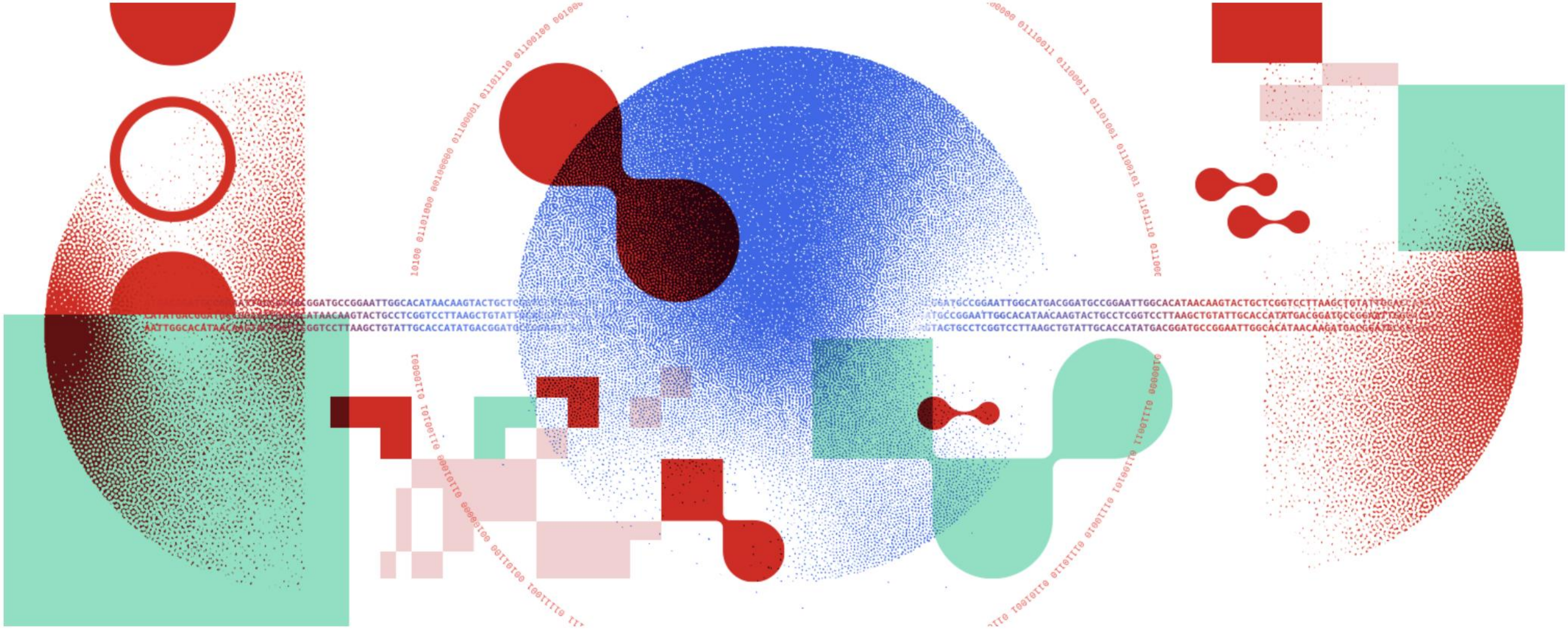


Introduction to Statistics and Data Visualisation with R

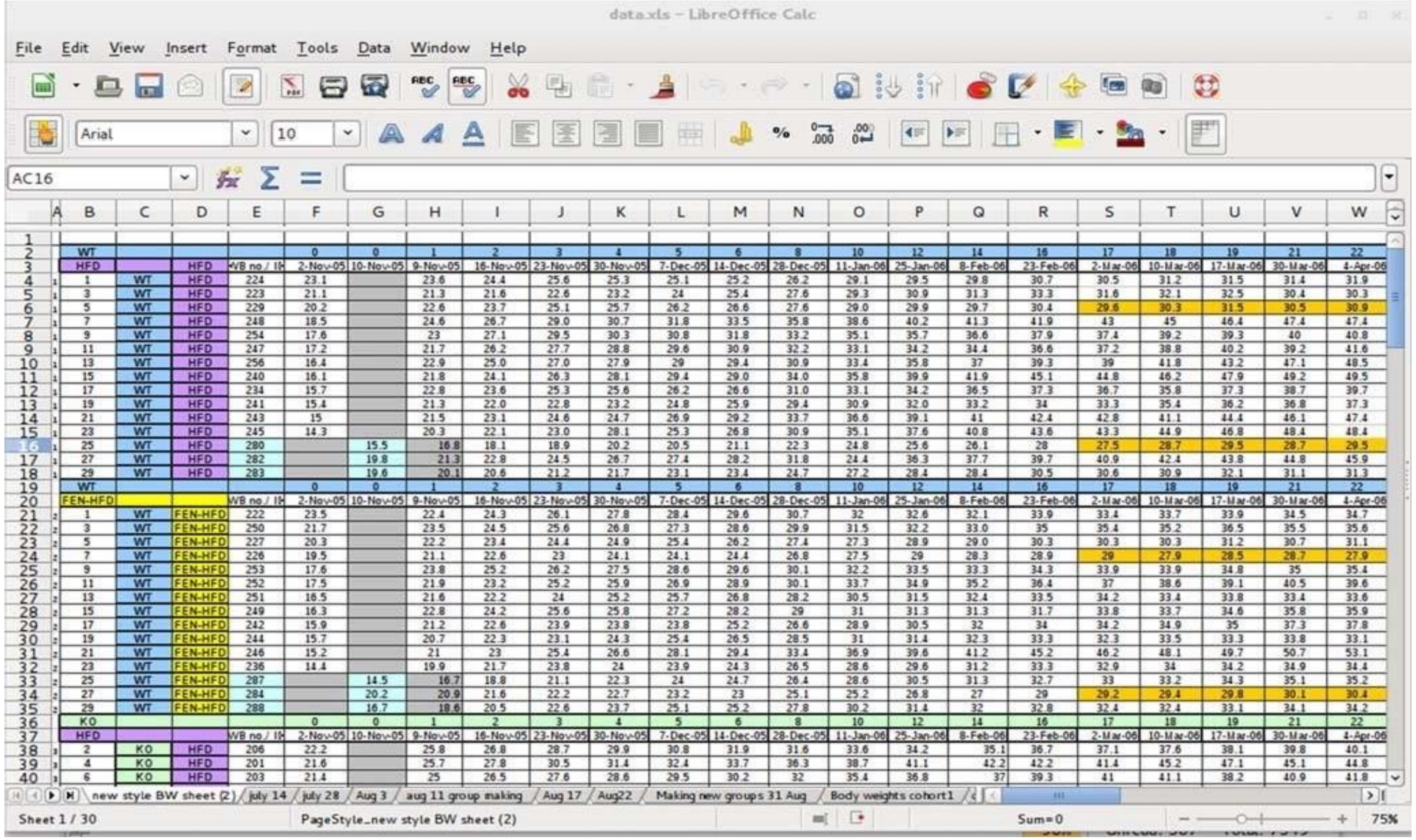
Lausanne, January 2026

Joao Lourenço and Rachel Marcone

Introduction to R

Prepare: make data available in a specific format

- Database
- Flat file
- Proprietary file



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1																							
2		WT				0	0	1	2	3	4	5	6	8	10	12	14	16	17	18	19	21	22
3		HFD		HFD	WB no / IH	2-Nov-05	10-Nov-05	9-Nov-05	16-Nov-05	23-Nov-05	30-Nov-05	7-Dec-05	14-Dec-05	28-Dec-05	11-Jan-06	25-Jan-06	8-Feb-06	23-Feb-06	2-Mar-06	10-Mar-06	17-Mar-06	30-Mar-06	4-Apr-06
4	1	1	WT	HFD	224	23.1		23.6	24.4	25.6	25.3	25.1	25.2	26.2	29.1	29.5	29.8	30.7	30.5	31.2	31.5	31.4	31.9
5	1	3	WT	HFD	223	21.1		21.3	21.6	22.6	23.2	24	25.4	27.6	29.3	30.9	31.3	33.3	31.6	32.1	32.5	30.4	30.3
6	1	5	WT	HFD	229	20.2		22.6	23.7	25.1	25.7	26.2	26.6	27.6	29.0	29.9	29.7	30.4	29.6	30.3	31.5	30.5	30.9
7	1	7	WT	HFD	248	18.5		24.6	26.7	29.0	30.7	31.8	33.5	35.8	38.6	40.2	41.3	41.9	43	45	46.4	47.4	47.4
8	1	9	WT	HFD	254	17.6		23	27.1	29.5	30.3	30.8	31.8	33.2	35.1	35.7	36.6	37.9	37.4	39.2	39.3	40	40.8
9	1	11	WT	HFD	247	17.2		21.7	26.2	27.7	28.8	29.6	30.9	32.2	33.1	34.2	34.4	36.6	37.2	38.8	40.2	39.2	41.6
10	1	13	WT	HFD	256	16.4		22.9	25.0	27.0	27.9	29	29.4	30.9	33.4	35.8	37	39.3	39	41.8	43.2	47.1	48.5
11	1	15	WT	HFD	240	16.1		21.8	24.1	26.3	28.1	29.4	29.0	34.0	35.8	39.9	41.9	45.1	44.8	46.2	47.9	49.2	49.5
12	1	17	WT	HFD	234	15.7		22.8	23.6	25.3	25.6	26.2	26.6	31.0	33.1	34.2	36.5	37.3	36.7	35.8	37.3	38.7	39.7
13	1	19	WT	HFD	241	15.4		21.3	22.0	22.8	23.2	24.8	25.9	29.4	30.9	32.0	33.2	34	33.3	35.4	36.2	36.8	37.3
14	1	21	WT	HFD	243	15		21.5	23.1	24.6	24.7	26.9	29.2	33.7	36.6	39.1	41	42.4	42.8	41.1	44.4	46.1	47.4
15	1	23	WT	HFD	245	14.3		20.3	22.1	23.0	28.1	25.3	26.8	30.9	35.1	37.6	40.8	43.6	43.3	44.9	46.8	48.4	48.4
16	1	25	WT	HFD	280		15.5	16.8	18.1	18.9	20.2	20.5	21.1	22.3	24.8	25.6	26.1	28	27.5	28.7	29.5	28.7	29.5
17	1	27	WT	HFD	282		19.8	21.3	22.8	24.5	26.7	27.4	28.2	31.8	24.4	36.3	37.7	39.7	40.9	42.4	43.8	44.8	45.9
18	1	29	WT	HFD	283		19.6	20.1	20.6	21.2	21.7	23.1	23.4	24.7	27.2	28.4	28.4	30.5	30.6	30.9	32.1	31.1	31.3
19		WT				0	0	1	2	3	4	5	6	8	10	12	14	16	17	18	19	21	22
20		FEN-HFD			WB no / IH	2-Nov-05	10-Nov-05	9-Nov-05	16-Nov-05	23-Nov-05	30-Nov-05	7-Dec-05	14-Dec-05	28-Dec-05	11-Jan-06	25-Jan-06	8-Feb-06	23-Feb-06	2-Mar-06	10-Mar-06	17-Mar-06	30-Mar-06	4-Apr-06
21	2	1	WT	FEN-HFD	222	23.5		22.4	24.3	26.1	27.8	28.4	29.6	30.7	32	32.6	32.1	33.9	33.4	33.7	33.9	34.5	34.7
22	2	3	WT	FEN-HFD	250	21.7		23.5	24.5	25.6	26.8	27.3	28.6	29.9	31.5	32.2	33.0	35	35.4	35.2	36.5	35.5	35.6
23	2	5	WT	FEN-HFD	227	20.3		22.2	23.4	24.4	24.9	25.4	26.2	27.4	27.3	28.9	29.0	30.3	30.3	30.3	31.2	30.7	31.1
24	2	7	WT	FEN-HFD	226	19.5		21.1	22.6	23	24.1	24.1	24.4	26.8	27.5	29	28.3	28.9	29	27.9	28.5	28.7	27.9
25	2	9	WT	FEN-HFD	253	17.6		23.8	25.2	26.2	27.5	28.6	29.6	30.1	32.2	33.5	33.3	34.3	33.9	33.9	34.8	35	35.4
