Max Planck Institute
of Immunobiology, Freiburg

Introduction to R

http://www.r-project.org/

Thomas Manke

Start and Stop

Start	> R
Stop	> q()

From R to Linux

1. Try the following R commands and understand their output

   - err=system("pwd")
   - pwd=getwd()
   - rf=list.files("/")
   - nd=setwd("~")
   - hf=list.files(".")

Getting help

?, ??	> ?matrix
examples	> example(rnorm)
web	http://www.r-project.org/ http://www.bioconductor.org/ http://epicenter/courses/2010.11.11_linux/materials/
Mailing lists	e.g. r-help@r-project.org
Packages	> vignette("affy")

When seeking help from the community, it is a good idea to send also the output of
> sessionInfo()

A better pocket calculator

- 2+2
- 2^2
- sin(pi/2)
- sqrt(2)
- factorial(20)
- 1+exp(-1i*pi)

Max Planck Institute
of Immunobiology, Freiburg

Data and Data Types

Thomas Manke

Assignments and Removal

- x=2
- y<-2
- 2->z
- rm(z)
- ls()
- rm(list=ls())
- ls()

Conversions

- x=2
- typeof(x)
- as.character(x)
- as.logical(x)

R-objects are made from atomic elements:
logical, numeric, character, complex

Creating a vector

```
> x = c(1,2,3,4,5,6,7,8,9,10)
> x = seq(1,10,by=1)
> x = 1:10
```

What is the difference between a=1:10-1 and b=1:(10-1) ?

Create a reversed vector (10,9,…,1)

“letters” and “LETTERS” are predefined vectors.
Of which type? Give an interpretation of >as.logical(LETTERS)

Naming vector elements

Vector elements can be named for convenience.

- x=c(a=1,b=2); str(x)
- X = 1:100; str(X)
- names(X)=letters; str(X)
- X[1]
- X["a"]
- names(X)[100]="This is the end"
- Y=as.characters(X); str(Y)

Combining vectors

- x=c(1,2,3); y=c(2,2,2);
- x>y # comparsion
- x+y # addition
- x*y # element-wise multiplication
- x %*% y # scalar-product

Vector elements are recycled

Vector Mystic

Execute the following and try to understand

- 1:8 + c(10,20)
- x=1:100; i=30:70; x[i]; x[-i]
- x[x %% 2 == 0]

Matrices

Definition:

- M=1:100; str(M); dim(M)
- dim(M)=c(20,5); str(M); dim(M)
- M=matrix(1:100,ncol=5)
- M=matrix(1:110,nrow=20)

Matrices

Access:

- M[20,1]; M[1,20]
- M[c(1,2),]; M[,c(1,4)]
- M[25]

Query:

> dim(M); nrow(M); ncol(M)

Matrices

Manipulation:

- t(M)
- cbind(M,1:5)
- rbind(M,1:5)

Naming:

- rownames(M)=letters[1:nrow(M)]
- colnames(M)=LETTERS[1:ncol(M)]

Data Frames

Definition:

- D=data.frame(x=1:9,l=letters[1:9], b=1:9<5)
- str(D)

Access:

- D[,1]; D$x

Data frames are generalized matrices where different columns can have different types.

Lists

Definition:

