Stats with 

By: Fatemeh Torabi

2017

Instructions for use

Presented codes in this document aims to illustrate how certain data retrieval tasks can be conducted using Rstudio.

SOFTWARES IN USE

› **Install R:**

› Go to https://cran.r-project.org/ and follow the link for your operating system.

› **Install RStudio:**

Go to https://www.rstudio.com/products/rstudio/download/ and click on the installer link for your operating system.

PACKAGES IN USE

```
install.packages("devtools")
library("devtools")

install.packages("dplyr")
library("dplyr")

install.packages("ggplot2")
library("ggplot2")
install.github("StatsWithR/statsr")
```

› See this video for step-by-step installation instructions if needed.

› You might experience problem while trying to install the "statsr" package from github, in that case you can download the package and manually import it or only import the datasets which we used from the package.

Load example dataset

```
# Import data
load(url('http://s3.amazonaws.com/assets.datacamp.com/course/dasi/ames.RData'))
```

Headings

Exploratory analysis

Getting started with data and conduct initial exploration.

Sampling

Taking a predetermined number of observations from a larger population.

Confidence Interval

Defining a range of values such that there is a specified probability that the value of a parameter lies within it.

Hypothesis test

A statistical test that is used to determine whether there is enough evidence in a sample for examining two opposing hypotheses.

Linear Regression

Estimating the linear relationships among variables using various techniques.

Multiple Regression

An extension of simple linear regression for prediction line based on multiple variabels.

Process

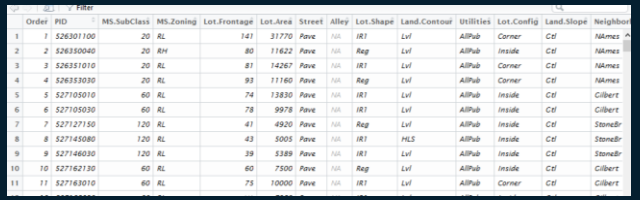
Exploratory data analysis

```
# View data table
View(ames)

# Dimension of dataset
dim(ames)

# column headings
names(ames)

# assignment
area = Ames$Gr.Liv.Area
price = Ames$SalePrice

# variables in the dataset
str(ames)

#compare two groups using by() and summary():
by(ames$Heating, ames$Yr.Sold, summary)
```

Exploratory data analysis

```
# View data table
View(ames)

# Dimension of dataset
dim(ames)

# column headings
names(ames)


# variables in the dataset
str(ames)


#compare two groups using by() and summary():
by(ames$Heating, ames$Yr.Sold, summary)  #sd or IQR can be used
```

outputs



	Order	PID	MS.SubClass	MS.Zoning	Lot.Frontage	Lot.Area	Street	Alley	Lot.Shape	Land.Contour	Utilities	Lot.Config	Land.Slope	Neighbor
1	1	526301100	20	RL	141	31770	Pave	NA	IR1	Lvl	AllPub	Corner	Gtl	NAmes
2	2	526350040	20	RH	80	11622	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	NAmes
3	3	526351010	20	RL	81	14267	Pave	NA	IR1	Lvl	AllPub	Corner	Gtl	NAmes
4	4	526353030	20	RL	93	11160	Pave	NA	Reg	Lvl	AllPub	Corner	Gtl	NAmes
5	5	527105010	60	RL	74	13830	Pave	NA	IR1	Lvl	AllPub	Inside	Gtl	Gilbert
6	6	527105030	60	RL	78	9978	Pave	NA	IR1	Lvl	AllPub	Inside	Gtl	Gilbert
7	7	527127150	120	RL	41	4920	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	StoneBr
8	8	527145080	120	RL	43	5005	Pave	NA	IR1	HLS	AllPub	Inside	Gtl	StoneBr
9	9	527146030	120	RL	39	5389	Pave	NA	IR1	Lvl	AllPub	Inside	Gtl	StoneBr
10	10	527162130	60	RL	60	7500	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	Gilbert
11	11	527163010	60	RL	75	10000	Pave	NA	IR1	Lvl	AllPub	Corner	Gtl	Gilbert

```
> names(ames)
 [1] "Order"          "PID"            "MS.SubClass"
 [4] "MS.Zoning"      "Lot.Frontage"   "Lot.Area"
 [7] "Street"         "Alley"          "Lot.Shape"
[10] "Land.Contour"   "Utilities"      "Lot.Config"
[13] "Land.Slope"     "Neighborhood"   "Condition.1"
[16] "Condition.2"    "Bldg.Type"      "House.Style"
[19] "Overall.Qual"   "Overall.Cond"   "Year.Built"
```

```
> str(ames)
'data.frame':   2930 obs. of  84 variables:
 $ Order          : int  1 2 3 4 5 6 7 8 9 10 ...
 $ PID            : int  526301100 526350040 526351010 526353030 527105010 527105030 ...
 $ MS.SubClass    : int  20 20 20 20 60 60 120 120 120 60 ...
 $ MS.Zoning      : Factor w/ 7 levels "A (agr)","C (all)",..: 6 5 6 6 6 6 6 6 6 6 ...
 $ Lot.Frontage   : int  141 80 81 93 74 78 41 43 39 60 ...
 $ Lot.Area       : int  31770 11622 14267 11160 13830 9978 4920 5005 5389 7500 ...
 $ Street         : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...
 $ Alley          : Factor w/ 2 levels "Grvl","Pave": NA NA NA NA NA NA NA NA NA NA ...
 $ Lot.Shape      : Factor w/ 4 levels "IR1","IR2","IR3",..: 1 4 1 4 1 1 4 1 1 4 ...
 $ Land.Contour   : Factor w/ 4 levels "Bnk","HLS","Low",..: 4 4 4 4 4 4 4 2 4 4 ...
 $ Utilities      : Factor w/ 3 levels "AllPub","NoSewa",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Lot.Config     : Factor w/ 5 levels "Corner","CulDSac",..: 1 5 1 1 5 5 5 5 5 5 ...
```

```
> by(ames$Heating, ames$Yr.Sold, summary)
ames$Yr.Sold: 2006
Floor  GasA  GasW  Grav  OthW  Wall
    0   616     6     1     0     2
------------------------------------------------
ames$Yr.Sold: 2007
Floor  GasA  GasW  Grav  OthW  Wall
    0   684     8     2     0     0
```