26	2	11	WT	FEN-HFD	252	17.5		21.9	23.2	25.2	25.9	26.9	28.9	30.1	33.7	34.9	35.2	36.4	37	38.6	39.1	40.5	39.6
27	2	13	WT	FEN-HFD	251	16.5		21.6	22.2	24	25.2	25.7	26.8	28.2	30.5	31.5	32.4	33.5	34.2	33.4	33.8	33.4	33.6
28	2	15	WT	FEN-HFD	249	16.3		22.8	24.2	25.6	25.8	27.2	28.2	29	31	31.3	31.3	31.7	33.8	33.7	34.6	35.8	35.9
29	2	17	WT	FEN-HFD	242	15.9		21.2	22.6	23.9	23.8	23.8	25.2	26.6	28.9	30.5	32	34	34.2	34.9	35	37.3	37.8
30	2	19	WT	FEN-HFD	244	15.7		20.7	22.3	23.1	24.3	25.4	26.5	28.5	31	31.4	32.3	33.3	32.3	33.5	33.3	33.8	33.1
31	2	21	WT	FEN-HFD	246	15.2		21	23	25.4	26.6	28.1	29.4	33.4	36.9	39.6	41.2	45.2	46.2	48.1	49.7	50.7	53.1
32	2	23	WT	FEN-HFD	236	14.4		19.9	21.7	23.8	24	23.9	24.3	26.5	28.6	29.6	31.2	33.3	32.9	34	34.2	34.9	34.4
33	2	25	WT	FEN-HFD	287		14.5	16.7	18.8	21.1	22.3	24	24.7	26.4	28.6	30.5	31.3	32.7	33	33.2	34.3	35.1	35.2
34	2	27	WT	FEN-HFD	284		20.2	20.9	21.6	22.2	22.7	23.2	23	25.1	25.2	26.8	27	29	29.2	29.4	29.8	30.1	30.4
35	2	29	WT	FEN-HFD	288		16.7	18.6	20.5	22.6	23.7	25.1	25.2	27.8	30.2	31.4	32	32.8	32.4	32.4	33.1	34.1	34.2
36		KO				0	0	1	2	3	4	5	6	8	10	12	14	16	17	18	19	21	22
37		HFD			WB no / IH	2-Nov-05	10-Nov-05	9-Nov-05	16-Nov-05	23-Nov-05	30-Nov-05	7-Dec-05	14-Dec-05	28-Dec-05	11-Jan-06	25-Jan-06	8-Feb-06	23-Feb-06	2-Mar-06	10-Mar-06	17-Mar-06	30-Mar-06	4-Apr-06
38	3	2	KO	HFD	206	22.2		25.8	26.8	28.7	29.9	30.8	31.9	31.6	33.6	34.2	35.1	36.7	37.1	37.6	38.1	39.8	40.1
39	3	4	KO	HFD	201	21.6		25.7	27.8	30.5	31.4	32.4	33.7	36.3	38.7	41.1	42.2	42.2	41.4	45.2	47.1	45.1	44.8
40	3	6	KO	HFD	203	21.4		25	26.5	27.6	28.6	29.5	30.2	32	35.4	36.8	37	39.3	41	41.1	38.2	40.9	41.8

Which tool to use for data analysis ?

Spreadsheets

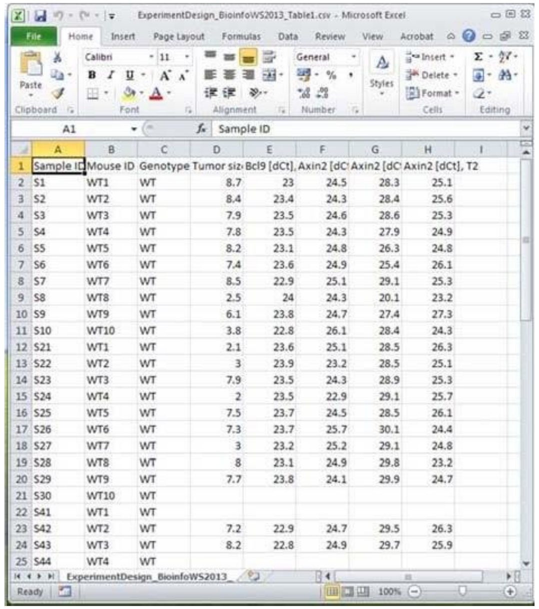
Statistical packages

Programming languages

Microsoft Office Excel
LibreOffice
The Document Foundation
gnumeric
Minitab 15
KaleidaGraph
GraphPad
PRISM 7
R
python™

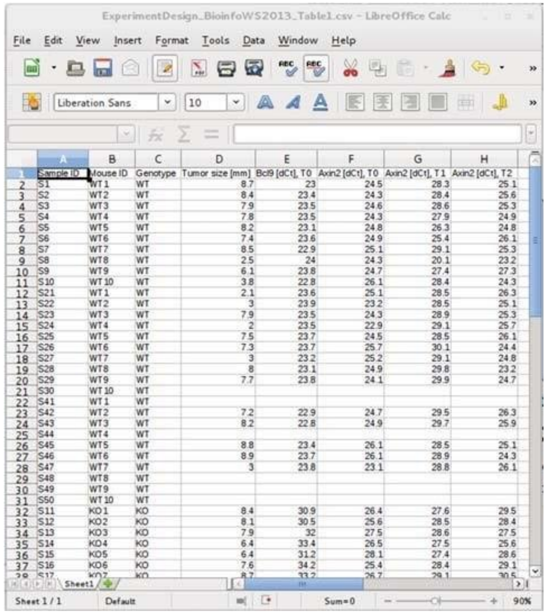
Annoyances with spreadsheets

Microsoft Excel



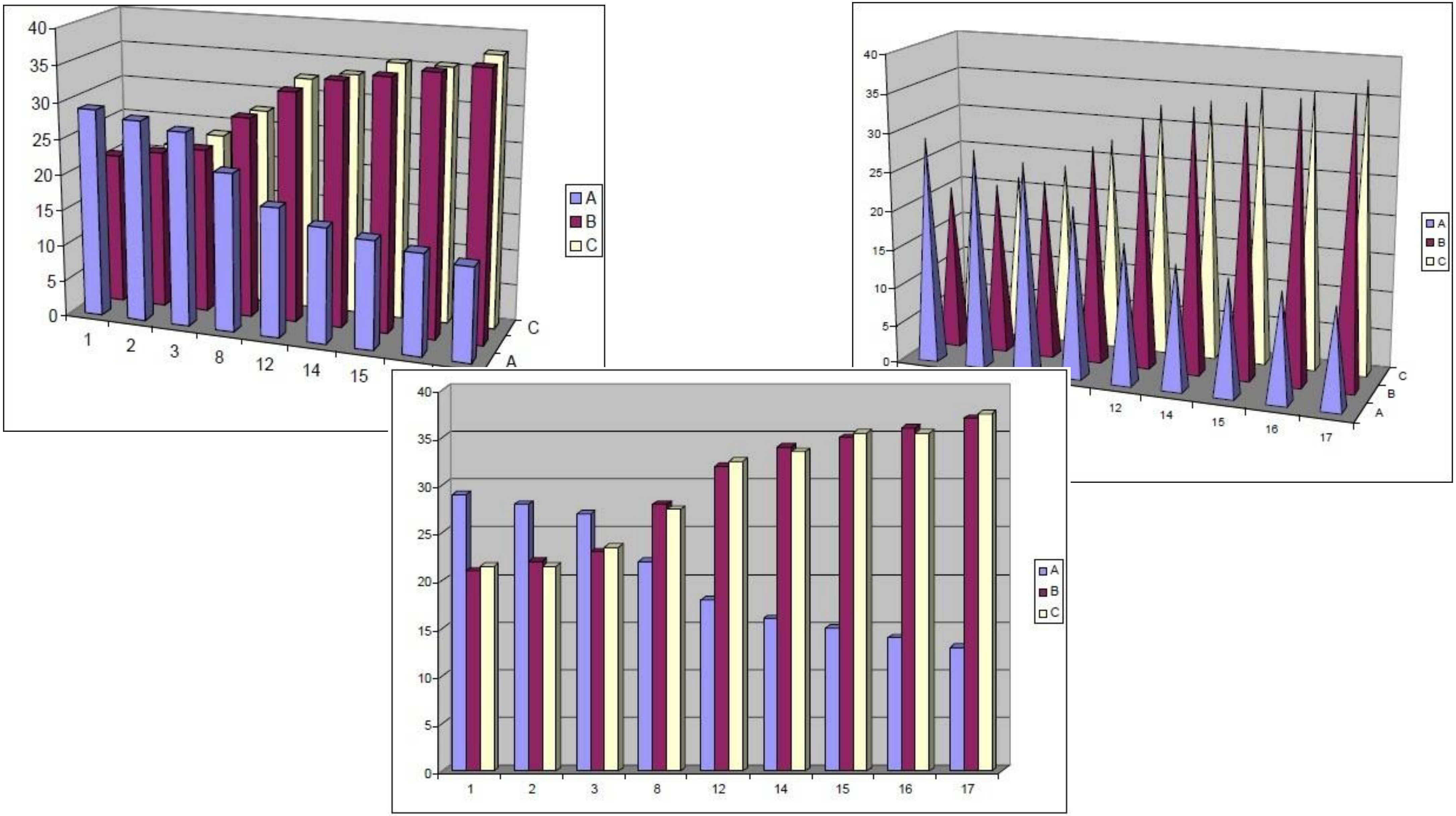
	A	B	C	D	E	F	G	H	I
1	Sample ID	Mouse ID	Genotype	Tumor siz	Bcl9 [dCt],	Axin2 [dC	Axin2 [dC	Axin2 [dCt], T2	
2	S1	WT1	WT	8.7	23	24.5	28.3	25.1	
3	S2	WT2	WT	8.4	23.4	24.3	28.4	25.6	
4	S3	WT3	WT	7.9	23.5	24.6	28.6	25.3	
5	S4	WT4	WT	7.8	23.5	24.3	27.9	24.9	
6	S5	WT5	WT	8.2	23.1	24.8	26.3	24.8	
7	S6	WT6	WT	7.4	23.6	24.9	25.4	26.1	
8	S7	WT7	WT	8.5	22.9	25.1	29.1	25.3	
9	S8	WT8	WT	2.5	24	24.3	20.1	23.2	
10	S9	WT9	WT	6.1	23.8	24.7	27.4	27.3	
11	S10	WT10	WT	3.8	22.8	26.1	28.4	24.3	
12	S21	WT1	WT	2.1	23.6	25.1	28.5	26.3	
13	S22	WT2	WT	3	23.9	23.2	28.5	25.1	
14	S23	WT3	WT	7.9	23.5	24.3	28.9	25.3	
15	S24	WT4	WT	2	23.5	22.9	29.1	25.7	
16	S25	WT5	WT	7.5	23.7	24.5	28.5	26.1	
17	S26	WT6	WT	7.3	23.7	25.7	30.1	24.4	
18	S27	WT7	WT	3	23.2	25.2	29.1	24.8	
19	S28	WT8	WT	8	23.1	24.9	29.8	23.2	
20	S29	WT9	WT	7.7	23.8	24.1	29.9	24.7	
21	S30	WT10	WT						
22	S41	WT1	WT						
23	S42	WT2	WT	7.2	22.9	24.7	29.5	26.3	
24	S43	WT3	WT	8.2	22.8	24.9	29.7	25.9	
25	S44	WT4	WT						

