



Data Frames

```
data$democracy[data$polity2>0]<-1  
data$democracy[data$polity2<0|data$polity2==0]<-0  
summary(data$democracy)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## 0.0000  0.0000  1.0000  0.6322  1.0000  1.0000
```

Data Frames

```
data$democracy.2<-rep(NA, length(data$polity2)) # 1305
```

```
for (i in 1:length(data$polity2)) {  
  if (data$polity2[i]>0) data$democracy.2[i]<-1  
  else data$democracy.2[i]<-0  
}
```

```
head(cbind(data$democracy, data$democracy.2))
```

```
##      [,1] [,2]  
## [1,]    1    1  
## [2,]    1    1  
## [3,]    1    1  
## [4,]    1    1  
## [5,]    1    1  
## [6,]    1    1
```

Data Frames

19. Subset the data frame to show only country name and GDP per capita
20. Rearrange the columns of the data frame ascending by polity score
21. Show only values of GDP per capita for South Africa from 2002 to 2008
22. Create a new variable that takes the first letter of the country and attaches it to the year of observation
23. Find the mean of GDP per capita for each year of observation

Data Frames

```
library(tidyverse)
head(select(data, country, GDP.per.capita.PPP.current.international))
```

```
##   country GDP.per.capita.PPP.current.international
## 23 Albania                                4259.308
## 24 Albania                                4658.009
## 25 Albania                                4860.035
## 26 Albania                                5230.007
## 27 Albania                                5673.623
## 28 Albania                                6161.608
```

```
head(data[, c(1,3)])
```

```
##   country GDP.per.capita.PPP.current.international
## 23 Albania                                4259.308
## 24 Albania                                4658.009
## 25 Albania                                4860.035
## 26 Albania                                5230.007
## 27 Albania                                5673.623
## 28 Albania                                6161.608
```

```
head(data.frame(data$country, data$GDP.per.capita.PPP.current.international))
```

```
##   data.country data.GDP.per.capita.PPP.current.international
## 1      Albania                                4259.308
## 2      Albania                                4658.009
## 3      Albania                                4860.035
## 4      Albania                                5230.007
## 5      Albania                                5673.623
## 6      Albania                                6161.608
```

Data Frames

```
head(arrange(data, polity2))
```

```
## country Year GDP.per.capita.PPP.current.international polity2 democracy
## 1 Bhutan 2000 2436.943 -10 0
## 2 Bhutan 2001 2587.442 -10 0
## 3 Bhutan 2002 2775.398 -10 0
## 4 Bhutan 2003 2984.397 -10 0
## 5 Bhutan 2004 3219.421 -10 0
## 6 Qatar 2000 55053.515 -10 0
## democracy.2
## 1 0
## 2 0
## 3 0
## 4 0
## 5 0
## 6 0
```

```
head(data[order(data$polity2),])
```

```
## country Year GDP.per.capita.PPP.current.international polity2
## 166 Bhutan 2000 2436.943 -10
## 167 Bhutan 2001 2587.442 -10
## 168 Bhutan 2002 2775.398 -10
## 169 Bhutan 2003 2984.397 -10
## 170 Bhutan 2004 3219.421 -10
## 1387 Qatar 2000 55053.515 -10
## democracy democracy.2
## 166 0 0
## 167 0 0
## 168 0 0
## 169 0 0
## 170 0 0
## 1387 0 0
```

Data Frames

```
head(filter(data, country=="South Africa"), Year>=2002 & Year<=2008))
```

```
##           country Year GDP.per.capita.PPP.current.international polity2
## 1 South Africa 2002                               7244.218           9
## 2 South Africa 2003                               7522.254           9
## 3 South Africa 2004                               7992.767           9
## 4 South Africa 2005                               8596.831           9
## 5 South Africa 2006                               9269.283           9
## 6 South Africa 2007                               10002.543          9
##  democracy democracy.2
## 1           1           1
## 2           1           1
## 3           1           1
## 4           1           1
## 5           1           1
## 6           1           1
```

```
head(subset(data, data$country=="South Africa") & data$Year>=2002 & Year<=2008))
```

```
##           country Year GDP.per.capita.PPP.current.international polity2
## 1444 South Africa 2002                               7244.218           9
## 1445 South Africa 2003                               7522.254           9
## 1446 South Africa 2004                               7992.767           9
## 1447 South Africa 2005                               8596.831           9
## 1448 South Africa 2006                               9269.283           9
## 1449 South Africa 2007                               10002.543          9
##  democracy democracy.2
## 1444           1           1
## 1445           1           1
## 1446           1           1
## 1447           1           1
## 1448           1           1
## 1449           1           1
```

Data Frames

```
head(mutate(data, paste(substring(data$country, 1, 1), data$Year, sep="")))
```

```
##   country Year GDP.per.capita.PPP.current.international polity2 democracy
## 1 Albania 2000                                4259.308           5           1
## 2 Albania 2001                                4658.009           5           1
## 3 Albania 2002                                4860.035           7           1
## 4 Albania 2003                                5230.007           7           1
## 5 Albania 2004                                5673.623           7           1
## 6 Albania 2005                                6161.608           9           1
##   democracy.2 paste(substring(data$country, 1, 1), ...
## 1             1                                A2000
## 2             1                                A2001
## 3             1                                A2002
## 4             1                                A2003
## 5             1                                A2004
## 6             1                                A2005
```


Data Frames

```
data%>%  
  group_by(Year)%>%  
  summarize(mean(GDP.per.capita.PPP.current.international, na.rm=T)  
            )
```

```
## # A tibble: 10 × 2  
##   Year `mean(GDP.per.capita.PPP.current.inter...`  
##   <chr> <dbl>  
## 1 2000 5757.223  
## 2 2001 5976.854  
## 3 2002 6167.580  
## 4 2003 6597.168  
## 5 2004 7157.506  
## 6 2005 7712.546  
## 7 2006 8416.708  
## 8 2007 9218.926  
## 9 2008 9566.308  
## 10 2009 9113.082
```