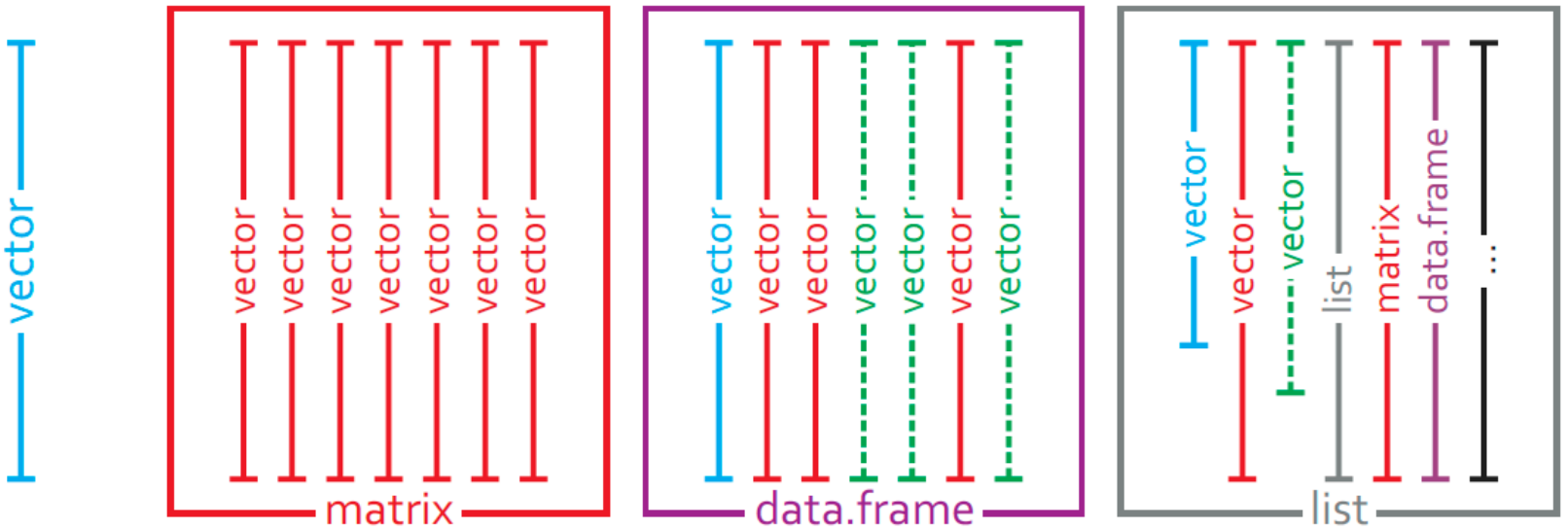
- L=list(x=1:9,l=letters[1:3], b=1:20<10)
- str(L)

Access:

- L[,1]; L[1], L$x
- str(L[c(2,3)]), str(L[[1]])

Lists are generalized data frame where different columns can have different length.

Summary: Data Types



Data type	single	single	multiple	multiple
elements	atomic	atomic	vector	any
subsets	X[i] X[“a”]	M[i,j] M[“row”,”col”]	D[i,j] D$”colname”	L[[i]] L$”colname”

i and j can be index vectors!

from Michael Stadler (FMI)

Max Planck Institute
of Immunobiology, Freiburg

Thomas Manke

Information on objects

Available objects	➢ls() ➢objects()
Information on objects	➢typeof(x) ➢class(x) ➢length(x) ➢str(x)

R has precompiles data sets that are suitable for exploration > data()

Exercises

Vectors and matrices. Create the following:

1. Create a vector v1 with number 1 to 200 in steps of 2.
2. Create a vector v2 with 100 with alternating TRUE/FALSE
3. Convert vector v1 into a 10x10 matrix M
4. Name the rows of M (a,b,..j) and the columns (A,B,…J)
5. Take the transpose of M

Data frames and lists. Create

1. a data frame D with column V that takes the vector v1 and logical vector which labels whether v1 is smaller or bigger than 100
2. A list L of four elements (a=v1, b=v2, c=M, d=D)

Access. Try and understand those three commands

which(L[[2]]); sum(L[[3]]["a",]); L[[4]]$v1

Max Planck Institute
of Immunobiology, Freiburg

Data Descriptions

Thomas Manke

Creating synthetic data

Sample from normal distribution

- X=rnorm(100,mean=2,sd=0.1)
- X
- plot(X)

R supports many different distributions.
Try to find some of them ?r<tab>

Plots

- X=1:100
- Y=X+rnorm(100,mean=0,sd=1)
- plot(X,Y)

The plot command is very customizable (labels, titles, etc). See ?plot for details.