Libreoffice

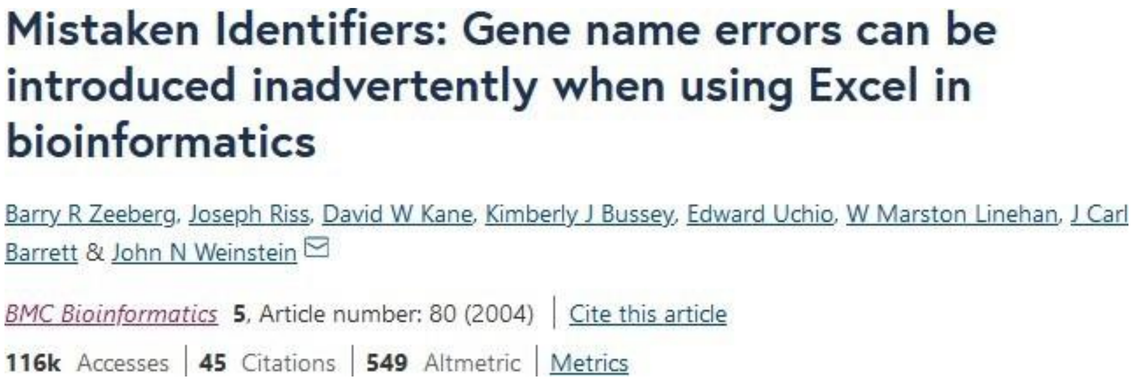


	A	B	C	D	E	F	G	H
1	Sample ID	Mouse ID	Genotype	Tumor size [mm]	Bcl9 [dCt], T0	Axin2 [dCt], T0	Axin2 [dCt], T1	Axin2 [dCt], T2
2	S1	WT 1	WT	8.7	23	24.5	28.3	25.1
3	S2	WT 2	WT	8.4	23.4	24.3	28.4	25.6
4	S3	WT 3	WT	7.9	23.5	24.6	28.6	25.3
5	S4	WT 4	WT	7.8	23.5	24.3	27.9	24.9
6	S5	WT 5	WT	8.2	23.1	24.8	26.3	24.8
7	S6	WT 6	WT	7.4	23.6	24.9	25.4	26.1
8	S7	WT 7	WT	8.5	22.9	25.1	29.1	25.3
9	S8	WT 8	WT	2.5	24	24.3	20.1	23.2
10	S9	WT 9	WT	6.1	23.8	24.7	27.4	27.3
11	S10	WT 10	WT	3.8	22.8	26.1	28.4	24.3
12	S21	WT 1	WT	2.1	23.6	25.1	28.5	26.3
13	S22	WT 2	WT	3	23.9	23.2	28.5	25.1
14	S23	WT 3	WT	7.9	23.5	24.3	28.9	25.3
15	S24	WT 4	WT	2	23.5	22.9	29.1	25.7
16	S25	WT 5	WT	7.5	23.7	24.5	28.5	26.1
17	S26	WT 6	WT	7.3	23.7	25.7	30.1	24.4
18	S27	WT 7	WT	3	23.2	25.2	29.1	24.8
19	S28	WT 8	WT	8	23.1	24.9	29.8	23.2
20	S29	WT 9	WT	7.7	23.8	24.1	29.9	24.7
21	S30	WT 10	WT					
22	S41	WT 1	WT					
23	S42	WT 2	WT	7.2	22.9	24.7	29.5	26.3
24	S43	WT 3	WT	8.2	22.8	24.9	29.7	25.9
25	S44	WT 4	WT					
26	S45	WT 5	WT	8.8	23.4	26.1	28.5	25.1
27	S46	WT 6	WT	8.9	23.7	26.1	28.9	24.3
28	S47	WT 7	WT	3	23.8	23.1	28.8	26.1
29	S48	WT 8	WT					
30	S49	WT 9	WT					
31	S50	WT 10	WT					
32	S11	KO 1	KO	8.4	30.9	26.4	27.6	29.5
33	S12	KO 2	KO	8.1	30.5	25.6	28.5	28.4
34	S13	KO 3	KO	7.9	32	27.5	28.6	27.5
35	S14	KO 4	KO	6.4	33.4	26.5	27.5	25.6
36	S15	KO 5	KO	6.4	31.2	28.1	27.4	28.6
37	S16	KO 6	KO	7.6	34.2	25.4	28.4	29.1
38	S17	KO 7	KO	8.7	33.2	26.7	29.1	30.5

- •Many standard methods in statistics are not available. Other methods only offer basic options (linear regression)
- • Different analysis require user to reorganize the data
- •Probably ok for simple calculations (basic summary statistics, simple regression)
- •Add-ons can be used for missing functions (e.g. StatPlus for Excel)
- • Many types of graphics violate standards of good graphics


40
35
30
25
20
15
10
5
0
1
2
3
8
12
14
15
16
17
A
B
C

Annoyances with spreadsheets

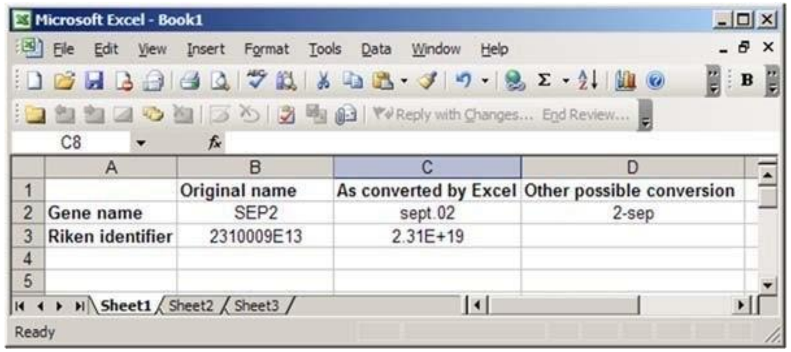


Mistaken Identifiers: Gene name errors can be introduced inadvertently when using Excel in bioinformatics

Barry R Zeeberg, Joseph Riss, David W Kane, Kimberly J Bussey, Edward Uchio, W Marston Linehan, J Carl Barrett & John N Weinstein

BMC Bioinformatics **5**, Article number: 80 (2004) | Cite this article

116k Accesses | **45** Citations | **549** Altmetric | Metrics

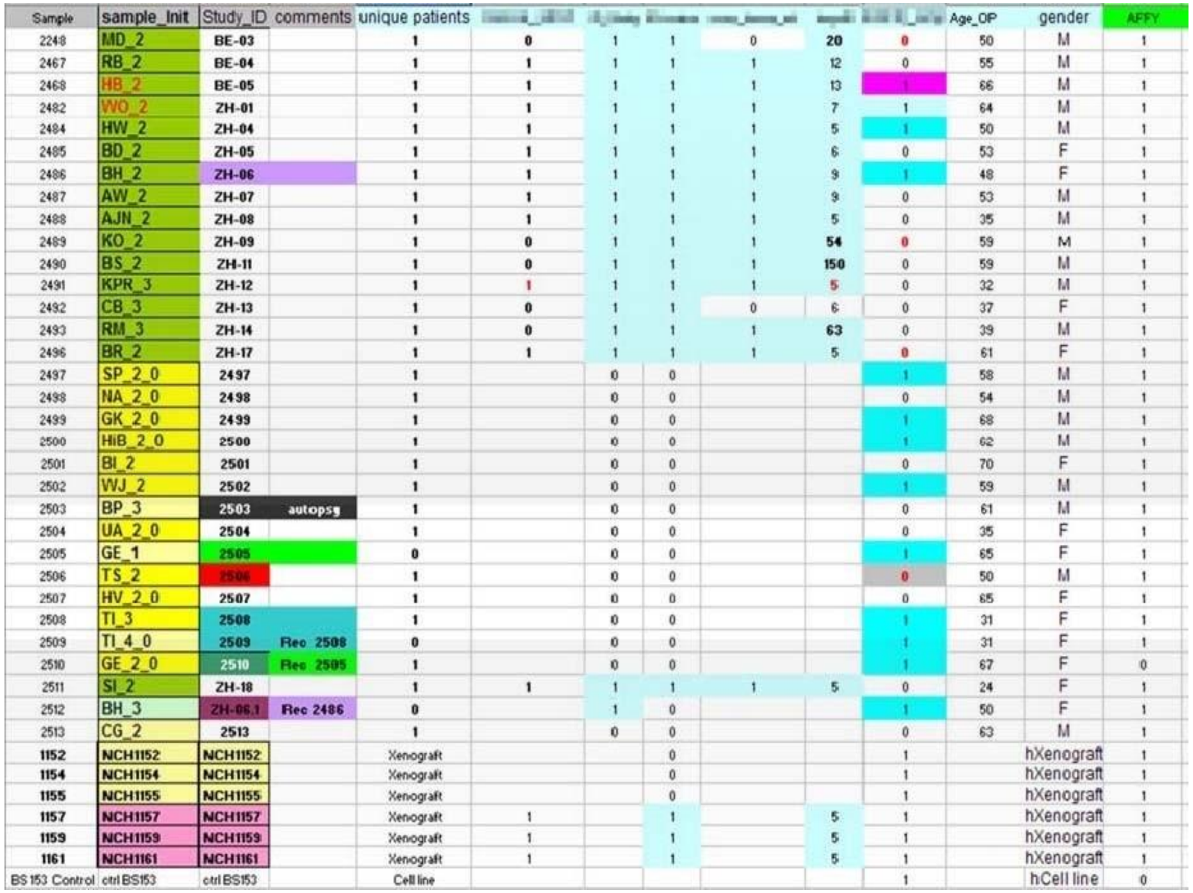


Microsoft Excel - Book1
File Edit View Insert Format Tools Data Window Help
Reply with Changes... End Review...
C8
fx
Sheet1 Sheet2 Sheet3
Ready

