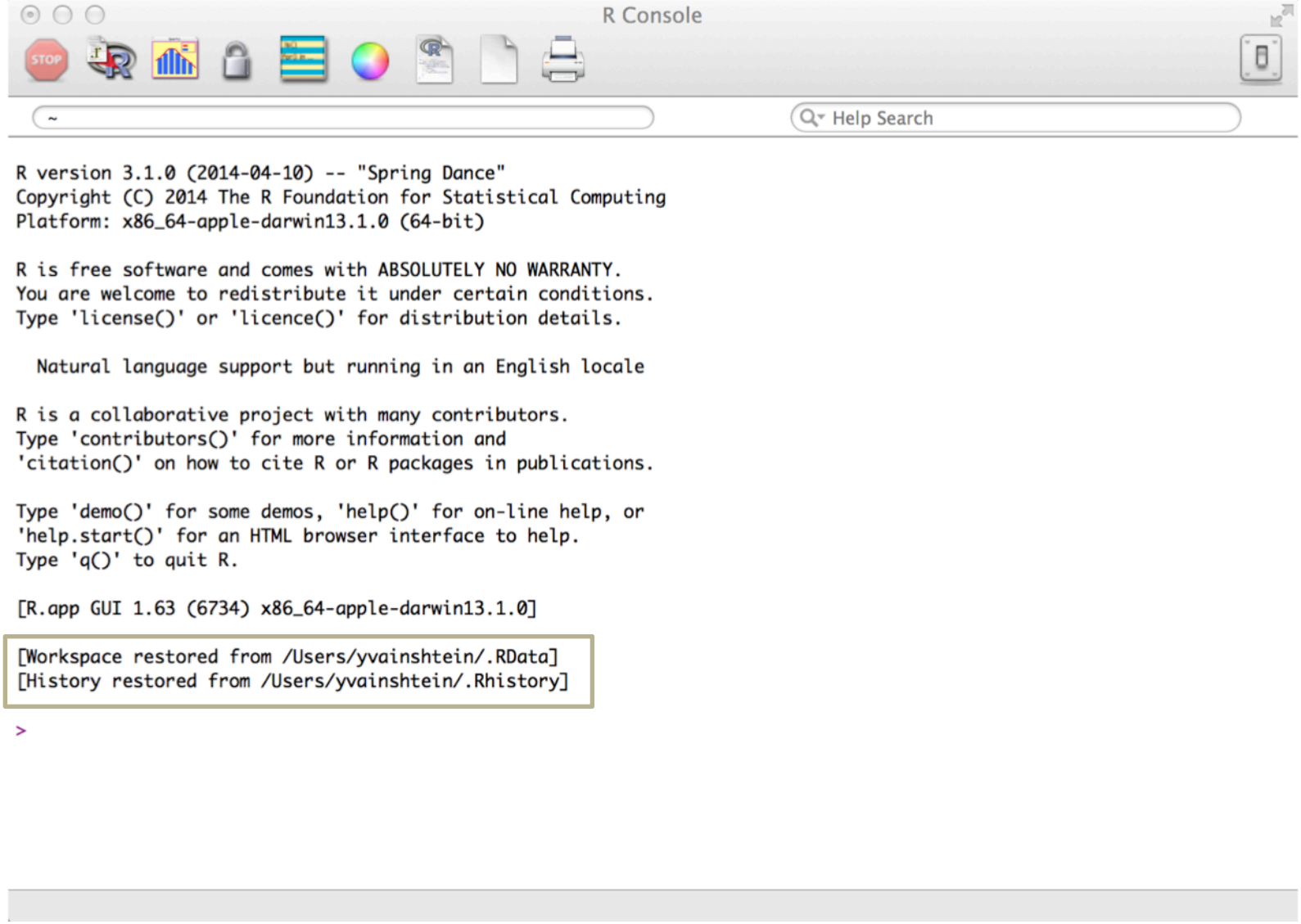
R course for beginners.

