An Overview of the R Data-Mining & Machine Learning Environment

Ariel Faigon

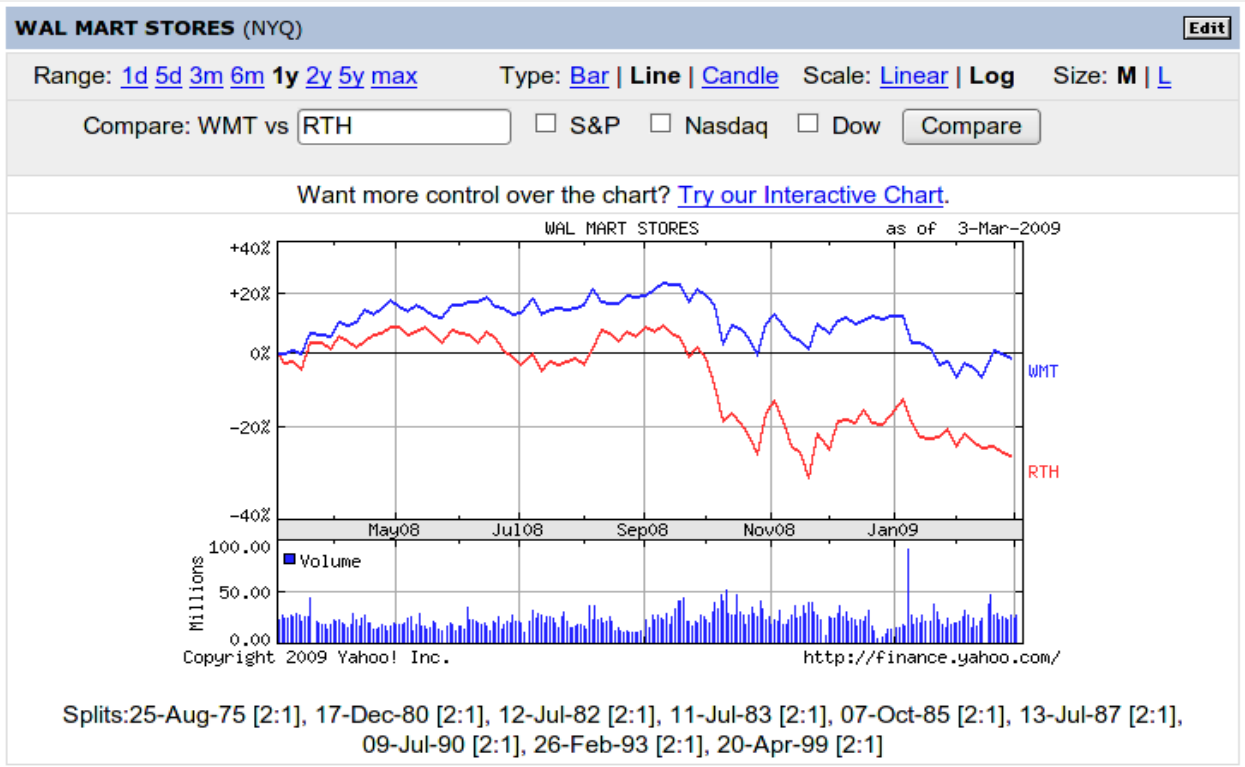
Outline

Introduction to "R" & Machine Learning

- ***Motivation:*** ***why bother?***
- ***Why R?*** ***there are so many other options...***
- ***R primer*** ***the basics, a quick how-to***
- ***Data Sets*** ***where it all starts***
- ***Models*** ***building & evaluation (the holy grail)***

Motivation: exhibit #1

WalMart vs. retail sector in a recession



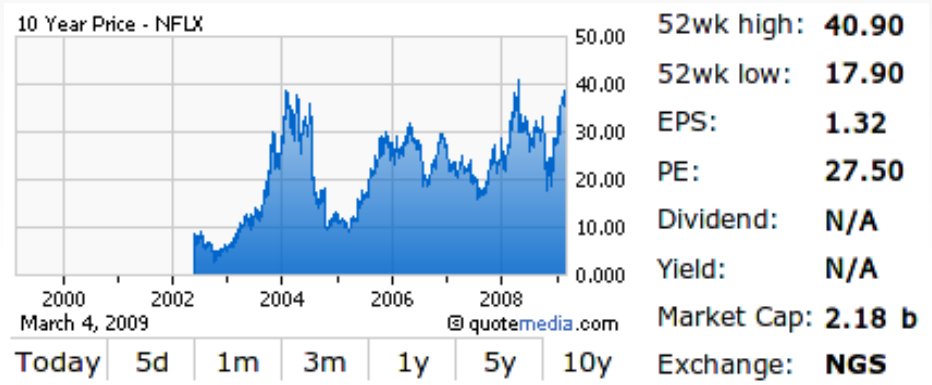
WalMart allows more than 3,500 suppliers, to access data on their products and perform data analyses...

Motivation: exhibit #2

Fall of Blockbuster; Rise of Netflix:

Blockbuster Inc.

Netflix Inc.



www.netflixprize.com/rules to read about Cinematch

Motivation: exhibit #3

Print vs. Online Media:

Newspaper Death Watch

Chronicling the Decline of Newspapers and the Rebirth of Journalism

R.I.P.

US metropolitan dailies that have ceased print publication since March, 2007

- Rocky Mountain News
- Baltimore Examiner
- Kentucky Post
- Cincinnati Post
- King County Journal
- Union City Register-Tribune
- Capital Times
- Halifax Daily News
- Albuquerque Tribune
- South Idaho Press
- San Juan Star

Podcast: Educator Challenges Convention

By paulgillin | March 4, 2009 - 10:57 am - Posted in Citizen Journalism, Future of Journalism, Journalism

As journalism educators struggle with questions of how to teach students about a new media world that even they don't fully understand, some teachers are moving forward and reshaping their programs to prepare students for a very different career track. Hanson Hosein was an Emmy-winning television news producer at NBC News before ditching the world of "big-box" media and setting out to discover how citizen publishers would change journalism. Today he heads the masters of communication program at the University of Washington, where his curriculum has created considerable debate over its rejection of traditional media models. We interview him on our MediaBlather podcast.

Click here to go to MediaBlather.

(Be the first to comment)

Discussing - and Discarding - Solutions

By paulgillin | March 3, 2009 - 9:00 pm - Posted in Business News, BusinessModel, Future of Journalism, Journalism, Layoffs, Local news, Newspapers, Solutions

The Associated Press wraps up the debate over public and non-profit funding for newspapers, concluding that the economics could work for a few large players but not for most metro dailies. *The New York Times* would need an endowment of about $5 billion to sustain its current newsgathering operation, for example. The more promising and popular approach is a targeted for-profit model like MinnPost.com and investigative journalism

Go

Pages

About
How I Can Help Your Newspaper
Speaking

Poll Question

Will Cablevision Succeed in Its Plan to Charge Subscription Fees for Newsday.com?

- Yes, it's a bold first step
- Yes, but it could only happen in New York
- No, it's a suicidal move
- Results will probably be inconclusive

Cast Vote!

Display Poll Results
Poll Archive
Suggest a Poll

Motivation: exhibit #4

Data mining goes mainstream & into popular culture

The New York Times

Business Computing

WORLD U.S. N.Y. / REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION
AUTOS

Search Technology Go

Inside Technology
Internet | Start-Ups | Business Computing | Companies

Data Analysts Captivated by R's Power

Stuart Isett for The New York Times

R first appeared in 1996, when the statistics professors Robert Gentleman, left, and Ross Ihaka released the code as a free software package.

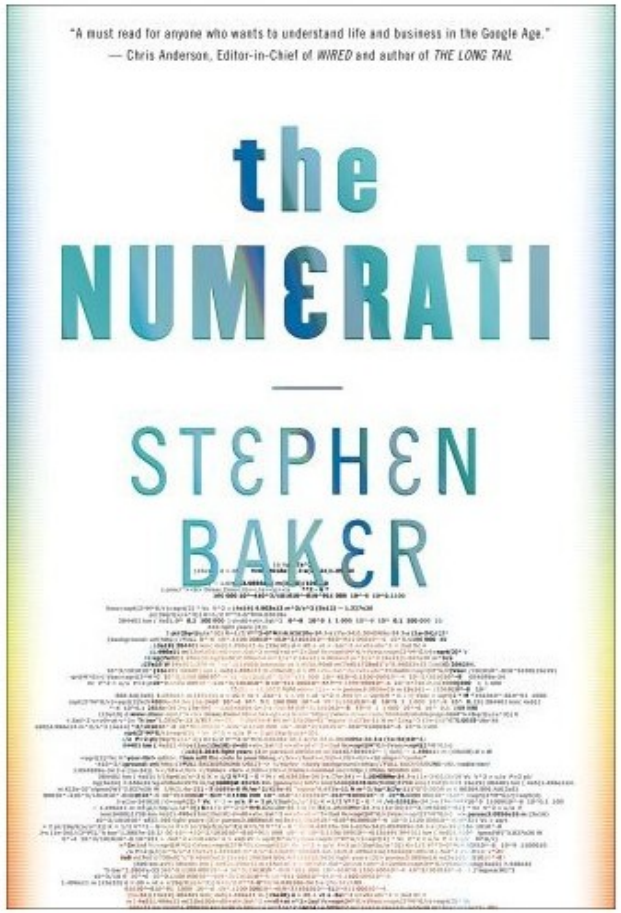
By ASHLEE VANCE
Published: January 6, 2009