Descriptive statistics

values	➢max(x) ➢mean(x) ➢var(x) ➢sd(x) ➢median(x) ➢summary(x)
plots	➢hist(x) ➢stem(x) ➢boxplot(x) ➢plot(density(x))

Exercise

- x=rnorm(300,mean=c(1,2,1),sd=c(0.1,0.1,1))

Of what type is x, how many elements does it have?

- v=sum((x-mean(x))^2)/(length(x)-1)

Identify all functions in this expression.

What is being calculated?

What is the maximum/minimum element of x?

Show the summary statistics?

What is the mean and standard variation

What is the distribution of X?

Generate a boxplot of X and try to customize it.

Histograms

>Hist(x); Hist(x,breaks=100); hist(x,breaks=100,prob=TRUE)

>lines(density(x),col="red")

- hist(x[seq(1,300,3)],xlim=c(0.5,2.5),30)
- hist(x[seq(2,300,3)],xlim=c(0.5,2.5),30,add=TRUE,col="red")

Replace title and the x-label with your favorite words.

Plot the histogram of a Poisson distribution with mean=16

Overlay it with the density plot of a normal distribution with mean=16 and standard deviation=4

Efficency with Data Frames

skipped

Generate a factor

- F=rep(letters[1:3],100)

Generate data frame

- D=data.frame(X=x,F=f)
- summary(D)
- tapply(DX, DF, mean)

  Describe the effect of this function.

- boxplot(D$X ~ D$F, col=c("red","green","blue"))

Converting Data Frames

skipped

- L=tapply(DX, DF, c)

  What does this function do?

  What data type is L?
- A=cbind(a=L$a,b=L$b,c=L$c)

  What data type is A and what is its dimension?

More images

- A=cbind(a=rnorm(100,1,0.1), b=rnorm(100,2,0.1), c=rnorm(100,1,1))
- image(A)
- image(t(A), axes=FALSE)

Use ?image to find out how to change the colours.

What is the default? Try with 100 topo.colors.

Correlations

- plot(A[,1],A[,3])
- cor(A[,1],A[,3])
- pairs(A)
- cor(A)

Dendrograms

- A = cbind(A, d=A[,1]+rnorm(100,0,0.2))
- pairs(A)
- cor(A)
- d=dist(t(A)); plot(hclust(d))

See ?dist and ?hclust to find out about other possible definitions of distance and clustering methods

Correlations

Plot is not sufficient for dense data

- n=10000
- x1=matrix(rnorm(n), ncol=2)
- x2=matrix(rnorm(n, mean=4, sd=1.2), ncol=2)
- x=rbind(x1,x2)
- smoothScatter(x)

Heatmaps

- heatmap
- heatmap(A)
- heatmap(A,labRow="",scale="none")

As with all plot commands, there are many ways to customize heatmaps. There is even a separate package "heatmap.2"

Saving images

- jpeg(“heatmap.jpg”)
- heatmap(A,labRow=“”,scale=“none”)
- dev.off() # close device

The device “jpeg” can be customized.
See ?jpeg for details.

Max Planck Institute
of Immunobiology, Freiburg

Thomas Manke

Max Planck Institute
of Immunobiology, Freiburg

Retrieving external data

Thomas Manke

Reading files

R-files (objects)	> load(file)
tables	> read.tables(file, … options …)
scan	> scan(files, … options …)

There are many different ways to read data. Try

- ?read<tab>

The reality of I/O

Getting data into R can be difficult and time-consuming.

Therefore:
Keep you data files simple and structured!!!