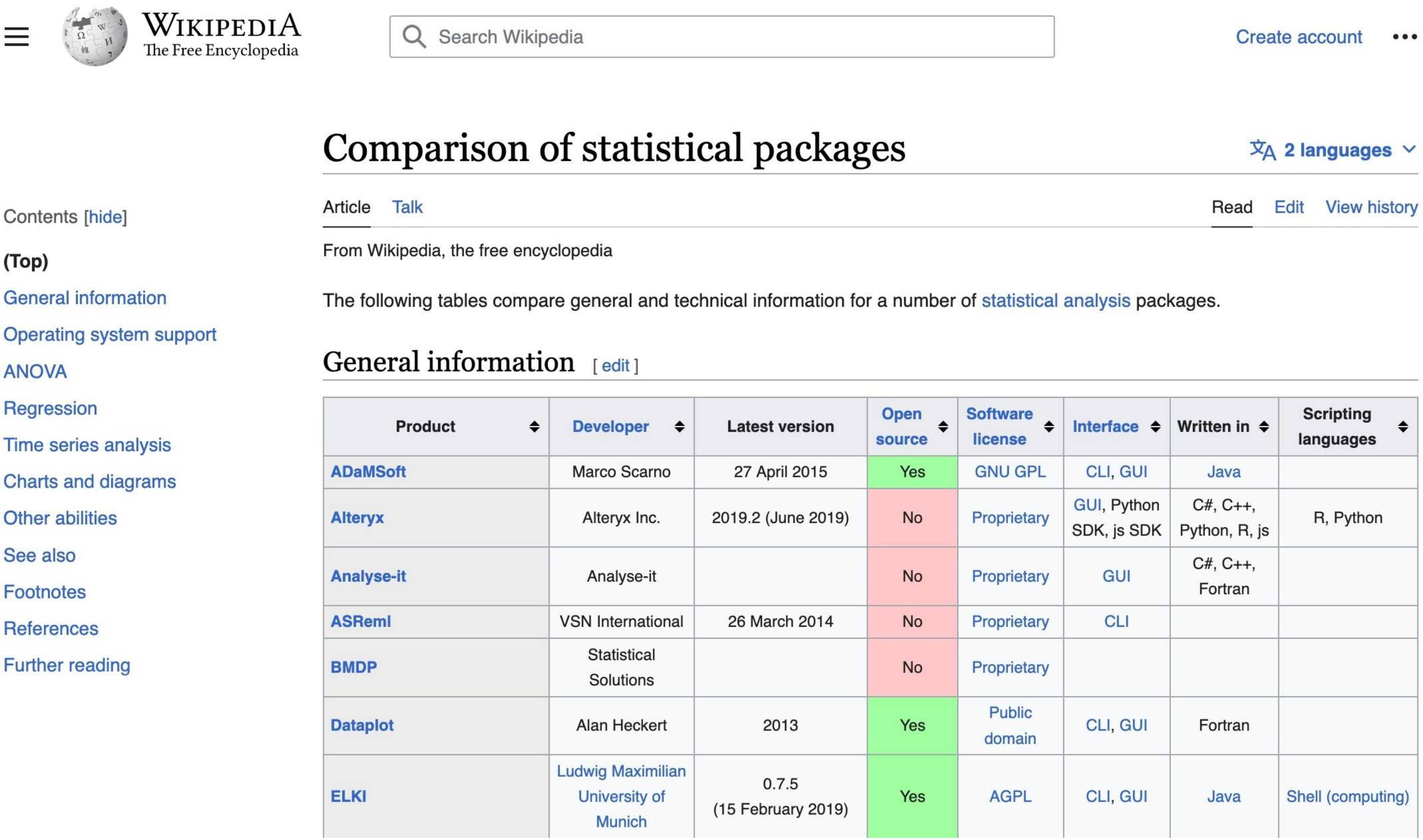
	A	B	C	D
1		Original name	As converted by Excel	Other possible conversion
2	Gene name	SEP2	sept.02	2-sep
3	Riken identifier	2310009E13	2.31E+19	
4				
5				

"The date conversions affect at least 30 gene names; the floating-point conversions affect at least 2,000 if Riken identifiers are included. These conversions are irreversible; the original gene names cannot be recovered."

Example of a dataset which is difficult to use with any statistical program



Sample	sample_Init	Study_ID	comments	unique patients	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	Age_OP	gender	AFFY
2248	MD_2	BE-03		1	0	1	1	0	20	0	50	M	1
2467	RB_2	BE-04		1	1	1	1	1	12	0	55	M	1
2468	HB_2	BE-05		1	1	1	1	1	13	1	66	M	1
2482	WO_2	ZH-01		1	1	1	1	1	7	1	64	M	1
2484	HW_2	ZH-04		1	1	1	1	1	5	1	50	M	1
2485	BD_2	ZH-05		1	1	1	1	1	6	0	53	F	1
2486	BH_2	ZH-06		1	1	1	1	1	9	1	48	F	1
2487	AW_2	ZH-07		1	1	1	1	1	9	0	53	M	1
2488	AJN_2	ZH-08		1	1	1	1	1	5	0	35	M	1
2489	KO_2	ZH-09		1	0	1	1	1	54	0	59	M	1
2490	BS_2	ZH-11		1	0	1	1	1	150	0	59	M	1
2491	KPR_3	ZH-12		1	1	1	1	1	5	0	32	M	1
2492	CB_3	ZH-13		1	0	1	1	0	6	0	37	F	1
2493	RM_3	ZH-14		1	0	1	1	1	63	0	39	M	1
2496	BR_2	ZH-17		1	1	1	1	1	5	0	61	F	1
2497	SP_2_0	2497		1		0	0			1	58	M	1
2498	NA_2_0	2498		1		0	0			0	54	M	1
2499	GK_2_0	2499		1		0	0			1	68	M	1
2500	HiB_2_0	2500		1		0	0			1	62	M	1
2501	BI_2	2501		1		0	0			0	70	F	1
2502	WJ_2	2502		1		0	0			1	59	M	1
2503	BP_3	2503	autopsy	1		0	0			0	61	M	1
2504	UA_2_0	2504		1		0	0			0	35	F	1
2505	GE_1	2505		0		0	0			1	65	F	1
2506	TS_2	2506		1		0	0			0	50	M	1
2507	HV_2_0	2507		1		0	0			0	65	F	1
2508	TI_3	2508		1		0	0			1	31	F	1
2509	TI_4_0	2509	Rec 2508	0		0	0			1	31	F	1
2510	GE_2_0	2510	Rec 2505	1		0	0			1	67	F	0
2511	SI_2	ZH-18		1	1	1	1	1	5	0	24	F	1
2512	BH_3	ZH-06.1	Rec 2486	0		1	0			1	50	F	1
2513	CG_2	2513		1		0	0			0	63	M	1
1152	NCH1152	NCH1152		Xenograft			0			1		hXenograft	1
1154	NCH1154	NCH1154		Xenograft			0			1		hXenograft	1
1155	NCH1155	NCH1155		Xenograft			0			1		hXenograft	1
1157	NCH1157	NCH1157		Xenograft	1		1		5	1		hXenograft	1
1159	NCH1159	NCH1159		Xenograft	1		1		5	1		hXenograft	1
1161	NCH1161	NCH1161		Xenograft	1		1		5	1		hXenograft	1
BS 153 Control	ctrl BS153	ctrl BS153		Cell line						1		hCell line	0



Comparison of statistical packages

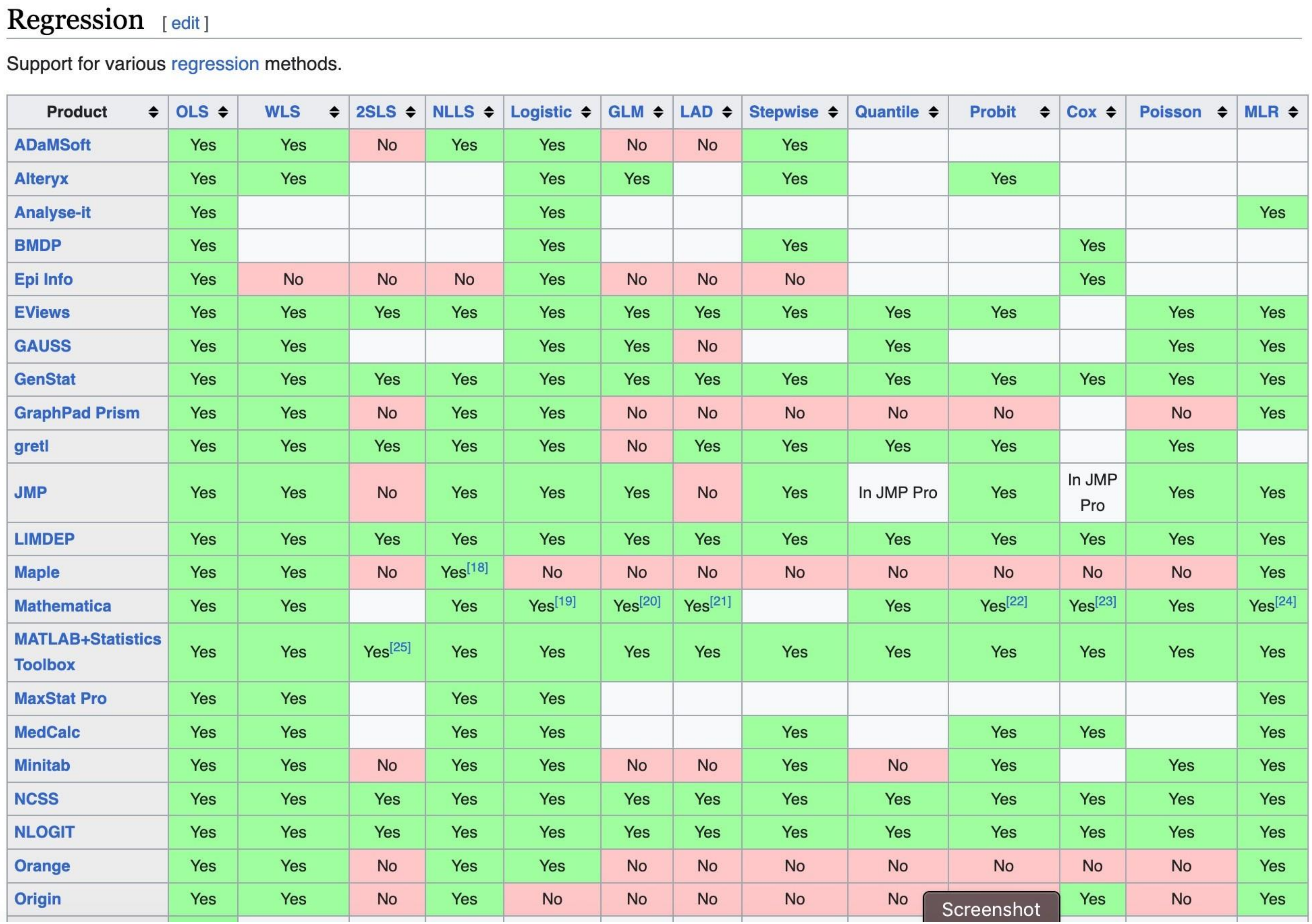
From Wikipedia, the free encyclopedia

The following tables compare general and technical information for a number of statistical analysis packages.