Practical sessions

Evgeniy Vainshtein
ZMBH, room 504

e-mail: y.vainshtein@zmbh.uni-heidelberg.de
tel: +49-6221-54-6796

Locate “history of commands” file in R



History of commands (Session 1)

- Basic arithmetical operation in console
- Data types: variables, vectors, arrays
- Manipulations with variables, vectors and arrays
- Basic commands and functions
- Reading/writing data
- working directory and files listing

History of commands (Session 1)

Variables and vectors

```
# simple calculations
10
10 * 20
10^4

# simple calculations of a variable
x <- 10
x
x * 20; x^4

# vectors
c(1,2,3,4)

# what does c() mean?
# c is a function - combine
?c

# lets assign the vector 1,2,3,4 to variable v
v <- c(1,2,3,4)

# simple calculations on a vector
v * 20
v^4
x <- v ^ 4
v
x

# accessing parts of vectors
x[1]
x[2]
x[c(1,4)]

# useful functions to apply to vectors
sum(v)
mean(v)
length(v)
paste(a,b,sep=" ")
```

History of commands (Session 1)

Manipulations with vectors

```
# generating sequences of numbers
1:10
c(1:10)  # exactly the same thing

#generate sequence of numbers with step 1
and 0.5
seq(1,10, by=1)
seq(1,10, by=.5)
x <- 1:10
rep(x, times=3)
rep(x, times=3, each=5)

# logical vectors
x <- rep(1:10,times=3)
# find all in x grater than 5
x > 5
gt5 <- x > 5
gt5
x[gt5]
```

Matrices

```
# create 2 matrices of 4 columns by 5 rows
containing sequence on numbers
x <- array(1:20, dim=c(4,5))
y <- array(21:41, dim=c(4,5))
# operations with matrices
x
x*20        x^2        x/10
# get element in first row, second column
# get first row or third column
      x[1,2]      x[1,]      x[,3]
# remove second row, first column
x[-2, ]      x[,-1]
# transpose the matrix
      t(x)
# combine by columns
      cbind(x,y)
# combine by rows
      rbind(x,y)
```

History of commands (Session 1)

Files and folders manipulations

```
# check working directory (where we are now)
getwd()
# set new working directory
setwd("~/Desktop/R_data")

# list all files in the folder matching pattern (contains word "sleep" in the filename)
dir(pattern="sleep")
# check user manual for more details about flags and parameters
?dir
# list files selected by pattern, using regular expression, in case insensitive mode, displaying full path
dir(ignore.case=TRUE, full.names=TRUE, pattern="^sleep.*copy.*")
# check manual for similar commands:
?list.files
?list.dirs

# read tab-delimited files
?read.delim
# read coma-separated files
?read.csv
# read generic table
    ?read.table
# load the data from sleep_data_simple.txt tab-delimited text file, with the header
    read.table("sleep_data_simple.txt", sep="\t", header=TRUE)
# load data from URL
    read.table(url("http://genome-www.stanford.edu/cellcycle/data/rawdata/combined.txt"), sep="\t", header=TRUE)

# save variable "sleepdata" to file without quoting characters, using "|" as a separator
    write.table(sleepdata, "sleepdata_mod.txt", quote=FALSE, sep="|")
```

History of commands (Session 2)

- Usage of standard datasets to test R scripts
- Basic data description
- Frequency distribution of quantitative data
    - cumulative frequency graph
    - relative frequency distribution
    - histogram
    - scatterplot
- Test data for normality
    - generate random binomial dataset
    - Shapiro-Wilk normality test;
    - QQ-plot (Quantile-Quantile plot)

History of commands (Session 2)

Standard data sets in R

```
# show full list of default datasets in R
        library(help = "datasets")
# detailed description of a datasets
        ?USArrests
        ?mtcars
        ?faithful
        ?sleep
```

```
# describe data
        summary(faithful)
```

```
# some useful commands
head()      # print first lines of array
colnames()      # columns name
rownames()      # rows name
range()     # data range (from min to max
nrow()      # number of rows
ncol()      # number of columns
```

faithful {datasets} R Documentation

Old Faithful Geyser Data

Description

Waiting time between eruptions and the duration of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

Usage

```
faithful
```

Format

A data frame with 272 observations on 2 variables.