SIGN IN TO E-MAIL
PRINT
SINGLE PAGE
REPRINTS
SHARE

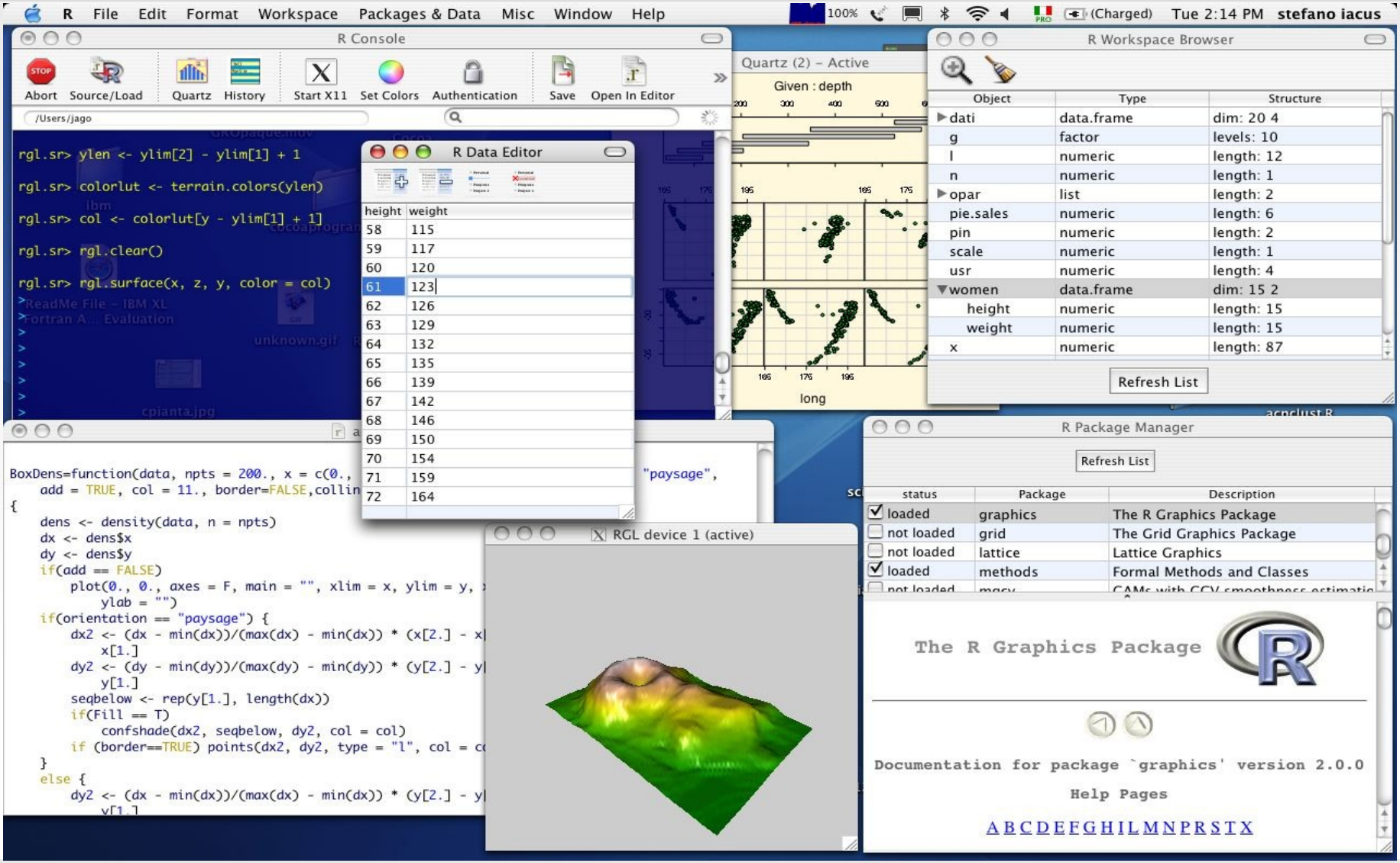
To some people R is just the 18th letter of the alphabet. To others, it's the rating on racy movies, a measure of an attic's insulation or what pirates in movies say.

Related

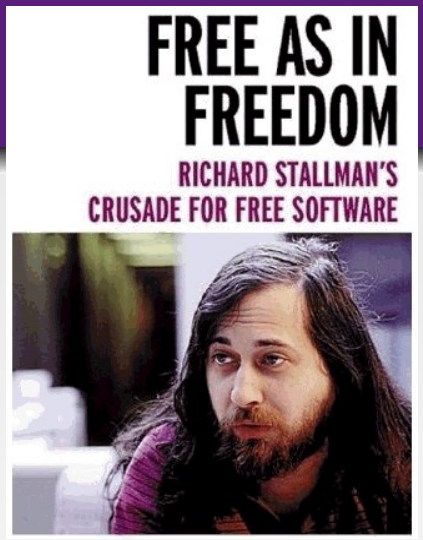
R is also the name of a popular programming language used by a



R workspace (on Mac OS-X)



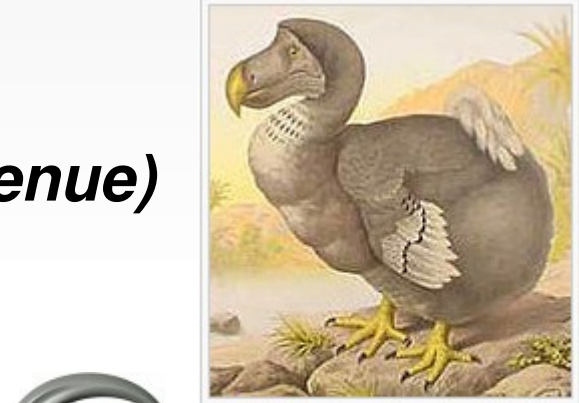
Why R (there are so many options)?



- *FOSS – GNU GPL*
- *Free both as in Beer and as in Freedom*
- *No license hassles, no nagware, forced upgrades...*
- *Large community, experience, help – critical mass*
- *Becoming the "lingua franca" for DM, ML*
- *Job security: used everywhere in the corp. world*
  - **Google, Pfitzer, Merck, BofA, Shell...**

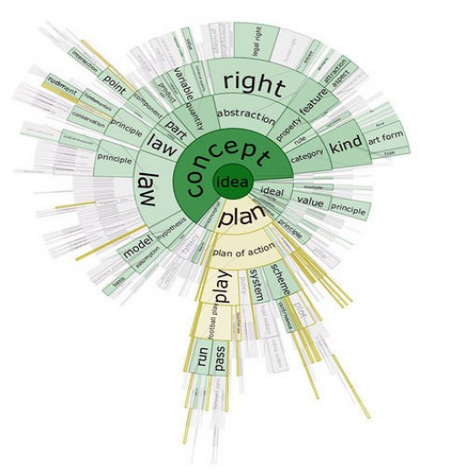
Why R?

- *Hides ML algorithmic complexity "under the hood"*
- *Huge/rich library: 1600 packages available (Jan 2009) "standing on the shoulders of giants" - Hal Varian*
- *Alternatives like SAS – 2B (yearly revenue) are in extinction danger denial*



The Dodo, shown here in a 1651 illustration by Jan Savery, is an often-cited example of modern extinction.[1]

Why R?

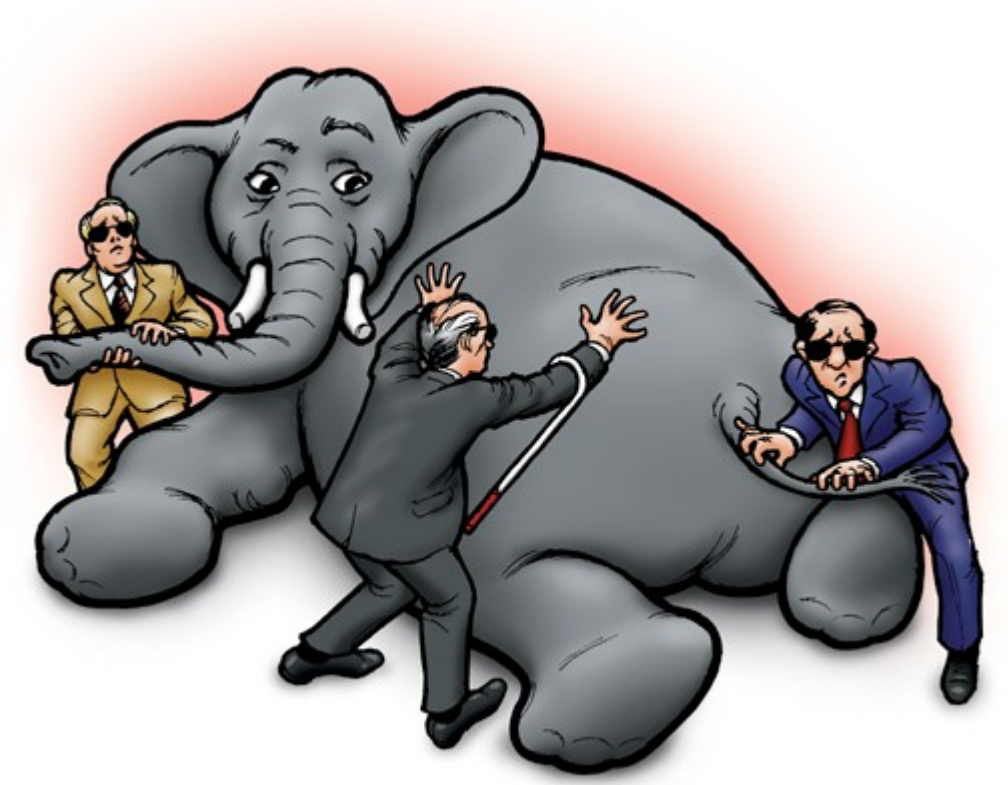


From a more technical perspective:

- ***Excellent visualization capabilities – leverages gnuplot, but hides all the nitty-gritty and warts***
- ***Relatively small language (only the libraries are big)***
- ***Short time to "get productive", a long time to becoming a master***

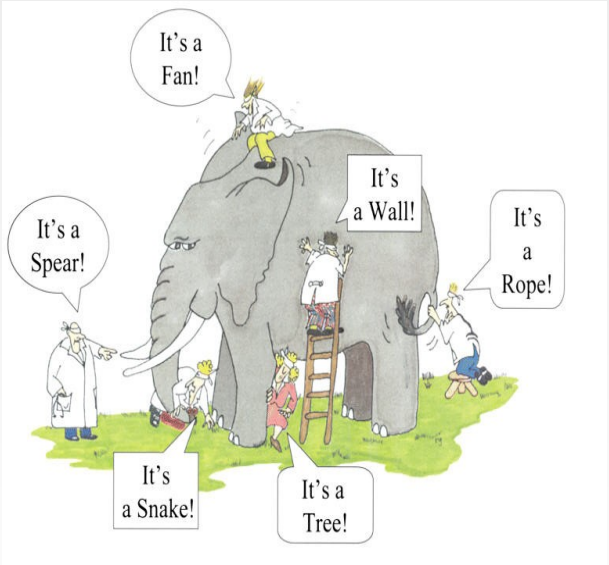
R primer: what is R?

The 4 blind men & the elephant metaphor:



R primer: what is R?

The 4 blind men & the elephant metaphor:



- ***Using Excel?***

  It's "Excel on steroids"

- ***For statisticians:***

  A statistical analysis suite

- ***A Programmer?***

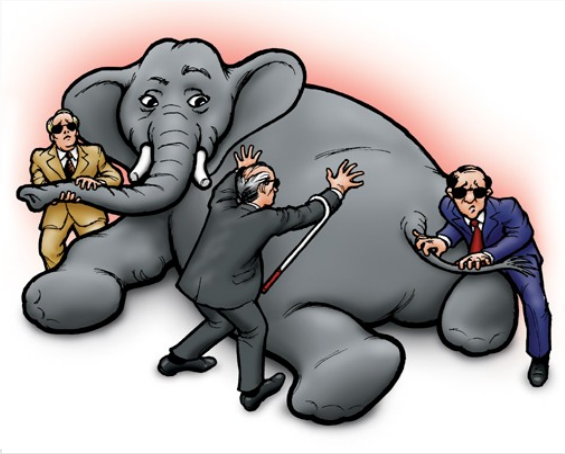
  A programming lang + (programmable) shell

R primer: what is R?