General information [edit]

Product	Developer	Latest version	Open source	Software license	Interface	Written in	Scripting languages
ADaMSoft	Marco Scarno	27 April 2015	Yes	GNU GPL	CLI, GUI	Java	
Alteryx	Alteryx Inc.	2019.2 (June 2019)	No	Proprietary	GUI, Python SDK, js SDK	C#, C++, Python, R, js	R, Python
Analyse-it	Analyse-it		No	Proprietary	GUI	C#, C++, Fortran	
ASReml	VSN International	26 March 2014	No	Proprietary	CLI		
BMDP	Statistical Solutions		No	Proprietary			
Dataplot	Alan Heckert	2013	Yes	Public domain	CLI, GUI	Fortran	
ELKI	Ludwig Maximilian University of Munich	0.7.5 (15 February 2019)	Yes	AGPL	CLI, GUI	Java	Shell (computing)

https://en.wikipedia.org/wiki/Comparison_of_statistical_packages



Regression [edit]

Support for various regression methods.

Product	OLS	WLS	2SLS	NLLS	Logistic	GLM	LAD	Stepwise	Quantile	Probit	Cox	Poisson	MLR
ADaMSoft	Yes	Yes	No	Yes	Yes	No	No	Yes					
Alteryx	Yes	Yes			Yes	Yes		Yes		Yes			
Analyse-it	Yes				Yes								Yes
BMDP	Yes				Yes			Yes			Yes		
Epi Info	Yes	No	No	No	Yes	No	No	No			Yes		
EViews	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes		Yes	Yes
GAUSS	Yes	Yes			Yes	Yes	No		Yes			Yes	Yes
GenStat	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
GraphPad Prism	Yes	Yes	No	Yes	Yes	No	No	No	No	No		No	Yes
gretl	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes		Yes	
JMP	Yes	Yes	No	Yes	Yes	Yes	No	Yes	In JMP Pro	Yes	In JMP Pro	Yes	Yes
LIMDEP	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Maple	Yes	Yes	No	Yes[18]	No	No	No	No	No	No	No	No	Yes
Mathematica	Yes	Yes		Yes	Yes[19]	Yes[20]	Yes[21]		Yes	Yes[22]	Yes[23]	Yes	Yes[24]
MATLAB+Statistics Toolbox	Yes	Yes	Yes[25]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
MaxStat Pro	Yes	Yes		Yes	Yes								Yes
MedCalc	Yes	Yes		Yes	Yes			Yes		Yes	Yes		Yes
Minitab	Yes	Yes	No	Yes	Yes	No	No	Yes	No	Yes		Yes	Yes
NCSS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
NLOGIT	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Orange	Yes	Yes	No	Yes	Yes	No	No	No	No	No	No	No	Yes
Origin	Yes	Yes	No	Yes	No	No	No	No	No	Screenshot	Yes	No	Yes

What is R ?

- R is an open source complete and flexible software environment for statistical computing and graphics.
- It includes :
  - Tools for data import and manipulation
  - Large set of data analysis tools
  - Graphical tools
  - As a programming language, a simple development environment, with a text editor
- R itself is written primarily in C and Fortran, and is an implementation of the statistical language S

Advantages of R

- Advantages of R
  - Free
  - Availability and compatibility
  - Well-designed publication-quality plots
  - Tons of graphic possibilities
  - Can import files from other (statistical) programs
  - New version every x months
  - Interactive development environments (IDEs) available
  - Large users community

- Advantages of *learning* R
  - Learn to program and do reproducible research
  - Speak the common language

Drawbacks of R

- «Expert friendly»
- Learn by example
- Not very (easily) interactive
- Command-based
- Documentation sometimes cryptic

- (Too) large amount of resources
- Constantly evolving
- Memory intensive and slow at times

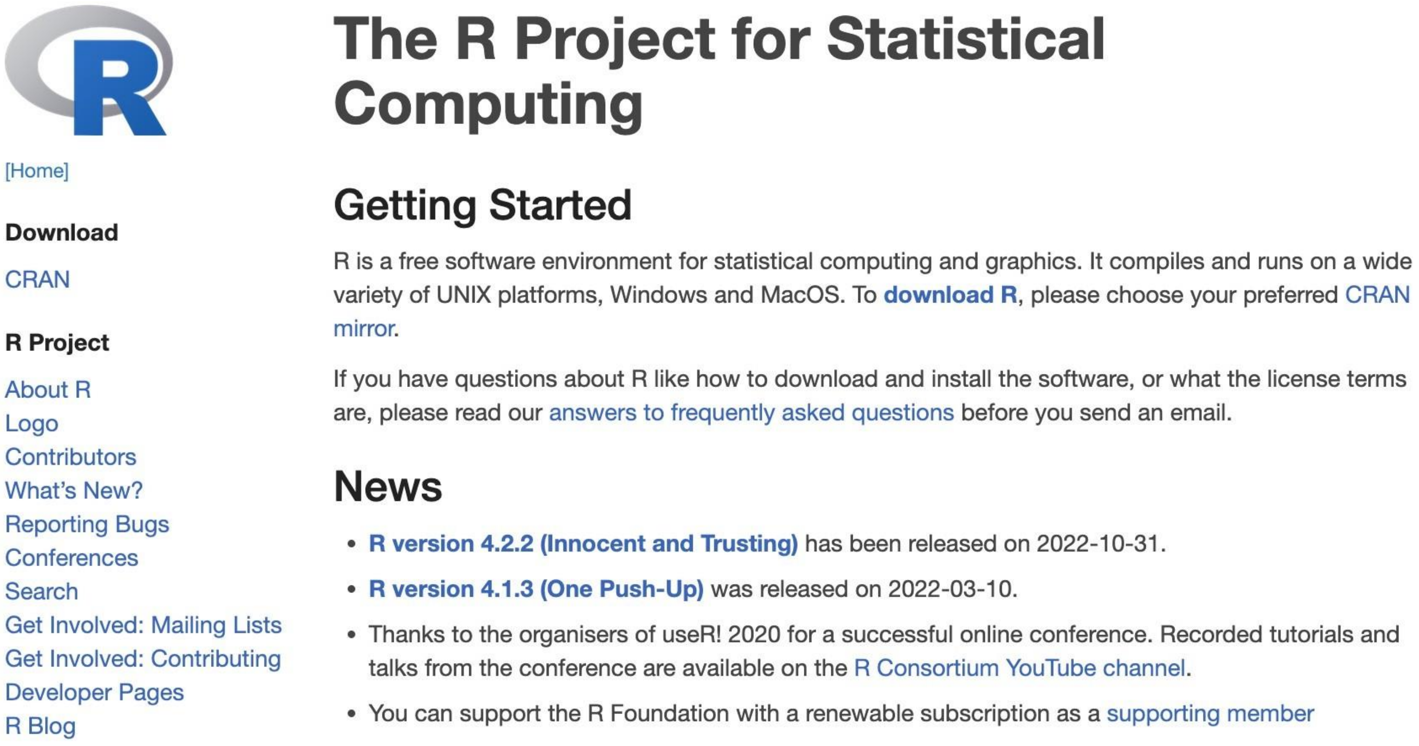
Now we open R

Go to website
Day 1 (https://sib-swiss.github.io/Introduction-to-statistics-with-R/day1/)

Click on the Download full data for the week button

Open the file easy_R_script.R file, which we will now look at together !

Downloading and installing R: the R website



[Home]

Download

CRAN

R Project

About R
Logo
Contributors
What's New?
Reporting Bugs
Conferences
Search
Get Involved: Mailing Lists
Get Involved: Contributing
Developer Pages
R Blog

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred CRAN mirror.