[,1] eruptions numeric Eruption time in mins
[,2] waiting numeric Waiting time to next eruption (in mins)

Details

A closer look at `faithful$eruptions` reveals that these are heavily rounded times originally in seconds, where multiples of 5 are more frequent than expected under non-human measurement. For a better version of the eruption times, see the example below.

There are many versions of this dataset around: Azzalini and Bowman (1990) use a more complete version.

Source

W. Härdle.

References

Härdle, W. (1991) *Smoothing Techniques with Implementation in S*. New York: Springer.

Azzalini, A. and Bowman, A. W. (1990). A look at some data on the Old Faithful geyser. *Applied Statistics* **39**, 357–365.

History of commands (Session 2)

Analyze frequency distribution of geyser eruption duration (use faithful dataset)

```
# access duration data
duration = faithful$eruptions
# check range of the observed duration
range(duration)

# Break the range into non-overlapping intervals
breaks = seq(1.5, 5.5, by=0.5)

# classify the eruption duration according to half-minute non-overlapping intervals
?cut
duration.cut = cut(duration, breaks, right=FALSE)

# Compute the frequencies of eruptions in each sub-interval
?table
duration.freq = table(duration.cut)

# Compute the relative frequencies
    duration.relfreq =
        duration.freq / nrow(faithful)

# Compute cumulative frequencies
    ?cumsum
    cumsum(duration.freq)
```

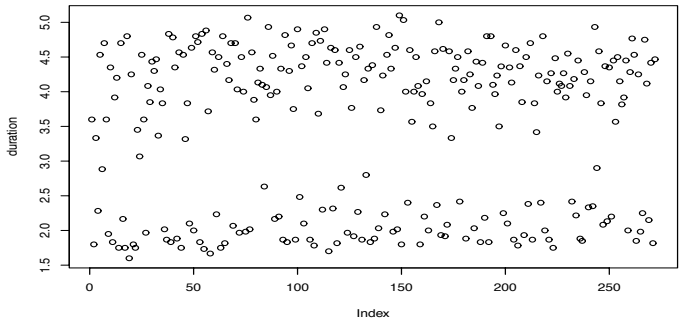
History of commands (Session 2)

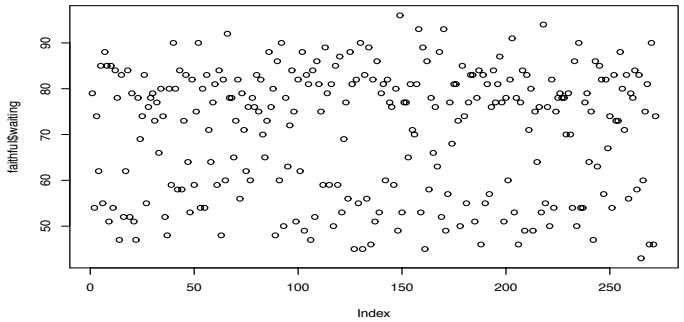
Plot analysis results

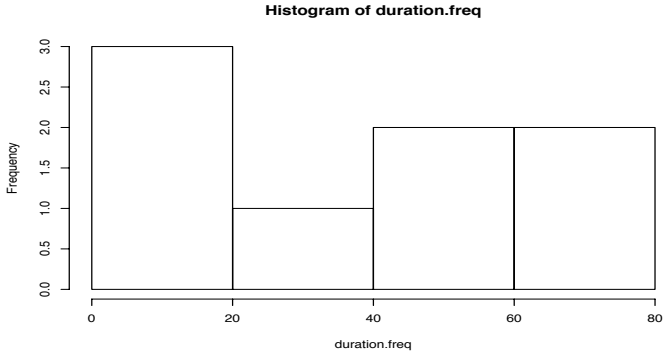
```
# commands used for plotting
# generic function to plot R objects
        ?plot;
        Cumfreq0 <- c(0,cumsum(duration.freq))
        plot(breaks,Cumfreq0)
#connect data points with lines
        ?lines;
        lines(breaks, Cumfreq0)
#draw a frequency distribution histogram
        ?hist
        hist(duration.freq)
# fit linear model to the data (use to carry out
regression)
        ?lm
        lm(faithful$waiting~duration)
#add straight line throw the current plot
        ?abline
        abline(lm(faithful$waiting~duration))
```