The 4 blind men & the elephant metaphor:

For a "machine learning person," R is:

- **A data mining and machine learning suite**
- **A tool to (easily) visualize data**
- **Standard API to cutting-edge learning algorithms from leading research institutions**



R primer: what is R?

A bit of history and trivia:

- ***Started as a clone of "S" from Bell Labs***
- ***Written by two Auckland, NZ statisticians***
- ***A mix of C + Scheme/LISP + Unix shell***
  - **Conditions:** *if (condition) { ... }*
  - **Loops:** *for (var in list...) { ... }; break; next*
  - **Functions:** *funcname(name=value, ...)*
  - **Unix heritage:** *rm(); ls(); head(); tail()*

R primer: what is R? More trivia

CRAN: Comprehensive R Archive Network:

- **A mirrored repository is always near you**

- ***~1600 libraries - by the best of the DM/ML researchers***
  - **Many libs are compiled C (or Fortran)**
- ***Very easy to extend:***
  - **mix compiled libs with your own interpreted code**
- ***"Polymorphic" in a sense:***
  - **Encourages defaults e.g. in function-params**
  - **e.g:** *plot(x)* **in most cases "just works", regardless of what** x **is**

R primer: the first baby steps

- *From the shell, just say 'R' (uppercase-R)*
- *On windows, Mac-OS you get a GUI*
- *Warming up, use it as a calculator:*

```
R> 4+4
[1] 8
R> 2^3
[1] 8
R> sqrt(64)
[1] 8
R> log(e) * 7 + 63/63
[1] 8
```

R primer: the first baby steps

- ***Stuck? Use some help:***

R> ***help(mean)***

R> ***?mean***

R> ***help.search('mean')***

R> ***apropos('mean')***

```
mean                    package:base                    R Documentation

Arithmetic Mean

Description:

     Generic function for the (trimmed) arithmetic mean.

Usage:

     mean(x, ...)

     ## Default S3 method:
     mean(x, trim = 0, na.rm = FALSE, ...)

Arguments:

       x: An R object.  Currently there are methods for numeric data
          frames, numeric vectors and dates.  A complex vector is
          allowed for 'trim = 0', only.

    trim: the fraction (0 to 0.5) of observations to be trimmed from
          each end of 'x' before the mean is computed. Values of trim
          outside that range are taken as the nearest endpoint.
```

R primer: 2nd baby steps

- ***Variables, vectors, ranges & vector operations***

```
R> x = 1:10

R> x
 [1]  1  2  3  4  5  6  7  8  9  10

R> x * 2
 [1]  2  4  6  8  10  12  14  16  18  20

R> x * 2 + x
 [1]  3  6  9 12 15  18  21  24  27  30
```

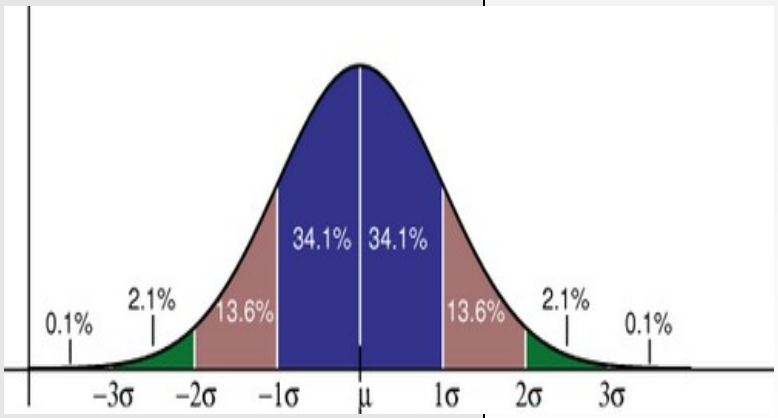
$$\mathbf{v} = \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \\ \mathbf{c} \end{bmatrix}$$

R primer: 2nd baby steps

- ***Vectors & some descriptive statistics:***

R> ***x = 1:10; x = 2 * x + x; x***

[1] 3 6 9 12 15 18 21 24 27 30



R> ***mean(x); sd(x)***

[1] 16.5

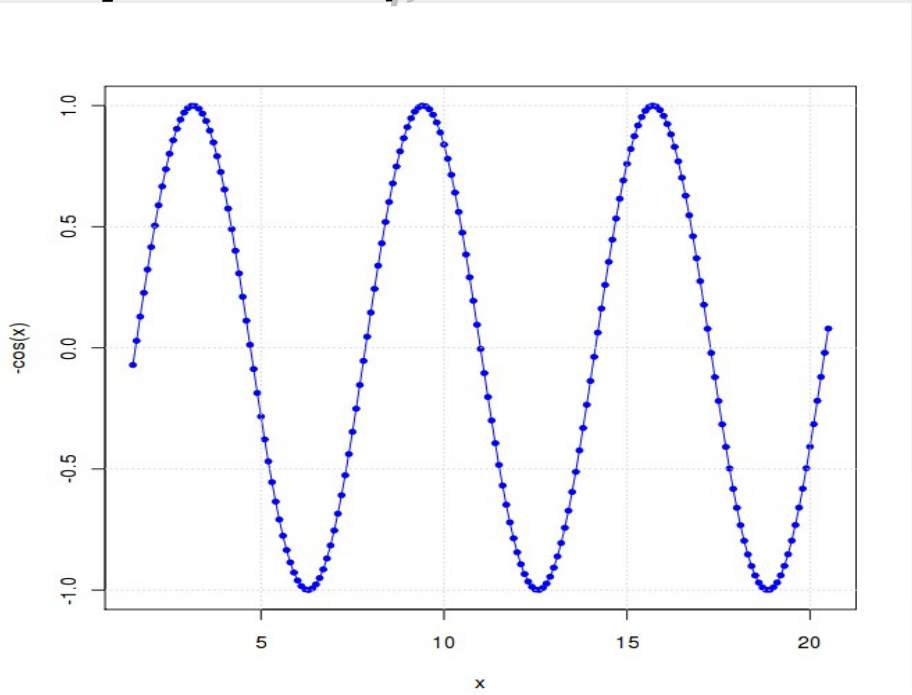
[1] 9.082951

R> ***summary(x)***

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
3.00	9.75	16.50	16.50	23.25	30.00

R primer: my first plot

Simple XY plot, seq, built-in function: cos



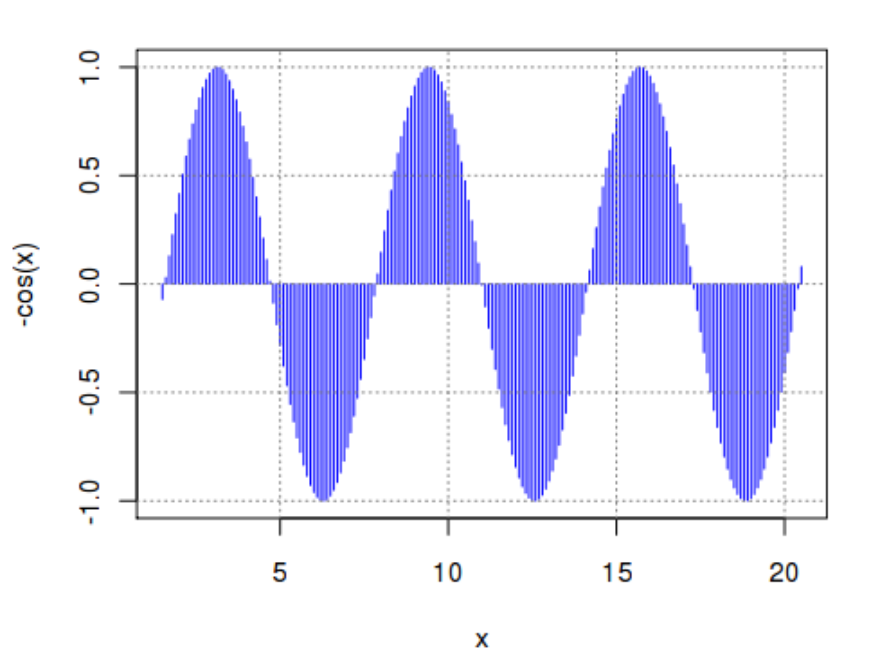
R> *x = seq(1.5, 20.5, by=0.1)*

R> *plot(x, -cos(x), pch=20, col='blue', type='o')*

R> *grid()*

R primer: my 2nd plot

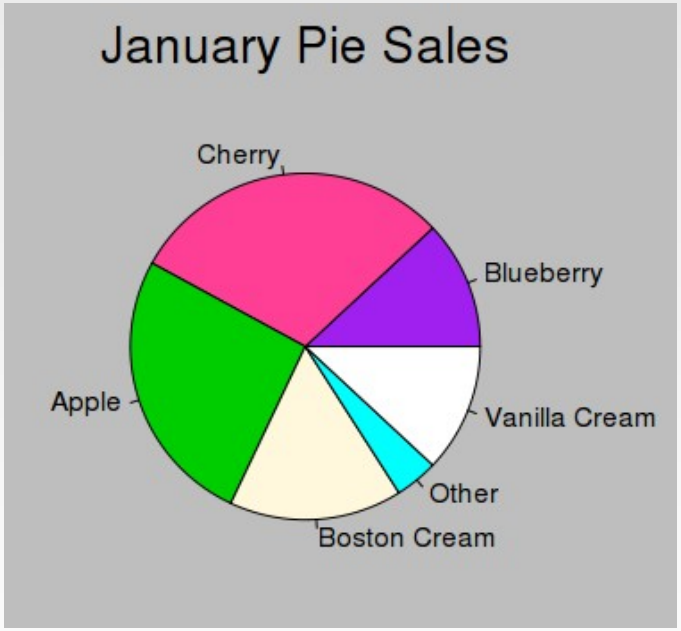
Simple XY plot, many options, (try: help(plot))




```
R> x = seq(1.5, 20.5, by=0.1)
R> plot(x, -cos(x), pch=20, col='blue', type='h')
R> grid(col='#777777')
```