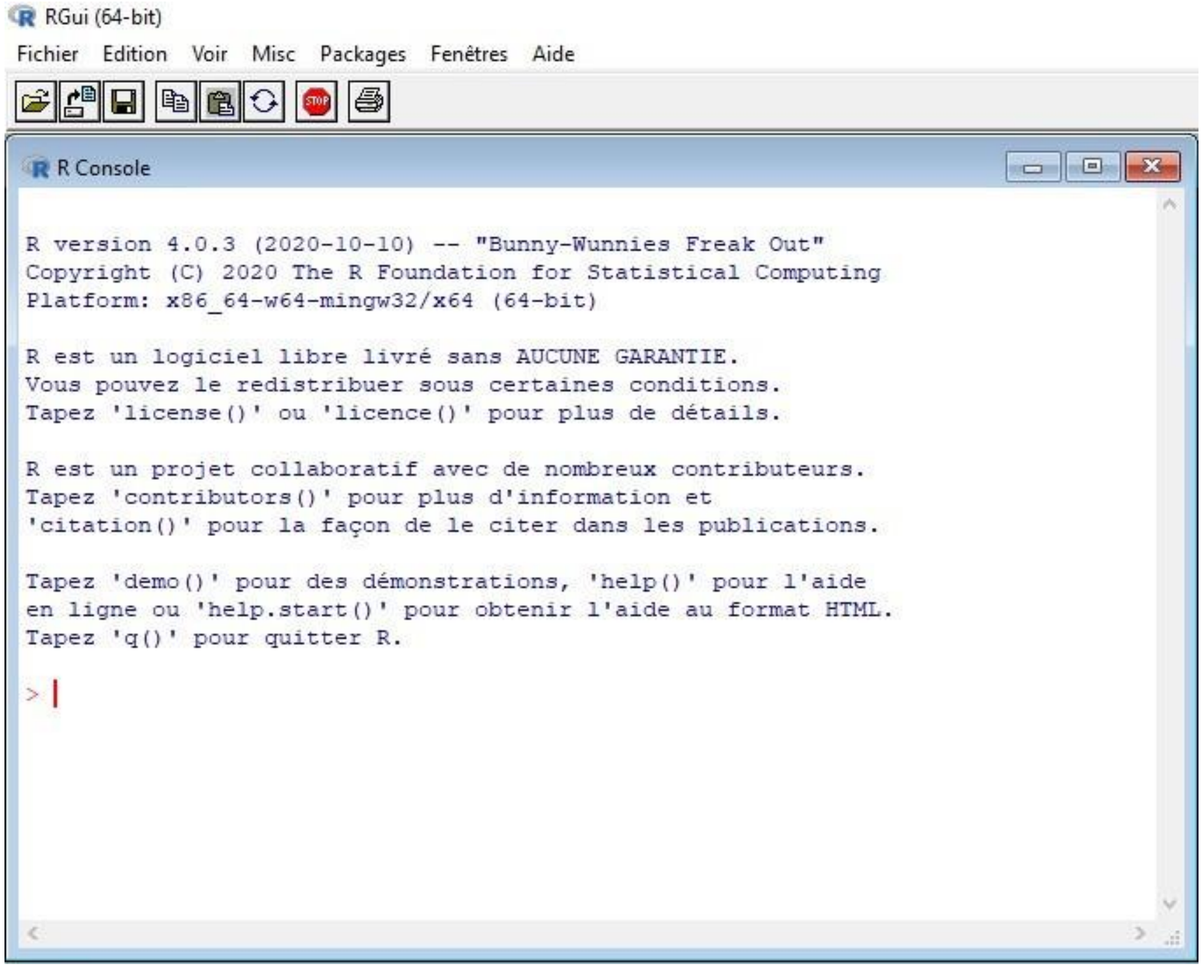
If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

News

- **R version 4.2.2 (Innocent and Trusting)** has been released on 2022-10-31.
- **R version 4.1.3 (One Push-Up)** was released on 2022-03-10.
- Thanks to the organisers of useR! 2020 for a successful online conference. Recorded tutorials and talks from the conference are available on the R Consortium YouTube channel.
- You can support the R Foundation with a renewable subscription as a supporting member

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

R console



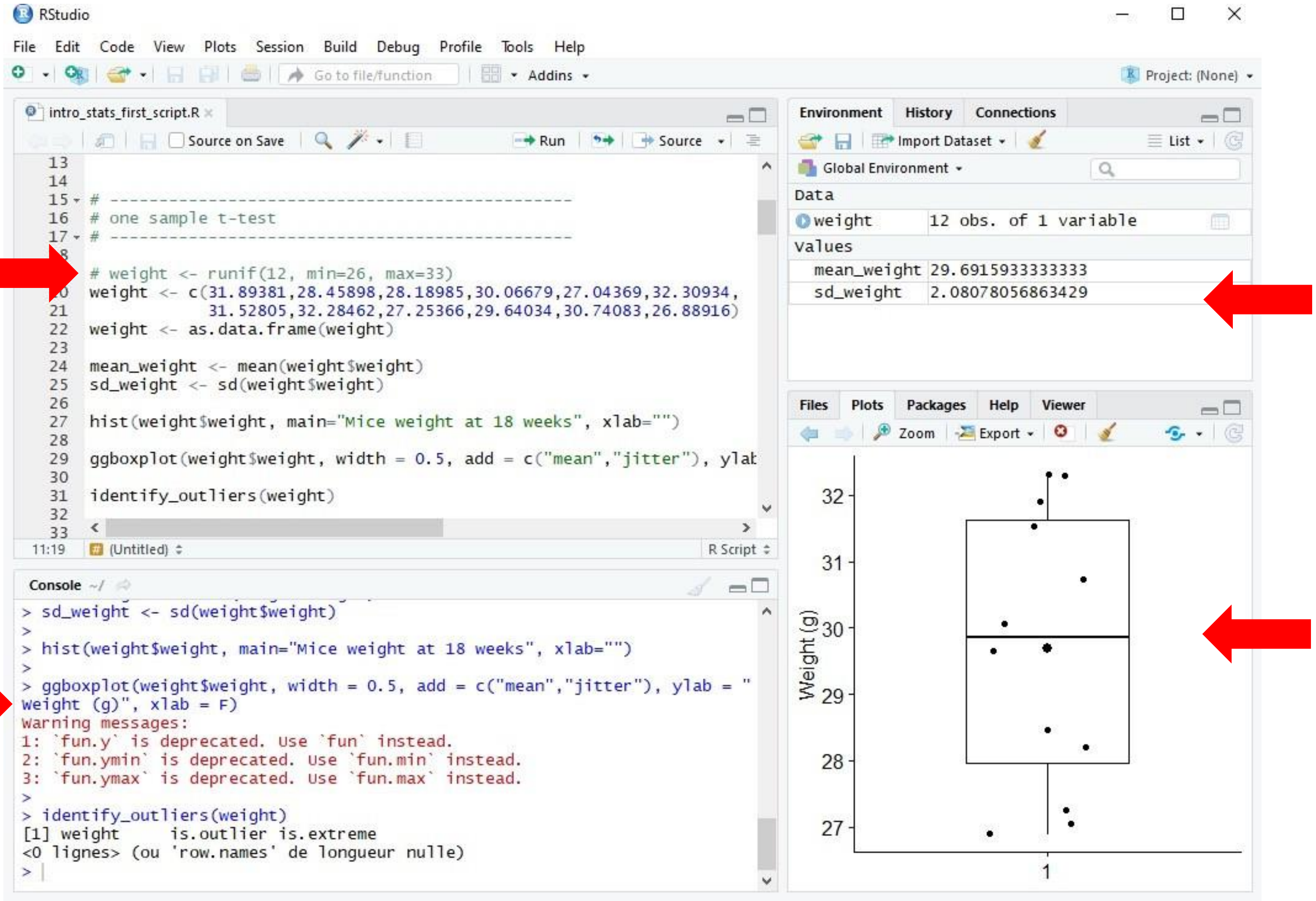
```
R version 4.0.3 (2020-10-10) -- "Bunny-Wunnies Freak Out"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

> |
```

The prompt “>” indicates that R is waiting for you to type a command

RStudio interface

Editor

Console, terminal



```r
# ------------------------------------------------
# one sample t-test
# ------------------------------------------------

# weight <- runif(12, min=26, max=33)
weight <- c(31.89381,28.45898,28.18985,30.06679,27.04369,32.30934,
            31.52805,32.28462,27.25366,29.64034,30.74083,26.88916)
weight <- as.data.frame(weight)

mean_weight <- mean(weight$weight)
sd_weight <- sd(weight$weight)

hist(weight$weight, main="Mice weight at 18 weeks", xlab="")

ggboxplot(weight$weight, width = 0.5, add = c("mean","jitter"), ylab

identify_outliers(weight)
```

```
> sd_weight <- sd(weight$weight)
>
> hist(weight$weight, main="Mice weight at 18 weeks", xlab="")
>
> ggboxplot(weight$weight, width = 0.5, add = c("mean","jitter"), ylab = "
weight (g)", xlab = F)
Warning messages:
1: `fun.y` is deprecated. Use `fun` instead.
2: `fun.ymin` is deprecated. Use `fun.min` instead.
3: `fun.ymax` is deprecated. Use `fun.max` instead.
>
> identify_outliers(weight)
[1] weight     is.outlier is.extreme
<0 lignes> (ou 'row.names' de longueur nulle)
> 
```

Data

weight	12 obs. of 1 variable

Values

mean_weight	29.6915933333333
sd_weight	2.08078056863429

Workspace, history

File explorer, plots, packages, help

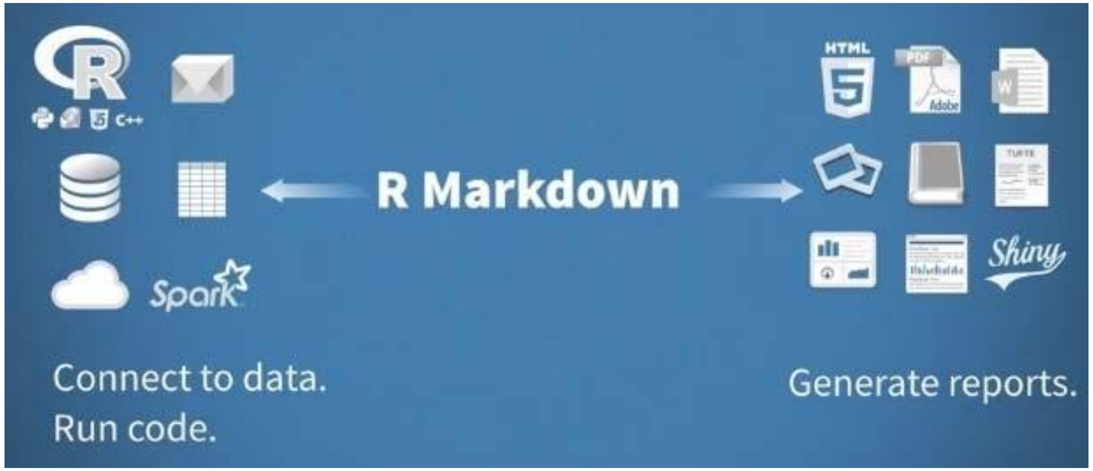
R scripts and workspace

- R script (.R file)
  - Very useful instead of typing commands on the console.
  - Allows you to keep track of what you are doing and make any modification easier
  - To actually execute some commands, you can select the lines and run the execution

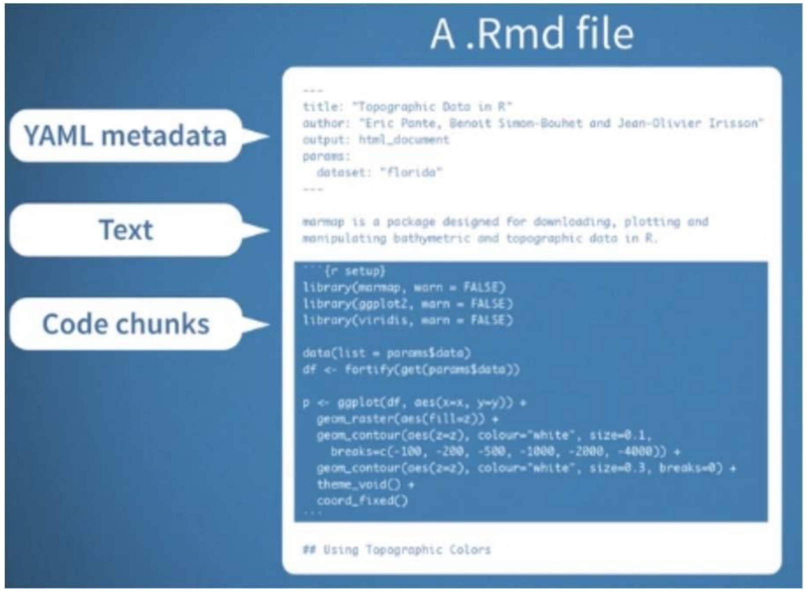
- Workspace (.Rdata file)
  - The internal memory where R will store the objects you created during the session.
  - To list what is in your workspace: `ls()`
  - To empty the workspace from all objects: `rm(list=ls())`
  - To save only specific R objects: `save(object_name(s),"name_of_file.RData")`
  - To save your entire workspace: `save.image("name_of_file.RData")`
  - To load your workspace / specific R objects: `load("name_of_file.RData")`

R Markdown

- R Markdown provides an authoring framework for data science. You can use a single R Markdown file to both:
    - save and execute code
    - generate high quality reports that can be shared with an audience
- R Markdown documents are fully reproducible and support dozens of static and dynamic output formats



https://rmarkdown.rstudio.com/lesson-1.html



~~~
---
title: "Topographic Data in R"
author: "Eric Pante, Benoit Simon-Bouhet and Jean-Olivier Irisson"
output: html_document
params:
  dataset: "florida"
---

marmap is a package designed for downloading, plotting and
manipulating bathymetric and topographic data in R.

