

R – Programming

- Tejas Vaidya

15th October 2016



CALIFORNIA STATE UNIVERSITY
FULLERTON™

About me ?

- Student like you guys..
- I love playing with data and learning new tools..
- Undergrad in Information technology..
- Worked on projects with C, C++, JAVA, VB, HTML, JSP, SQL..
- Lets learn R..



R is easy to learn because...

- Its free.. who doesn't like free stuff...
- Object Oriented..
- Made with C, Fortran
- It has command line interface.. That's not exiting..
- But we still have the R Studio IDE..That's a life saver.
- R studio is free too..



We will learn the following thing today..

1. Playing with variables.
2. R programming for handling data.
3. Using statistics.
4. Working with visualization.
5. Understanding IDE.



Using the command line interface.

- $2+2$
- π
- $0/0$
- $2*\text{Inf}$
- “CSUF BDAC”
- `?sqrt`



Assigning values to variables

- `X<-2`
- `Y=10`
- `Z=x*y`
- `B="BDAC"`
- `A=B*Z`
- `k=<-TRUE`
- `1==2`
- `Ls()`
- `Rm()`



Vectors

- `c(1,2,3,4,5,6,7,8)`
- `c(1:8)`
- `c(a,b,c,d,e)`
- `c("a","b","c","d","e")`
- `a=c(1:8)`
- `a*10`
- `b<-c(1,1,1,1,1,1,1,1)`
- `a+b`



Factors

Type of vector but with levels.

```
>r<-gl(2,5,labels=c("CSUF","BDAC"))
```

Description:

Defines a factor with labels CSUF and BDAC with 2 levels and 5 repetitions.

Lets try the following

```
>Table(r)
```

Factors can be ordered or unordered and used for categorical variables



Matrix

- `a<-c(1:6)`
- `matrix(a,nrow=2)`
- `matrix(a,ncol=2)`
- `x=matrix(a,nrow=2)`
- `x*2`
- `y=matrix(c(10,20,30,40,50,60),nrow=2)`
- `z=x*y`
- `z`
- `z[2,1]`



Array

```
z<-array(1:100)
```

Creates one dimensional array

```
z<-array(1:4,dim=c(2,2))
```

Creates two dimensional array of numbers 1:4

```
z<-array(1:12,dim=c(2,2,3))
```

Creates three dimensional array of numbers 1:4

How will you create 4 dimensional array..



Data frame

- Used to store data with different data type.
- Best for data operations.
- `a<- data.frame(
department=c("ISDS","MBA","MSIT","ACCOUNTING"),
total=c(100,200,150,180))`
- `a[,2]`
- `Plot(a[,2])`



Loops

```
for(a in 1:3)
{
cat("Iteration ", a, "\n")
}
```

Cat prints the arguments passed to it.



Condition

```
for(a in 1:5)
{
if(a>3)
{
cat("Iteration ", a, "\n")
}
}
```



Lets do some basic statistics



Descriptive Statistics

- `a<-c(1,1,2,3,3,4,4,5,5,5,6,9,9,9,9)`
- `b<-c(8,9,9,9,10,11,18,19,21,22,20,20,20,25,50)`
- `length(a)`
- `length(b)`
- `mean(a)`
- `median(b)`



Descriptive Statistics

- `quantile(a)`
- `var(b)` //Calculates variance
- `cor(a,b)` //Calculates co-relation
- `summary(b)`
- `min(a)`
- `max(b)`



Sample Data Analysis

- Loading data into objects
- `a<-read.csv("C:/Users/Tejas/Documents/R/Data.csv")`
- The above way is the normal way to load data in objects on local drive. For today's workshop we are going to use data from MASS library of R
- `library("MASS")`
- `cars`
- `with(cars,plot(speed,dist))`
- `with(cats,cor(speed,dist))`
- `summary(cars)`



Continued

- Regression

Use lm function with formula Dependent Variable ~ Independent Variable.

- `lm(speed ~ dist,data=cars)`

Now save the output in a variable

- `lm(speed ~ dist,data=cars)`

- `regoutput<-lm(speed ~ dist,data=cars)`

Plot the data again with regression line

- `abline(regoutput,col="red")`

- More statistical input through the following

Try this for more statistical insights.

- `summary(regoutput)`
- `anova(regoutput)`



Working with data and visualization



Browsing Data

- `library("MASS")`
- `cpus`
- `a<cpus`
- `head(a)`
- `tail(a)`
- `colnames(a)`
- `a[1]`
- `a[1:4]`
- `a[,2]`
- `a[1,]`
- `a[1:3,1:3]`
- `summary(a)`
- `with(a,plot(cach))`



Visualizations

- We will use ggplot2 package for graphics.
- Please make sure your computer is connected to internet.
- Use the following command to install the package
- `install.packages("ggplot2")`
- select server as USA CA
- Let the package install
- Load the library using following command
- `require(ggplot2)`

- create histogram using following command
- `ggplot(a,aes(x=cach))+geom_histogram()`



Visualizations (continued)

- Creating density plot
- `ggplot(a,aes(x=cach))+geom_density()`
- Create boxplot using the following code.
- `ggplot(a,aes(x=1, y=mmin))+geom_boxplot()`

Working with categorical data

- `a<-crabs`
- `a`
- Generating side by side density plot
- `ggplot(a,aes(x=FL))+geom_density()+facet_wrap(~sex)`
- Generating side by side boxplot
- `ggplot(d,aes(x=FL))+geom_density()+facet_wrap(~sex)`
- Working with correlation



Visualizations (continued)

- Working with correlation
- `cor(a[,4:8])`
- Gives tabular view of correlation
- Lets get a visual overview of co-relation.
- Install the following package
- `install.packages("GGally")`
- Load the package
- `library("GGally")`
- `ggpairs(a[4:8])`

Study the visualization



R STUDIO



THANK YOU



CALIFORNIA STATE UNIVERSITY
FULLERTON™