R primer: more plots

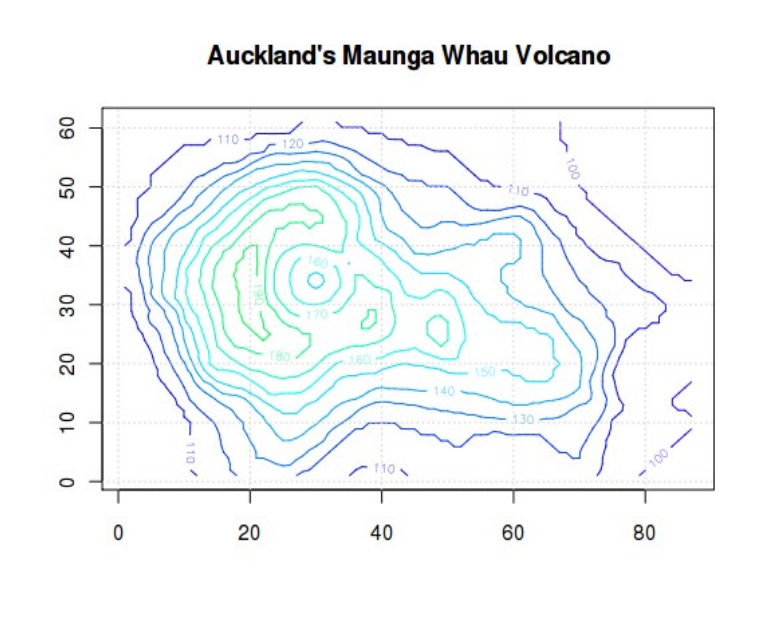
Simple pie chart (try: "demo(graphics)")

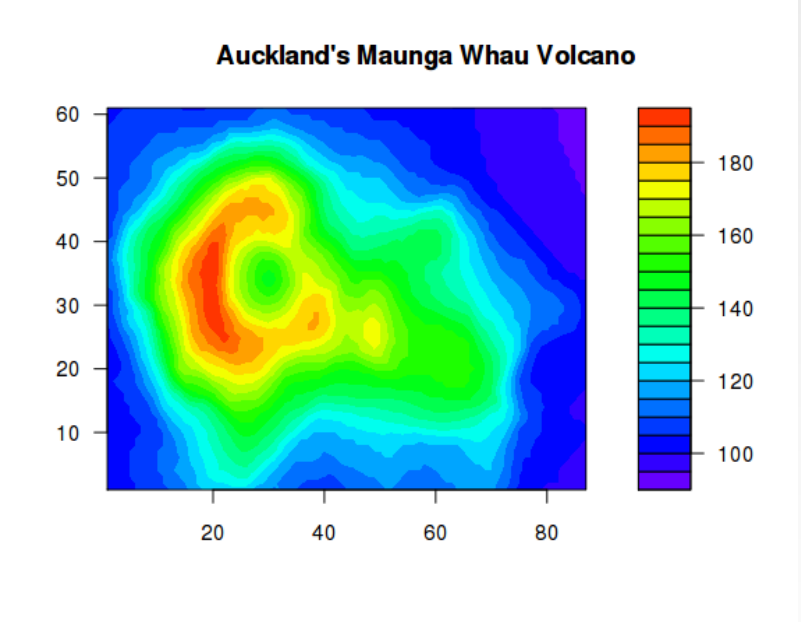


```
R> pie.sales <- c(0.12, 0.3, 0.26, 0.16, 0.04, 0.12)
R> names(pie.sales) <- c("Blueberry", "Cherry", "Apple", "Boston Cream",
                         "Other", "Vanilla Cream")
R> pie(pie.sales, col=c("purple","violetred1","green3","cornsilk","cyan","white"))
R> title(main="January Pie Sales", cex.main=1.8, font.main=1)
```

R primer: more plots

Contour plots:

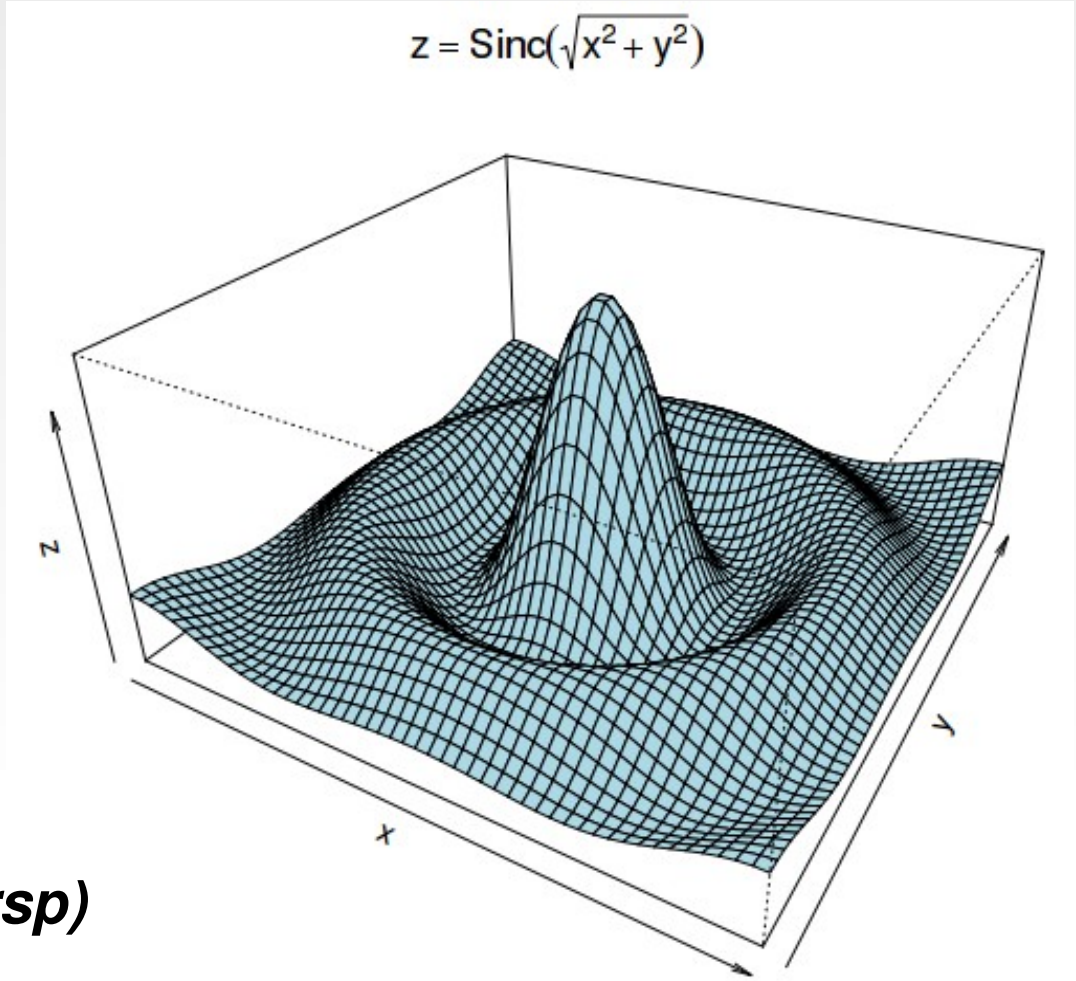





```
R> contour(1:NROW(volcano), 1:NCOL(volcano), volcano, ...); grid()
R> title(main="Auckland's Maunga Whau Volcano")
R> filled.contour(1:NROW(volcano), 1:NCOL(volcano), volcano,
                                        col=rev(rainbow(22, end=11/15)))
R> title(main="Auckland's Maunga Whau Volcano")
```

R primer: more plots

3D perspective:

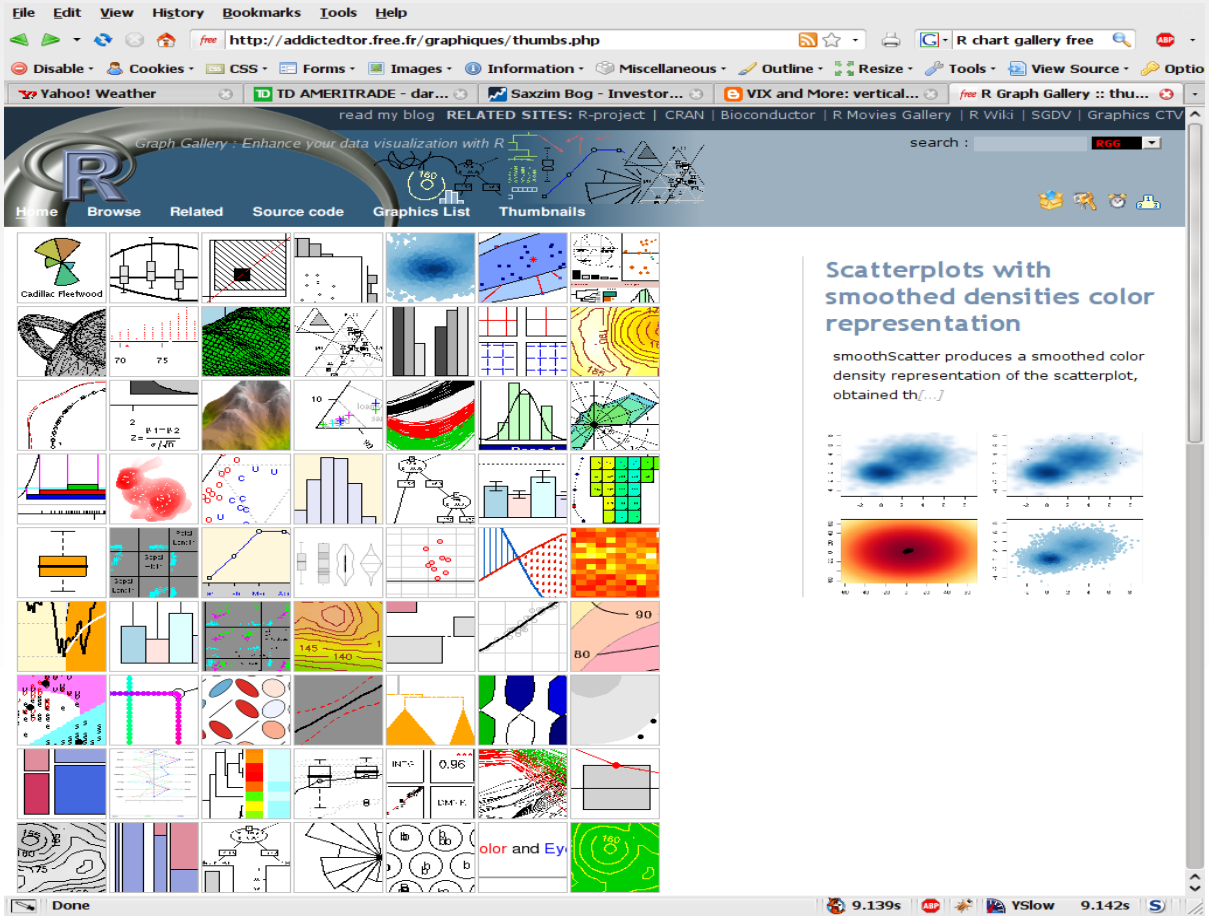


R> ***demo(persp)***

R graphics: more plots than you can imagine

Get your fix (examples with full R source) at:

http://addictedtor.free.fr/graphiques/



R functions: generic by design

- ***By default: args passed by value, name, position***
- ***Local name space, referenced by name***

```
R> fib <- function(x1=0, x2=1, len=10) {      # non recursive fibonacci
       res <- vector(len=len)
       for (i in 1:len) {
            sum <- x1+x2;     res[i] <- sum
            x1 <- x2;         x2 <- sum
       }
       return(res)
   }
R> fib()                      # use default parameter values
[1]  1  2  3  5  8  13  21  34  55  89
R> fib(x1=7, len=6)           # override x1, len defaults, leave x2 intact
[1]  8  9  17  26  43  69
```