```{r setup}
library(marmap, warn = FALSE)
library(ggplot2, warn = FALSE)
library(viridis, warn = FALSE)

data(list = params$data)
df <- fortify(get(params$data))

p <- ggplot(df, aes(x=x, y=y)) +
  geom_raster(aes(fill=z)) +
  geom_contour(aes(z=z), colour="white", size=0.1,
    breaks=c(-100, -200, -500, -1000, -2000, -4000)) +
  geom_contour(aes(z=z), colour="white", size=0.3, breaks=0) +
  theme_void() +
  coord_fixed()
```

## Using Topographic Colors
~~~

Leaving R

- To leave R, use the q()command (or "quit" from the menu in RStudio):

```
> q()
Save workspace image? [y/n/c]:
```

Answers:

y save workspace image

n **don't save workspace image**

c cancel quitting

Functions, operators and variables

```
CIhigh <- mean(x) + 1.96*sd(x)/sqrt(n)
```

Variables: objects stored in memory

Functions: always followed by parenthesis

Operators

R syntax

- Case sensitive: A is not a
- Variable names can include A-Z, a-z, 0-9, ... but can not start with a number
- Commands can be separated by ; or newline

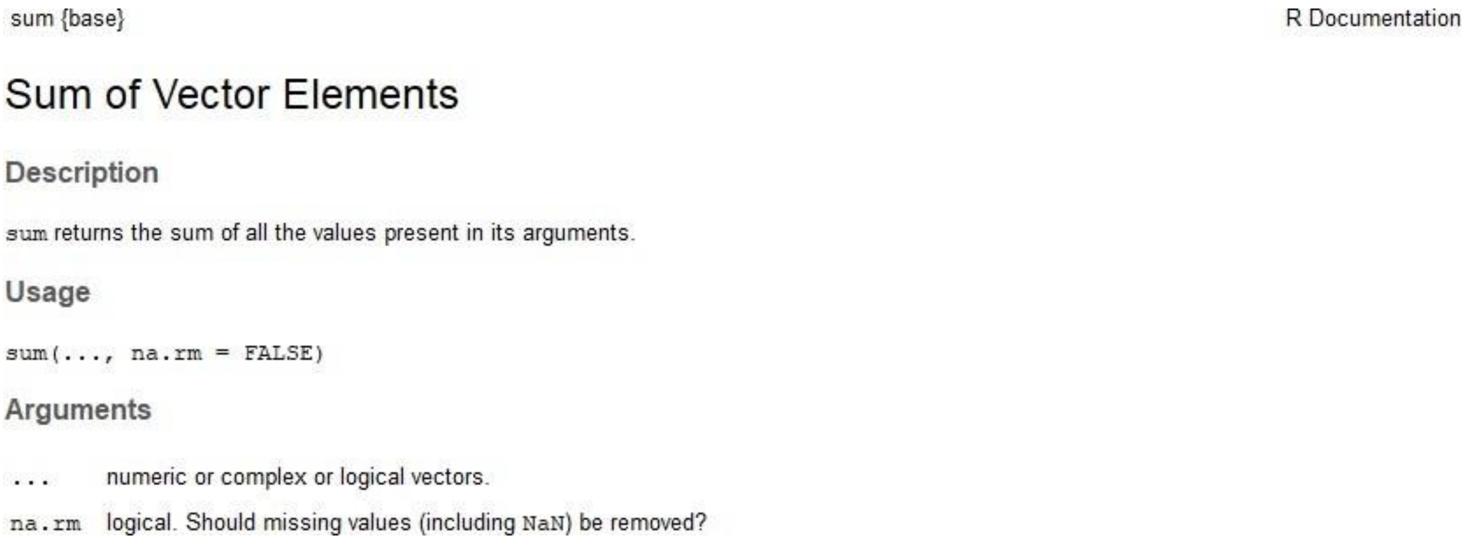
```
> x <- 2; x+2
[1] 4
```

- # indicates comments:

```
> maxvalue <- 2 # Data above two is not relevant
```

R help

```
> ?sum # equivalent to help(sum)
```



sum {base} R Documentation

Sum of Vector Elements

Description

`sum` returns the sum of all the values present in its arguments.

Usage

```
sum(..., na.rm = FALSE)
```

Arguments

`...`	numeric or complex or logical vectors.
`na.rm`	logical. Should missing values (including `NaN`) be removed?

Using R as a calculator

```
> 2*3
[1] 6
>log(6)/2^2
[1] 0.4479399
>exp(6)-4
[1] 399.4288
> pi-3
[1] 0.1415927
```

Using R as a programming language

```
> x <- 2.0
> x
[1] 2.0
> y = 3.0 # Equivalent to y <- 3.0
> y; x
[1] 3
[1] 2
>1/x
[1] 0.5
```

Creating vectors using the c() command

```
> x <- c(1.3, 0.32  10.5, 5.9, 6.3)
              ,
> x
[1] 1.30 0.32 10.5  5.90 6.30
              0
> y <- c(x, 1.4, x, x); y
[1] 1.30 0.32 10.5  5.90 6.30
              0
[6] 1.40 1.30 0.32 10.50 5.90
[11] 6.30 1.30 0.3  10.50 5.90
               2
[16] 6.30
```

Vector operations

Vector operations work element by element:

```
> x <- c(1.3, 0.32, 10.5, 5.9, 6.3)
> y <- x*2; y
[1] 2.60 0.64 21.00 11.80 12.60
>z  <- x*y; z
[1] 3.38 0.21 220.50 69.62 79.38
```

Recycling

- If a vector is too short, R recycles it (reuses it) as needed:

```
> x <- c(1.3, 0.32, 10.5, 5.9)
> y <- c(2, 10)
> x*y
[1] 2.6 3.2 21.0 59.0
1.3*2 0.32*10 10.5*2 5.9*10
```

- A warning message is displayed if the shortest vector can not be recycled entirely:

```
> x <- c(1.3, 0.32, 10.5, 5.9, 6.3)
> x*y
[1] 2.6 3.2 21.0 59.0 12.6
Warning message:
In x * y :
longer object length is not a multiple of shorter object length
```

Generating sequences of numbers

```
> 1:10
[1] 1 2 3 4 5 6 7 8 9 10
```

This is equivalent to:

```
>c(1,2,3,4,5,6,7,8,9,10)
[1] 1 2 3 4 5 6 7 8 9 10
> 10:1
[1] 10 9 8 7 6 5 4 3 2 1
```

Beware of operator priority

```
> x <- 2*1:10
# equivalent to x <- 2*(1:10)
> x
[1] 2 4 6 8 10 12 14 16 18 20
> n <- 10
> 1:n-1
# equivalent to (1:n)-1
[1] 0 1 2 3 4 5 6 7 8 9
> 1:(n-1)
[1] 1 2 3 4 5 6 7 8 9
```

The seq() function: the same, but more flexible

```
> seq(from=1, to=10)
[1] 1 2 3 4 5 6 7 8 9 10
> seq(from=1, to=5, by=0.5)
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0
> x <- seq(from=1, to=5, length=17)
> x
[1] 1.00 1.25 1.50 1.75 2.00 2.25 2.50 2.75
[9] 3.00 3.25 3.50 3.75 4.00 4.25 4.50 4.75
[17  5.0
]     0
```

Non numeric vectors: boolean (logical) values

```
> x <- seq(from=1, to=5, length=17)
> x
 [1] 1.00 1.25 1.50 1.75 2.00 2.25 2.50 2.75
 [9] 3.00 3.25 3.50 3.75 4.00 4.25 4.50 4.75
 [17] 5.00
> y <- x<5  # help(“<”) shows list of relational operators
> y
 [1]  TRUE TRUE  TRUE TRUE  TRUE TRUE
 [7]  TRUE TRUE  TRUE TRUE  TRUE TRUE
 [13] TRUE TRUE FALSE
>sum(x<5)
 [1] 16
```

Missing values are designated by NA

```
> z <- c(1:3,NA)
> z
[1] 1 2 3 NA
> is.na(z)
[1] FALSE FALSE FALSE TRUE
> mean(z)
[1] NA
> mean(z, na.rm=TRUE)
[1] 2
```

Character strings

```
> char <- c("hello","world","!"); char
[1] "hello" "world" "!"
```

Vectors can not combine numbers and characters:

```
> char <- c("hello",3:5,"world"); char
[1] "hello" "3" "4" "5" "world"
> char <- c(char, NA); char
[1] "hello" "3" "4" "5" "world" NA
```

Selecting subsets of vectors using [ ]

```
> x <- 10:30
> x[2]
[1] 11
> x[1:5]
[1] 10 11 12 13 14
```

Selecting subsets of vectors using [ ] and boolean vectors

```
> x <- 10:30
> x[x>25]
[1] 26 27 28 29 30
>x <-c(seq(from=5, to=10,by=0.5),NA,
seq(from=11,to=15,by=0.5),NA,
seq(from=16,to=20,by=0.5))
> x[!is.na(x)]
[1] 5.0 5.5 6.0 6.5 7.0 7.5 8.0 8.5
[9] 9.0 9.5 10.0 11.0 11.5 12.0 12.5 13.0
[17] 13.5 14.0 14.5 15.0 16.0 16.5 17.0 17.5
[25] 18.0 18.5 19.0 19.5 20.0
```