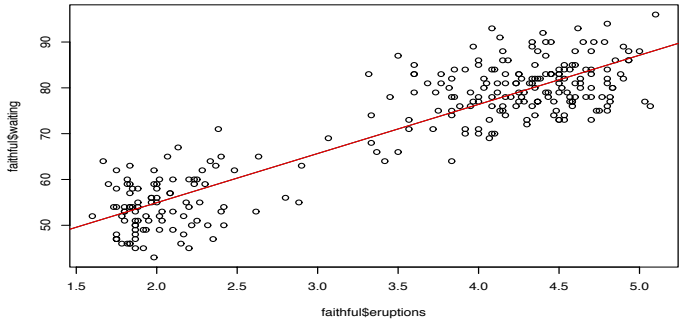
```
# generate a scatterplot
        plot(faithful$eruptions, faithful$waiting)

# save resulting scatter plot on the disk
        ?png        ?jpeg        ?pdf

# open a graphical device
        png(filename="my_plot.png")
        plot(faithful$eruptions, faithful$waiting)
        abline(lm(faithful$waiting~duration))
        deff.off() # don't forget to close!

# save current plot
        plot(faithful$eruptions, faithful$waiting)
        abline(lm(faithful$waiting~duration))
        dev.copy(jpeg,filename="my_plot.jpg");
        dev.off ();
```

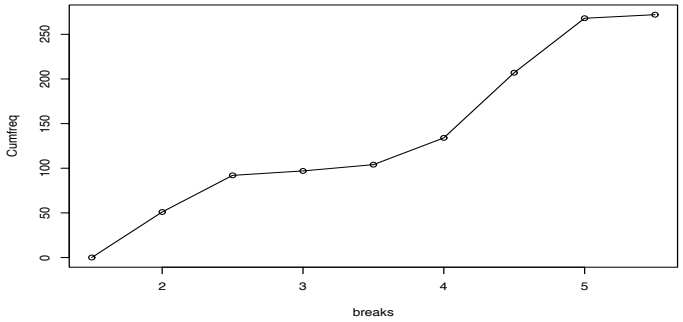
Frequency distribution of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.











History of commands (Session 2)

Test data for normality

```
# generate binomial data
        ?rnorm
        measurements  <-length(duration)
        r_data<-rnorm(measurements)

# Perform Shapiro-Wilk test of normality
        shapiro.test(random_data)
        >W = 0.9969, p-value = 0.884
        shapiro.test(duration)
        >W = 0.8459, p-value = 9.036e-16
```

NOTE. *The null-hypothesis for a data normality test is: "the data is not normally distributed". Low P-value means – null-hypothesis is correct and, therefore, data is NOT normally distributed. High P-value shows that we can't say that the data is not normally distributed. One should use as well QQ-plots/density plots to visually confirm results of the Shairo-Wilk or similar tests.*

Have a look on the data

```
# draw several plots on same figures
(details during session 3)
        par(mfrow=c(3,2))
# generate plots for random and
experimental data
        plot(r_data)
        plot(duration)
# generate density plots
        plot(density(r_data))
        plot(density(duration))
# draw normal QQ plots with QQ lines
        qqnorm(r_data)
        qqline(r_data, col=2)
        qqnorm(duration)
        qqline(duration, col=2)
# save image as vector image
dev.copy2eps(file="normality_test.eps")
dev.off()
```

Test Old Faithful geyser eruption data for normality