Leonardo Fibonacci
(dall'opera *I benefattori dell'umanità*; vol. VI, Firenze, Ducci, 1850)

Random sampling with R

- ***Coin Toss: "Heads", "Tails"***

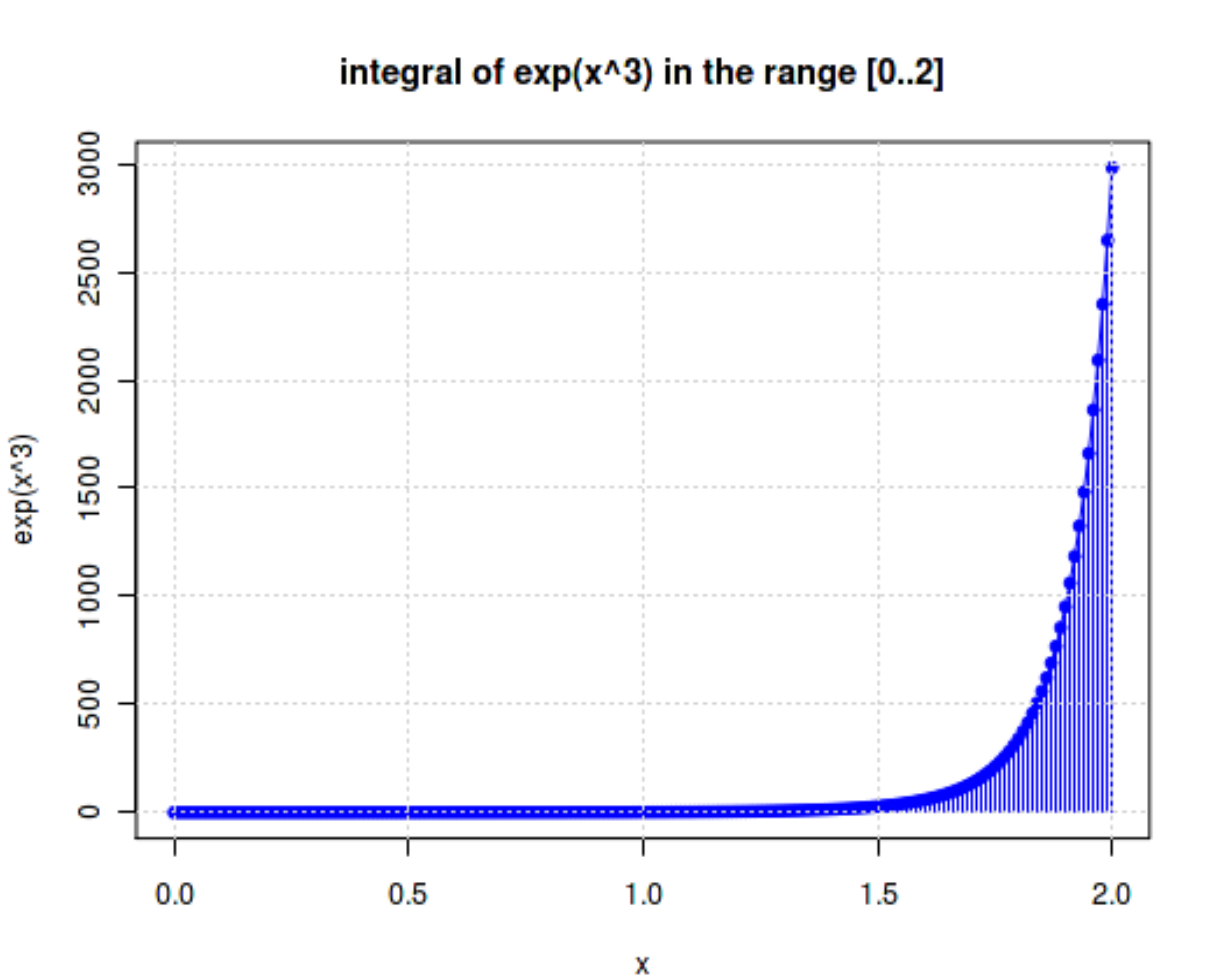
```
R> sample(c('H', 'T'), 8, replace=TRUE)
[1] "T" "T" "T" "H" "T" "T" "H" "T"
```

- ***Balls from an urn: "red", "blue", "yellow"***

```
R> urn = c(rep("red", 8), rep("blue", 15), rep("yellow", 9))
R> sample(urn, 6, replace=FALSE)
[1] "red"    "blue"   "yellow" "blue"   "yellow" "blue"
```

The power of random: Monte Carlo Simulation

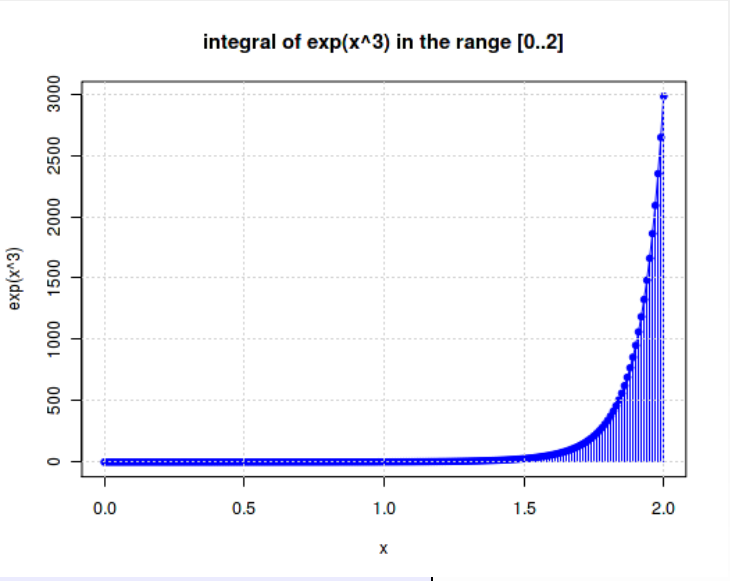
Compute definite integral using Monte-Carlo:



Monte Carlo Simulation

- ***runif(): random uniformly-distributed sequence***
- ***Compute definite integral using Monte-Carlo:***

```
R> random.x = runif(1000000, min=0, max=2)
R> random.y= exp(random.x^3)
R> width = 2 - 0
R> estimated.area = width * mean(random.y)
R> estimated.area
[1]  276.7474
# Or, in one step:
# remember: this is just an approximation
R> 2*mean(exp(runif(1000000, min=0, max=2)^3))
[1]  275.5778
```



Hypothesis testing: t-test

- ***Given two vectors: are they really different?***
- ***Example: 6 cars MPG with vs. without additive:***

```
R> add   <- c(24.6, 18.9, 27.3, 25.2, 22.0, 30.9)   # MPG w/ add
R> noadd <- c(23.8, 17.7, 26.6, 25.1, 21.6, 29.6)   # MPG w/o add
R> t.test(add, noadd, paired=T, alt = "greater")
Paired t-test        data: add and noadd
t = 3.9994, df = 5, p-value = 0.005165
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:     0.3721225     Inf
sample estimates:  mean of the differences   0.75
```

- ***p=0.005165: conclusion, additive improves MPG***

When statisticians invent a programming language...

Gotchas & warts in R:

- *"switch" is a function, not a keyword*
- *Array indexes start at [1]*
- *Assignment: '=' and '<-' are the same, except...*
- *"mean()" applies only to 1st arg*
- *} else { must be on the same line*
- *Type conversions aren't implicit, except...*
- *Constant vector syntax is funny: c(1,2,3, ...)*
- *Different libs, different names/styles*
- *many more...*

Moving into “real life”

Using R for predicting the future

R Data Sets

- ***Many toy data-sets are included in R***
- ***RDB-like tables are called "data-frames"***

R> ***data()***

... faithful Old Faithful Geyser Data

R> ***help(faithful)***

... A data frame with 272 observations of two variables ...

R> ***head(faithful, 4)***

	eruptions	waiting
1	3.600	79
2	1.800	54
3	3.333	74
4	2.283	62

R types, introspection

- ***Simple types: character, numeric (integer or real), factor (enum), ...***
- ***Complex types: vector, matrix, data-frame, list, time-series, ...***

R> ***dim(faithful)***

[1] 272 2

R> ***str(faithful)***

'data.frame': 272 obs. of 2 variables:

$ eruptions: num 3.60 1.80 3.33 2.28 4.53 ...

$ waiting : num 79 54 74 62 85 55 88 85 51 85 ...

R> ***attributes(faithful)***

$names

[1] "eruptions" "waiting"

$row.names

[1] 1 2 3 4 5 6 7 ...

[19] 19 20 21 22 23 24 25 ...