A real-world example

Loading ChIP-chip data (yeast; Harbinson et al. 2004)

Repeat the steps below and watch out for errors

1. D= read.table("ChIP-chip.tab")
2. D= read.table("ChIP-chip.tab", skip=1)
3. D= read.table("ChIP-chip.tab", skip=1, header=TRUE)
4. D= read.table("ChIP-chip.tab", skip=1, header=TRUE, sep="\t")
5. D= read.table("ChIP-chip.tab", skip=1, header=TRUE, sep="\t", comment.char="")
6. D= read.table("ChIP-chip.tab", skip=1, header=TRUE, sep="\t", comment.char="", quote="\"")
7. D= read.table("ChIP-chip.tab", skip=1, header=TRUE, sep="\t", comment.char="", quote="\"",nrows=6229)

Saving files and data

R-files (objects)	> save(objects, file)
tables	> write.tables(object,file, … options …)

There are many different ways to write data. Try

- ?write<tab>
- ?save<tab>

Binary R-files are a very compact, convenient and efficient way to share results and data

Loading and saving

- rm(list=ls()) # clear all objects
- load(url("http://epicenter/courses/2010.11.11_linux/data/ChIP-chip.Rdata"))

Which objects were loaded? Of which type?
Display their structure.
Interprete the output of summary(D)
Create a scatter plot for D$FHL1_YPD and D$RAP1_YPD

Convert data frame to matrix

- A = as.matrix(cbind(D[,4:207]))
- rownames(A)=D$X

Save objects

- save(D,A,file="mydata.Rdata")

Removing missing values

Undefined or missing values is a common problem with real data. The can be identified using is.na()

What is the output of x=is.na(D$FHL1_YPD)?
What is !x ? What is which(x)?
Compare D$FHL1_YPD[x] and D$FHL1_YPD[!x]

Repeat the above for y=is.na(D$RAP1_YPD).

Given an interpretation of

- I = intersect(which(x), which(y))
- U= union(which(x),which(y))

Thresholding

Convert matrix A into binary matrix B

- A[is.na(A)]=1
- B = A<1e-3

Create a contingency matrix of FHL1 and RAP1

- table(B[,"FHL1_YPD"],B[,"RAP1_YPD"])

Question: is this overlap large?

Venn Diagrams

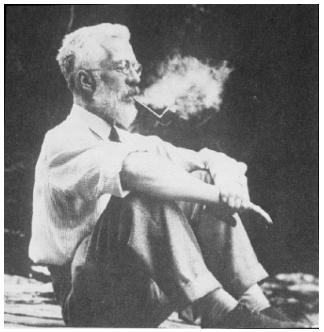
- library(gplots)
- t=1E-3
- L=list(FHL1=which(D$FHL1_YPD<t),
  RAP1=which(D$RAP1_YPD<t),
  MCM1=which(D$MCM1_YPD<t))
- venn(L)

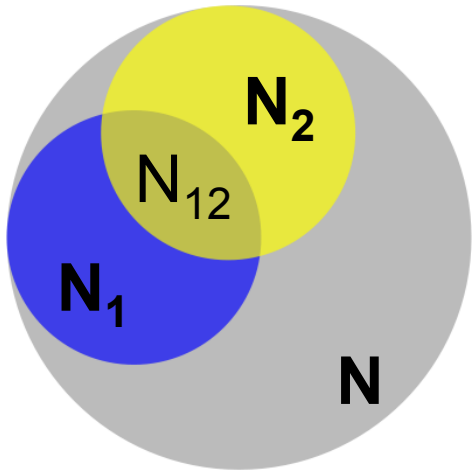
Compare the plot with the contingency table.

Fisher Exact Test

	+	-	
+	a	b	N_1
-	c	d	
	N_2		N

2x2 table





Venn diagram

Probability that overlap > N_{12}

➢ phyper(q=N_{12}-1,m=N_2,n=N-N_2,k=N_1,lower.tail=FALSE)

Is the overlap between FHL1 and RAP1 significant?

The final exercise

- rm(list=ls()) # clear all objects
- load(url("http://epicenter/courses/2010.11.11_linux/data/expression.Rdata"))

Which objects were loaded? Of which type?
Display their structure.

Create a data subset for the top 100 genes

Create an image of the data.

Create a heatmap of the same.

Not covered

- Bioconductor (biobase package)
- Scripts and logical control
- User-defined functions
- Plot-configurations and parameters
- Distribution function
- String functions and regular expression
- Matrix algebra, inversions, etc …
- models