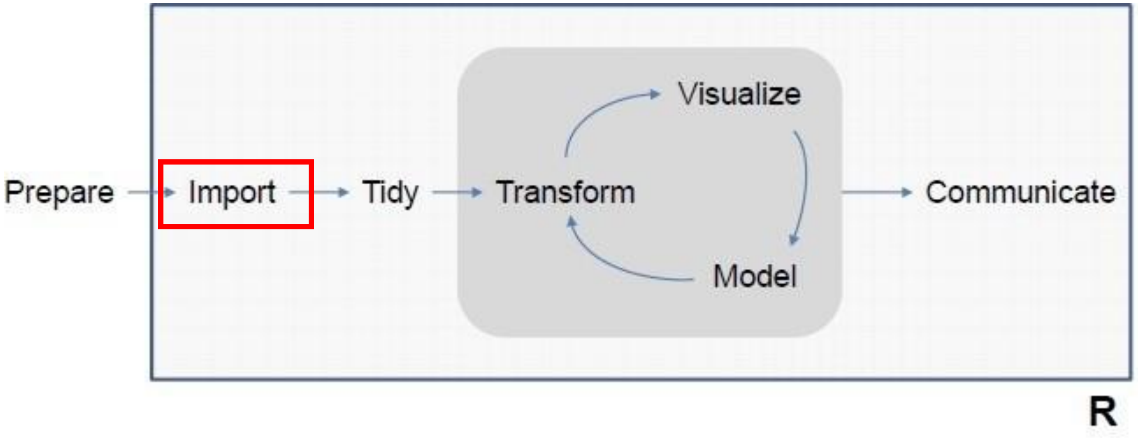
Changing parts of vectors using [ ]

```
> x[32] <- 200
> x[c(10,29)] <- c(1,100)
> x[x>15] <- NA
```

Finding the length of a vector

```
> x <- 1:5
> length(x)
[1] 5

> y <- 1:16
>len <- length(y) ; len
[1] 16
```

Data analysis workflow



Adapted from Hadley Wickham

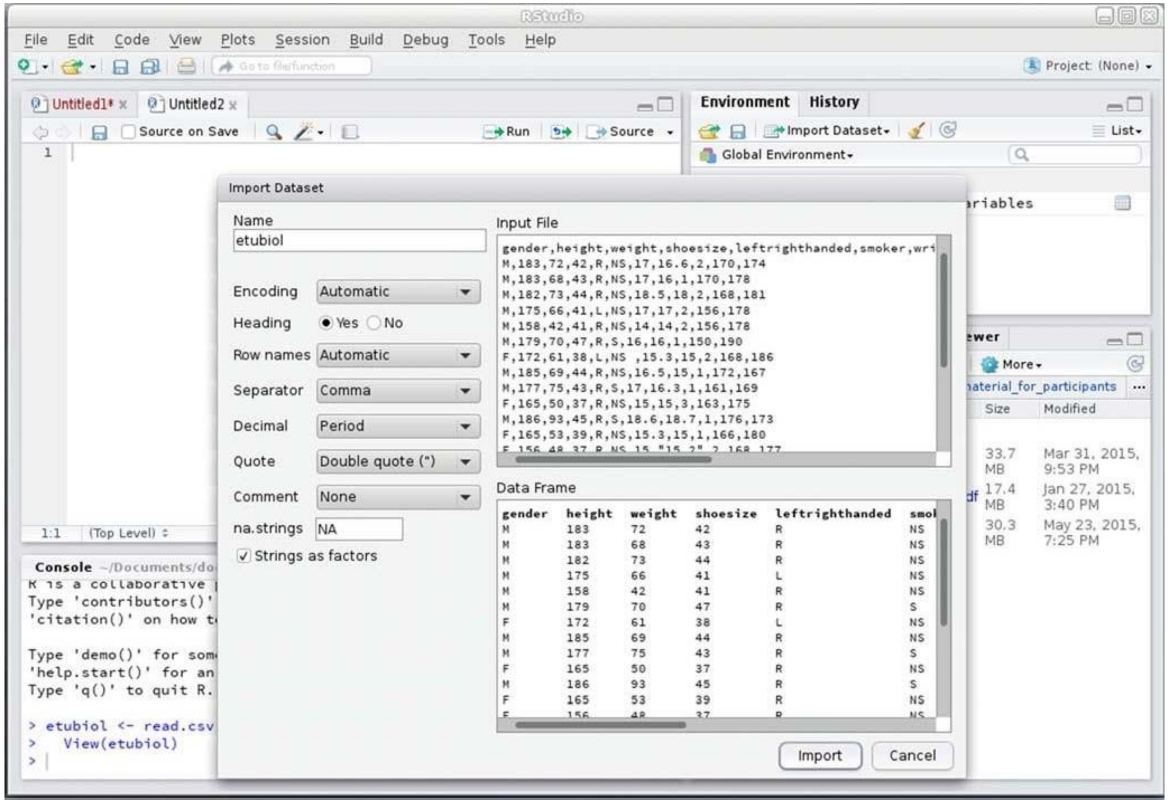
Importing data into R

- R can import flat files using e.g. the commands:

```
read.table()
read.csv()
read.delim()
```

(with many options – check the help).

- R can also:
  - Read Excel spreadsheets
  - Read plenty of other formats
  - Directly access databases
  - Access files over the web



Import Dataset

Name: etubiol

Encoding: Automatic

Heading: ● Yes ○ No

Row names: Automatic

Separator: Comma

Decimal: Period

Quote: Double quote (")

Comment: None

na.strings: NA

☑ Strings as factors

Input File

```
gender,height,weight,shoesize,leftrighthanded,smoker,wri
M,183,72,42,R,NS,17,16.6,2,170,174
M,183,68,43,R,NS,17,16,1,170,178
M,182,73,44,R,NS,18.5,18,2,168,181
M,175,66,41,L,NS,17,17,2,156,178
M,158,42,41,R,NS,14,14,2,156,178
M,179,70,47,R,S,16,16,1,150,190
F,172,61,38,L,NS ,15.3,15,2,168,186
M,185,69,44,R,NS,16.5,15,1,172,167
M,177,75,43,R,S,17,16.3,1,161,169
F,165,50,37,R,NS,15,15,3,163,175
M,186,93,45,R,S,18.6,18.7,1,176,173
F,165,53,39,R,NS,15.3,15,1,166,180
F,156,48,37,R,NS,15,"15.2",2,168,177
```

Data Frame

```
gender  height  weight  shoesize  leftrighthanded  smol
M       183     72      42        R                NS
M       183     68      43        R                NS
M       182     73      44        R                NS
M       175     66      41        L                NS
M       158     42      41        R                NS
M       179     70      47        R                S
F       172     61      38        L                NS
M       185     69      44        R                NS
M       177     75      43        R                S
F       165     50      37        R                NS
M       186     93      45        R                S
F       165     53      39        R                NS
F       156     48      37        R                NS
```

Console ~/Documents/do

```
R is a collaborative
Type 'contributors()'
'citation()' on how t

Type 'demo()' for som
'help.start()' for an
Type 'q()' to quit R.

> etubiol <- read.csv
>   View(etubiol)
> |
```

Data frames

- Data frames are made of columns having all the same number of elements
- They look like matrices, except that the columns can hold different variables types
- They are typically used to store data, with
    - Each row being an experimental unit
    - Each column being a measurement

```
> data[,1] # access first column
> data[, "data1"] # access column "data1"
> data$data1 # … same
```

Creating data frames

```
> x <- 1:10
> y <- seq(from=5,to=10,length=10)
> z <- c("A","B","B","A","A","A","B","A","B","B")
> df <- data.frame(d1=x, d2=y, fact=z)
> df
 d1          d2 fact
1   1   5.000000    A
2   2   5.555556    B
..
> names(df)
[1] "d1" "d2" "fact“
>dim(df)
[1] 10  3
```

Adding new columns

```
> df$d3 <- 10:1
> df
 d1        d2 fact d3
1   1 5.000000    A 10
2   2 5.555556    B  9
…
> summary(df)
       d1              d2             fact                 d3
 Min.   : 1.00   Min.   : 5.00   Length:10          Min.   : 1.00
 1st Qu.: 3.25   1st Qu.: 6.25   Class :character   1st Qu.: 3.25
 Median : 5.50   Median : 7.50   Mode  :character   Median : 5.50
 Mean   : 5.50   Mean   : 7.50                      Mean   : 5.50
 3rd Qu.: 7.75   3rd Qu.: 8.75                      3rd Qu.: 7.75
 Max.   :10.00   Max.   :10.00                      Max.   :10.00
```

Select data from a data frame

- Select all values of "d2" for which "fact" is "B"

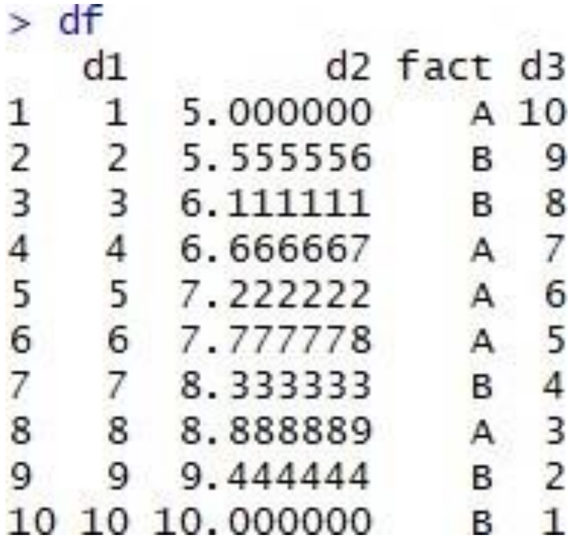
```
> df[ df$fact == "B", "d2" ]
[1]  5.555556  6.111111  8.333333  9.444444 10.000000
```

- Select all values of "d1" for which "fact" is "B " and "d2" > 7

```
> df[ (df$fact == "B" & df$d2 > 7), "d1" ]
[1]  7  9 10
```

- Select all values of "d3" for which "fact" is “A " or "d2" < 6

```
>df[ (df$fact == "B" | df$d2 < 6), "d3" ]
[1] 10  9  8  4  2  1
```



```
> df
   d1        d2 fact d3
1   1  5.000000    A 10
2   2  5.555556    B  9
3   3  6.111111    B  8
4   4  6.666667    A  7
5   5  7.222222    A  6
6   6  7.777778    A  5
7   7  8.333333    B  4
8   8  8.888889    A  3
9   9  9.444444    B  2
10 10 10.000000    B  1
```

Exercise

- Import students.csv into a variable (call it data)
- Extract the weight of women only in a new variable
- Extract the weights of the people who weight more than 80 kilos
- Extract the entries of men who weight more than 80 kg (you can use the "&" operator to include two conditions)

If you do not know what to do:

1.Extract the weight of women only in a new variable

2.Extract the weights of the people who weight more than 80 kilos

3.Extract the entries of men who weight more than 80 kg

[you can use the "&" operator to include two conditions]