$class

[1] “data.frame”

R types, introspection & conversion

- ***Simple types: character, numeric (integer or real), factor (enum), ...***
- ***Complex types: vector, matrix, data-frame, list, time-series, ...***

```
R> typeof(5);  typeof(5.7)
[1]  "double"
[1]  "double"
R> as.character(5);   as.numeric("123.45")
[1]  "5"
[1]  123.45
R> as.Date("22jan2006", format="%d%b%Y")
[1] "2006-01-22"
R> as.factor(c('low', 'medium', 'high'))
[1] low    medium high
Levels: high low medium
```

R Data Sets

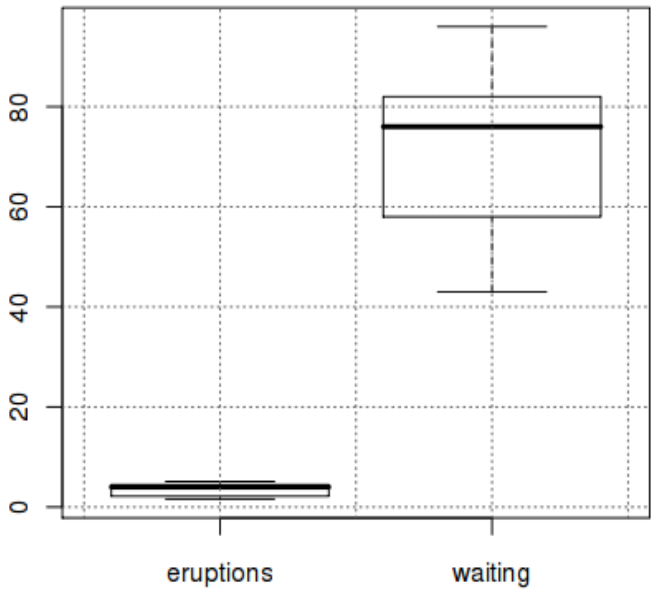
- ***boxplot: quick visual min/max/mean/quartile***

R> ***head(faithful, 4)***

	eruptions	waiting
1	3.600	79
2	1.800	54
3	3.333	74
4	2.283	62

R> ***boxplot(faithful)***

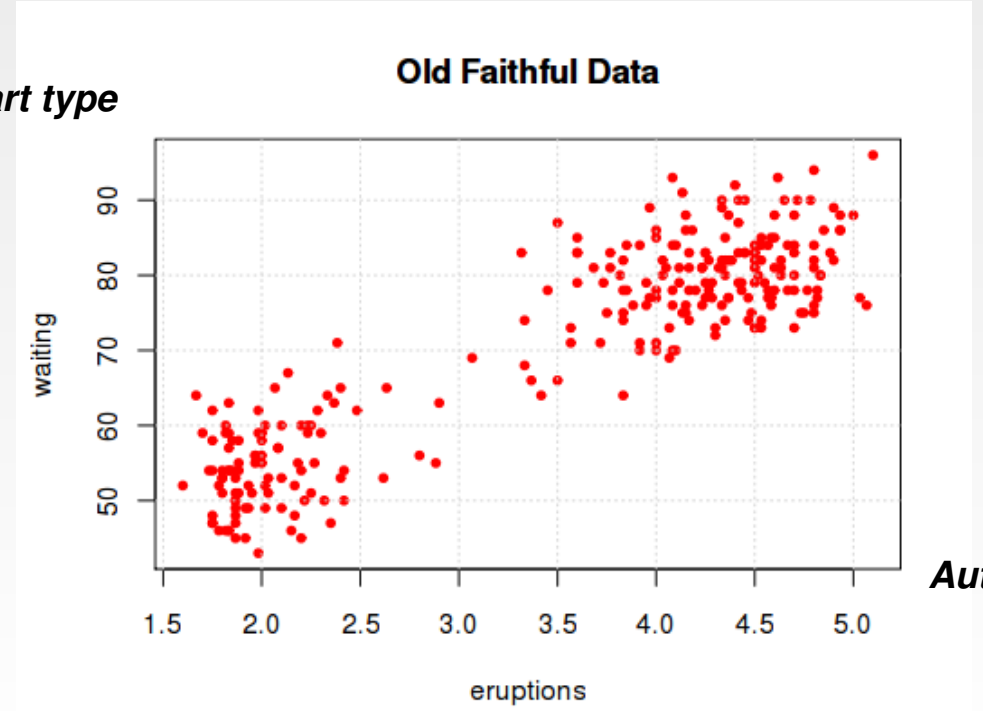
R> ***grid(col="#555555")***



R Data Sets

- ***Plot a data-frame: plot is "clever"***

Automatic chart type



Automatic axis labeling

R> *plot(faithful, pch=20, col='red', main='Old Faithful Data')*
R> *grid(col="#555555")*

R Data Sets

Accessing data frame elements:

R> ***faithful$waiting*** # "member" column by name

[1] 79 54 74 62 85 55 88 85 51 85 54 84 78 ...

[26] 83 55 76 78 79 73 77 66 80 74 52 48 80 ...

...

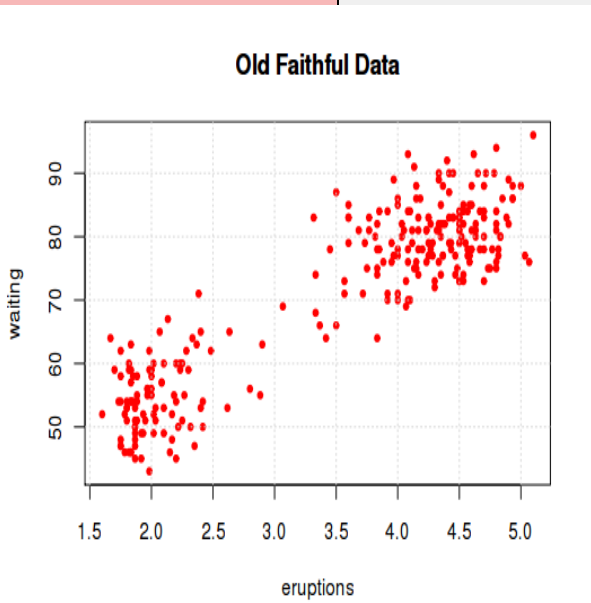
R> ***faithful[['waiting']]*** # column slice by name

R> ***faithful[[2]]*** # column slice by index

R> ***faithful[,2]*** # ditto (row omitted)

R> ***faithful[20,]*** # a single row

	eruptions	waiting
20	4.25	79



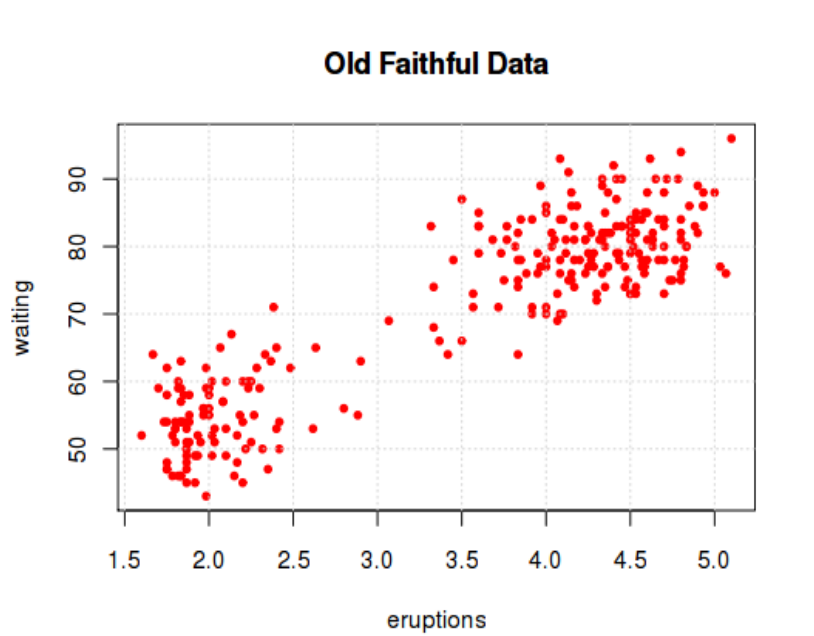
R Data Sets: arbitrary expression slices

Accessing data frame elements:

R> ***attach(faithful)*** *# "globalize" members, save typing*

R> ***faithful[eruptions > 5.0,]*** *# row slice by expression*

	eruptions	waiting
76	5.067	76
149	5.100	96
151	5.033	77

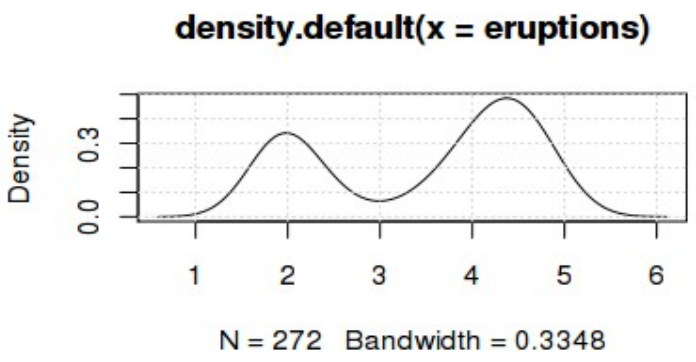


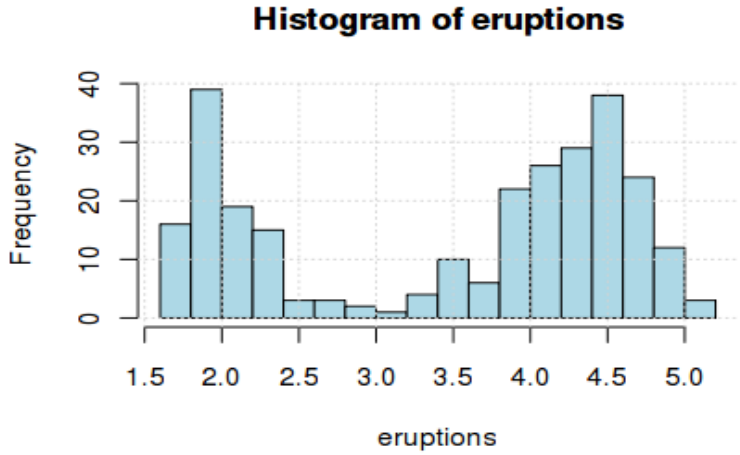
R Data Sets

Old Faithful: density plot, histogram

R> *attach(faithful)*

R> *plot(density(eruptions))*



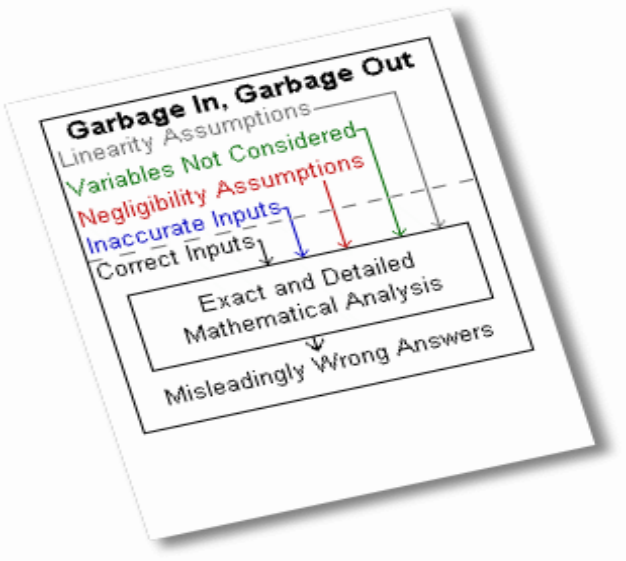


R> *hist(eruptions, br=20, ...)*

The holy grail: good models

What is a Model?