“dplyr” package and Piping operator: %>%

mutate()	adds new variables that are functions of existing variables
select()	picks variables based on their names.
filter()	picks cases based on their values.
summarise()	reduces multiple values down to a single summary.
arrange()	changes the ordering of the rows.

Summary statistics

```
# Obtaining summary statistics from data

ames %>% summarise(mu = mean(area), pop_med = median(area),
sigma = sd(area), pop_iqr = IQR(area),
pop_min = min(area), pop_max = max(area),
pop_q1 = quantile(area, 0.25), # first quartile, 25th percentile
pop_q3 = quantile(area, 0.75), # third quartile, 75th percentile
N = n())
```

Select

```
# Selecting subset columns
Ames %>% select(Yr.Sold, Misc_Feature)
```

Mutate

```
# calculating the total area of properties in a new column
ames <- ames %>% mutate(Total_area = Pool.Area + Garage.Area +
Gr.Liv.Area + Mas.Vnr.Area)
```

```
# Add a flag for those houses with sale Price greater than 150k:
ames <- ames %>% mutate(Price150 = SalePrice > 150000)
```

Example:

SQL DB2

```
SELECT
          Gr.Liv.Area, SalePrice, Year.Built
FROM
          ames
WHERE
          Year.Built < 1900
```

R

```
ames %>%

select(Gr.Liv.Area, SalePrice, Year.Built) %>%

filter(Year.Built < 1900)
```

Example:

SQL DB2

```
SELECT
          distinct Year.Built,
          count (*)
FROM
          ames
GROUP BY
          Year.Built
```

R

```
ames %>%

Group_by(Year.Built) %>%

summarise(group_size = n())
```

Question:

What is the proportion of 1 story houses built in 2010 and sold for more than 250000$?

```
ames10 <- ames %>% filter(House.Style == '1Story', Year.Built=="2010" )
```

```
# calculating the proportion

sum(ames10$SalePrice > '250000') / length(ames10$Order)
```

Sampling

Simple Random Sampling (SRS)

```
# SRS: randomly selecting 50 houses from the dataset
samp1 <- ames %>% sample_n(size = 50)
```

Repeating SRS for 15000

```
sample_means50 <- ames %>%
          rep_sample_n(size = 50, reps = 15000, replace = TRUE) %>%
          summarise(x_bar = mean(Gr.Liv.Area))
```

Confidence Interval

for population mean

$$\bar{x} \pm z^* \frac{s}{\sqrt{n}}$$

```
population <- ames$Gr.Liv.Area
samp <- sample(population, 60)
```

Point estimate

```
sample_mean <- mean(samp)
```

95% CI

```
se <- sd(samp) / sqrt(60)
lower <- sample_mean - 1.96 * se
upper <- sample_mean + 1.96 * se
c(lower, upper)
```

Hypothesis Testing

for mean

H0: Average sale Price for Houses with or without Central Air

H1: Average sale Price for Houses with or without Central Air are different

```
t.test(SalePrice ~ Central.Air, data =ames, conf.level = 0.95)
```

Further investigation:

```
by (ames$SalePrice , ames$Central.Air, summary)
by (ames$SalePrice , ames$Central.Air, sd)
```

```
boxplot(SalePrice ~ Central.Air, data =ames, ylab="Sales Price
of Houses" , xlab = "Central Air")
```

Linear Regression & Multiple Linear Regression

Parsimonious model: prefer the simplest best model.

Visually inspect the data:

```
scatter.smooth( ames$Gr.Liv.Area , ames$SalePrice)
```

```
# ANOTHER OPTION
am<-ames[,c(48,82)] #RUN THE RIGHT CORNER FUNTION
pairs(am, upper.panel = panel.cor)
```

Fit the regression line:

```
LinReg <-lm( Gr.Liv.Area ~ SalePrice, data=ames)
Summary(reg_line) # get the summary of coefficients
```

```
mlR <-lm( Gr.Liv.Area ~ SalePrice + House.Style,
data=ames)
Summary(reg_line) # get the summary of coefficients
```

```
panel.cor <- function(x, y, digits = 2, cex.cor)
{                usr <- par("usr"); on.exit(par(usr))
                 par(usr = c(0, 1, 0, 1))
# correlation coefficient
r <- cor(x, y)
txt <- format(c(r, 0.123456789), digits = digits)[1]
txt <- paste("r= ", txt, sep = "")
text(0.5, 0.6, txt)
# p-value calculation
p <- cor.test(x, y)$p.value
txt2 <- format(c(p, 0.123456789), digits = digits)[1]
txt2 <- paste("p= ", txt2, sep = "")
if(p<0.01) txt2 <- paste("p= ", "<0.01", sep = "")
text(0.5, 0.4, txt2)}
```

THANK YOU!

Any questions?

You can find resources at

ownCloud\PrudentHealthcare\PH_TeamManagement\Training\Courses\Statistics with R