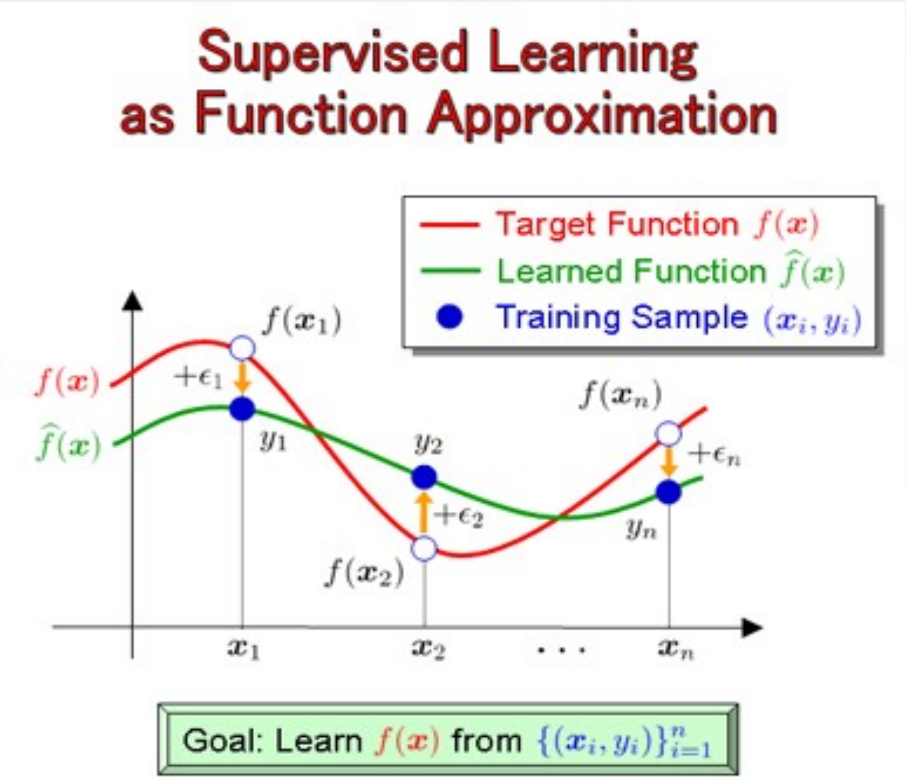
- ***A simplified representation of the "real world"***
- ***A good model: an** accurate **description of "reality"***
- ***Given partial data, a good model can accurately "guess" or "predict" the missing part(s)***



Models: created by "learning" from examples

Supervised learning:

Creating a model from examples (a "training set")



Models: created by “learning” from examples

Gasket Lookup

Part Number	Minimum Temperature	Maximum Temperature	Type	Thickness
4001	-946	931	1	0.25
4002	-601	1894	1	0.0312
4003	456	791	1	0.125
4004	-259	2433	0	0.0625
4005	-982	3543	0	0.0625
4006	652	3285	1	0.25
4007	326	1187	1	0.0625
4008	-960	-271	1	0.25
4009	904	3529	0	0.0625
4010	-149	406	0	0.0625
4011	972	2686	0	0.25
4012	-321	346	0	0.0312

Supervised learning:

Creating a model from examples (a “training set”)

Common case (M x N values):

- *N variables (aka: dimensions, attributes, or features)*
- *M instances (aka: examples)*
- *Last instance is missing a variable – guess or predict its value*

In R: a data frame has MxN (Rows x Columns)

- *So... a data-frame can be a training set from which we can create a model, and a test-set to apply the model on*

The holy grail: models

Revisiting Old Faithful:

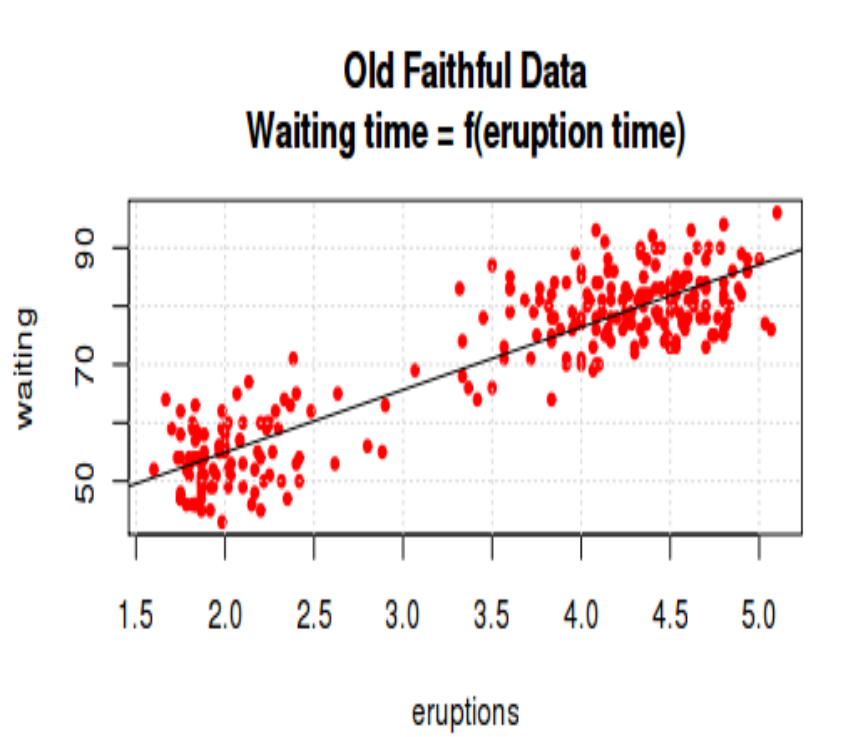
- **Can we predict the waiting time till next eruption?**
- **As it turns out -- given last eruption duration -- yes we can**
- **Because the two are highly correlated**

 # according to Pearson

R> ***cor(waiting, eruptions)***

[1] 0.9008112

The correlation is 0.9008112



Models: create a simple linear model (lm)

Predicting Old Faithful:

- **Model for predicting the waiting time till next eruption**
- **The model is the regression line: Y = a + bX**

R> mymodel = lm(waiting ~ eruptions) *# fit a linear model to the data*

R> abline(mymodel) *# overlay/plot the model*

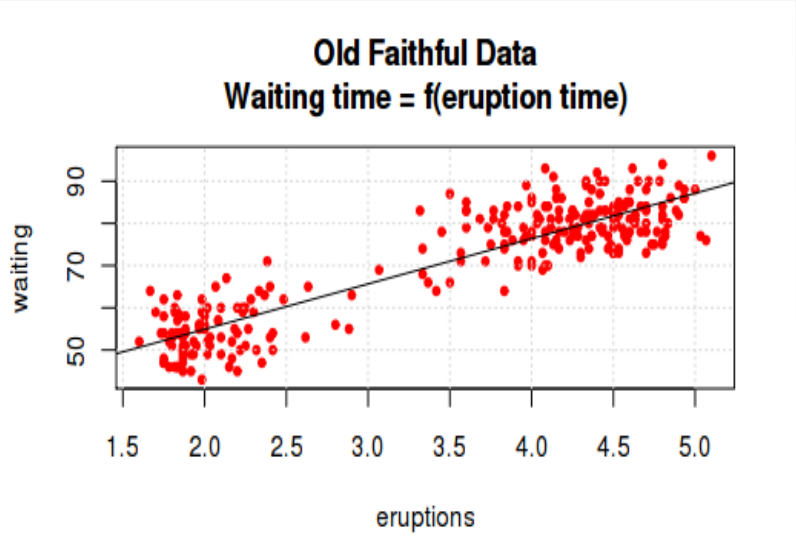
R> mymodel

Call: lm(formula = waiting ~ eruptions)

Coefficients:

(Intercept)	*eruptions*
33.47	*10.73*

IOW: ***Y*** *= a +* ***bX*** *a=33.47 b=10.73*



A simple linear model (lm)

OK**, so we have plotted the regression line, but **how do we predict?

Use: ***predict.lm(model, test_set)***

Let's invent some (mostly crazy) eruption duration times:

```
R> test.eruptions = data.frame(eruptions=c(2, 7, 8, 90))
```

and see what our model predicts for them:

```
R> predict.lm(mymodel, test.eruptions)
       1        2        3        4
54.93368 108.58189 119.31153 999.14212
```

A simple linear model (lm): evaluation

- ***How accurate is our model?***

For an optimistic estimate, use the training set as a test-set, ignore Y

```
R> bestguess = predict.lm(mymodel, faithful)
R> cor(bestguess, waiting)   # (Pearson) compare predicted vs actual
[1] 0.9008112
```

Not surprisingly, exactly the same as the X vs Y correlation:

About 90% accurate.

- ***This method is great for comparing different models***
- ***But "out-of-sample" test-set estimate is more realistic***

A simple linear model (lm): evaluation

Cross validation is a more robust approach:

Break your data into separate training-set + test-set, rinse, repeat

R> *library(DAAG)*

R> *cv.lm(faithful, formula(waiting ~ eruptions), m=3, dots=F)*

... Overall ms

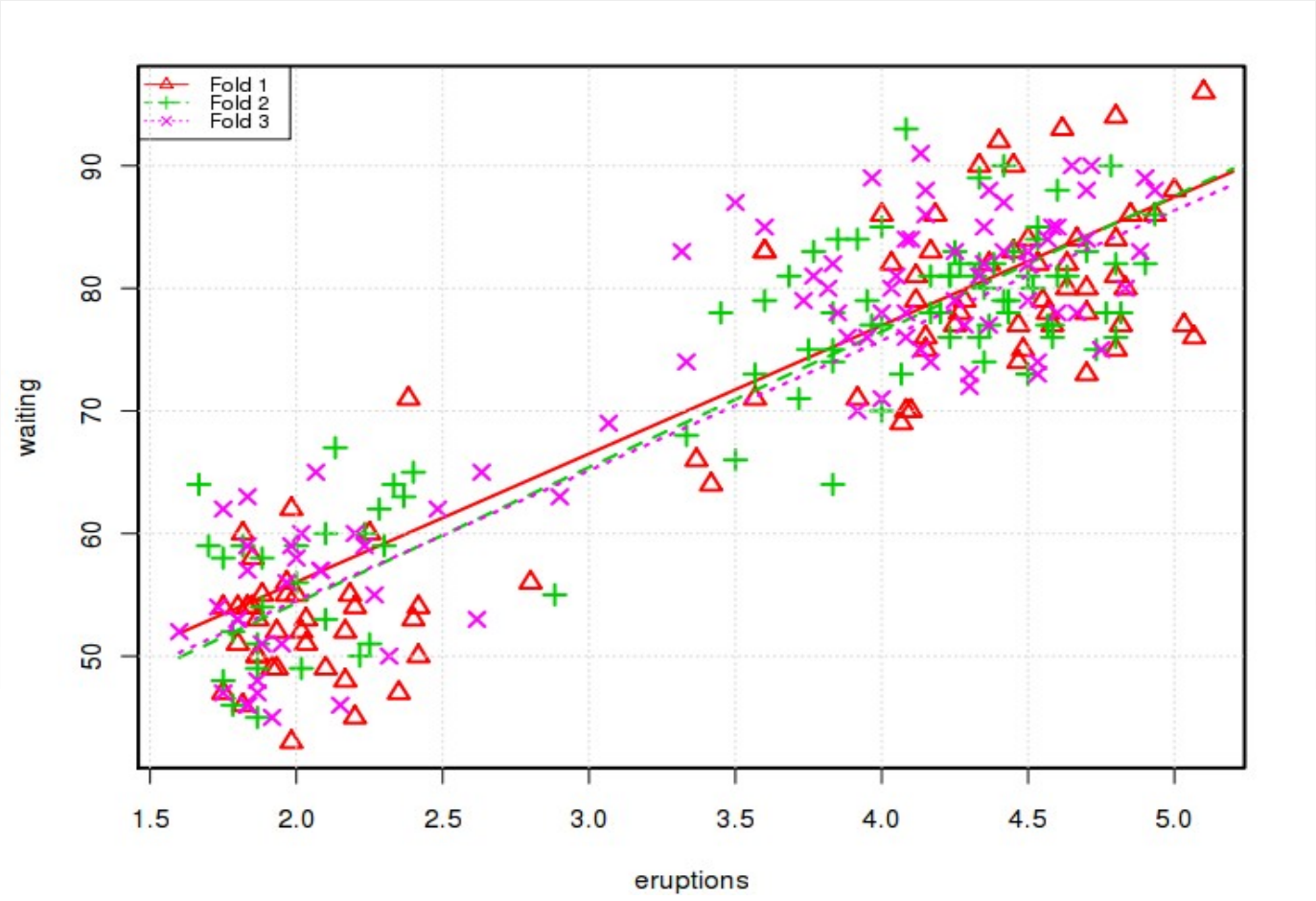
36.6

R> *sqrt(36.6) / mean(waiting)*

[1] 0.0853 *# about 8.5% mean relative error*

Evaluation using cross-validation (cv.lm)

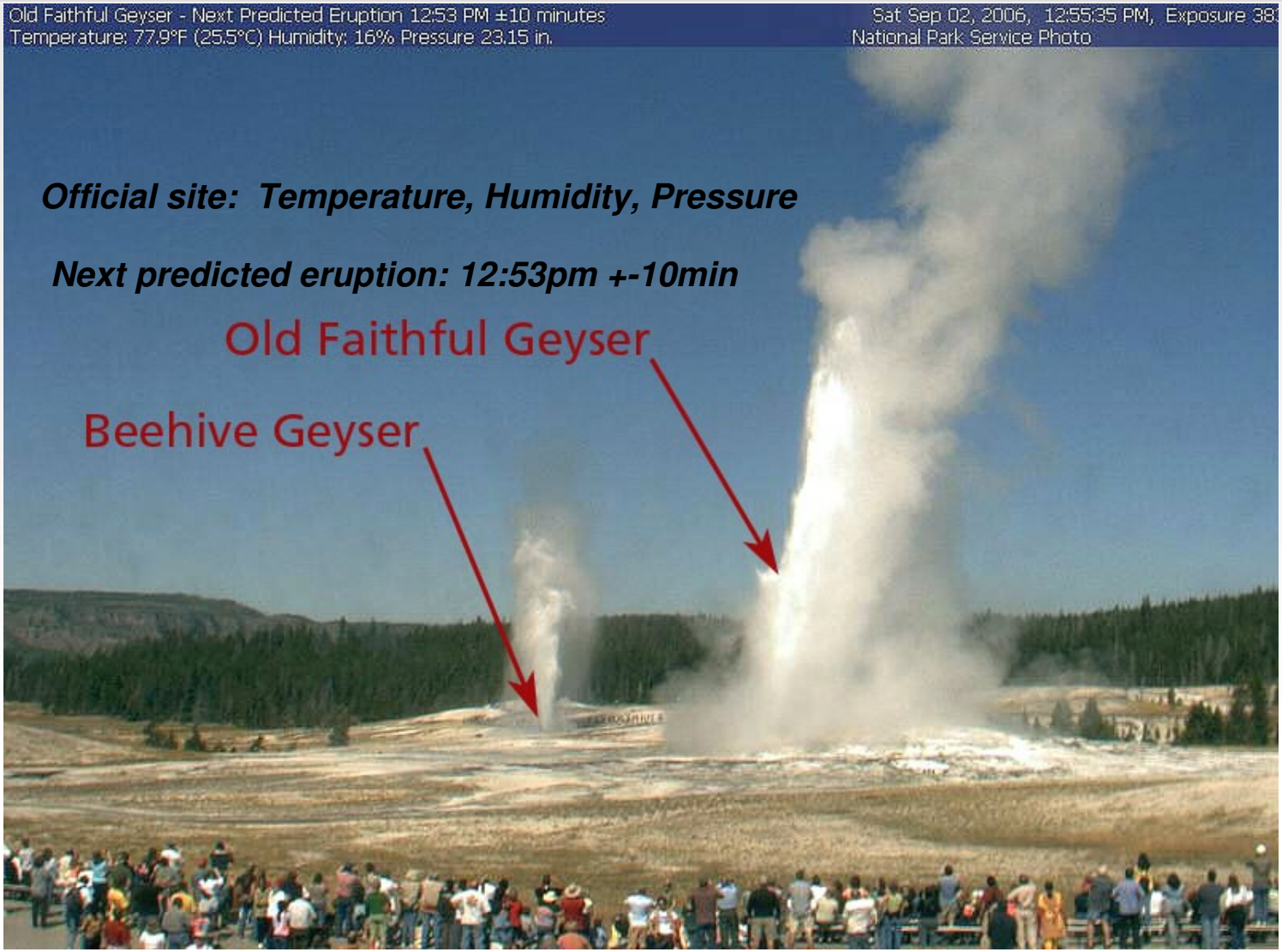
Cross validation in a picture (generated by cv.lm()):



Better (more accurate) models:

- *More/better data:*
  - *Bigger sample*
  - *Different times*
- *More/better input features:*
  - *Atmospheric pressure*
  - *Temperature*
  - *Humidity*
- *Better Modeling:*
  - *Ridge-regression*
  - *Non linear models*
  - *Multiple "experts" (committee)*

More powerful models:



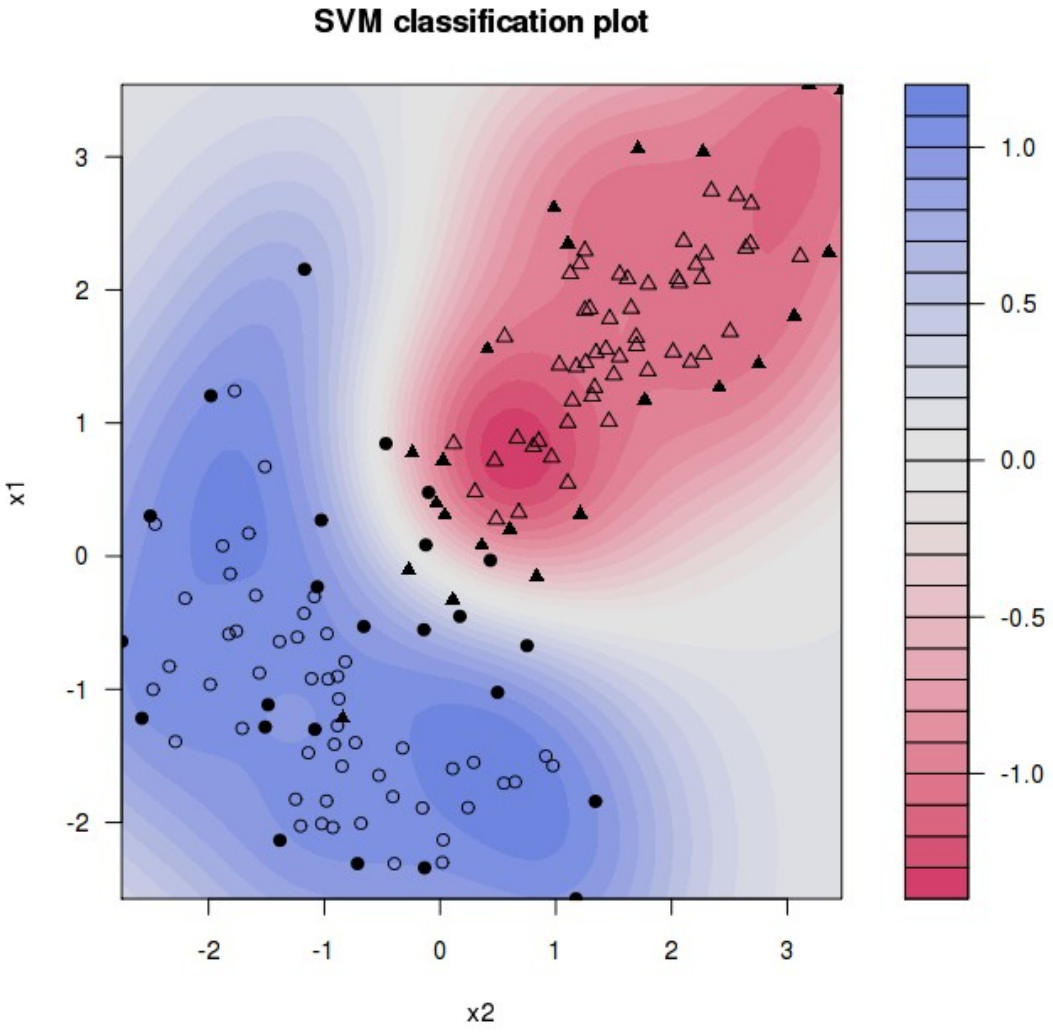
Non linear model example: SVM classification

- *SVM w/ radial kernel*
- *2 input variables: x1, x2*
- *2-way classification*
- *Circles vs. Triangles*

```
R> library(mvtnorm)
R> library(kernlab)

   ...data creation details omitted...

R> fm = ksvm(class ~ . ,
                data=d, C=0.5)
R>  plot(fm, data = d)
```



There's so much more in R...

In the immortal words of R. Hillel:

Challenged to teach the whole torah while standing on one foot:
Said "NP, it is simple...love others as you would love yourself"

"Ve Idach, Zil Gmor"

Meaning:

...the rest, [is just interpretation,] go study it...

http://en.wikipedia.org/wiki/Hillel_the_Elder

RTFMs, Docs, etc.

- ***There are a lot of great docs online:***

  Starting point: ***www.r-project.org***

  Search engine for all things R: ***rseek.org***

- ***When all else fails:***
  - *Go to a shell, type 'R' and experiment*
  - *Inside "R":*
    - ***Use*** *help('topic')*
    - ***Use*** *help.search('topic')*
    - ***Try*** *demo()*
    - ***Look at examples in help***
    - ***Use trial and error, inspect/plot your vars***

Questions?

"The future is already here.

It is just unevenly distributed"

-- William Gibson

Thank You!