

# Introduction to GWAS using R and GenABEL

## LUPA Workshop in Statistical Methods for GWAS studies

Marcin Kierczak\*

\*Computational Genetics Group  
Faculty of Veterinary and Animal Breeding  
Swedish University of Agricultural Sciences  
Uppsala, SWEDEN

04-05 May 2011, Uppsala

# Introduction to R

## What is R?

### What is R?

R is a:

- programming language
- software environment

for:

- statistical computing
- beautiful graphics



# Introduction to R

## Why?

- *de facto* standard among statisticians
- widely used for development and data analysis
- an implementation of the S programming language
- partially inspired by Scheme
- created by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand
- source code is freely available under the GNU GPL
- mainly command line
- GUIs exist
- many tools/tests at hand
- project homepage: <http://www.r-project.org>

# Introduction to R

## What is R?

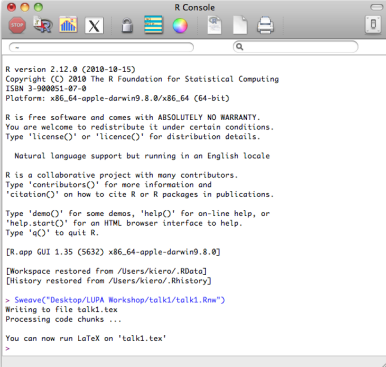
### R code example

```
> 2 + 2
[1] 4

> p.value <- 0.05
> p.value
[1] 0.05

> -log10(p.value)
[1] 1.30103

> print("Hello!")
[1] "Hello!"
```



```
R Console

R version 2.12.0 (2010-10-15)
Copyright (C) 2010 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-apple-darwin9.8.0/x86_64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.35 (5632) x86_64-apple-darwin9.8.0]

[Workspace restored from /Users/kiero/.RData]
[History restored from /Users/kiero/.Rhistory]

> Sweave("~/Desktop/LUPA Workshop/talk1/talk1.Rnw")
Writing to file talk1.tex
Processing code chunks ...

You can now run LaTeX on 'talk1.tex'
>
```

### Power of R

R is modular – there is a core and you can load packages containing custom functions.

- 2984 packages available on CRAN (02.05.2011) – <http://cran.r-project.org/>
- 998 projects registered on R-Forge (02.05.2011) – <http://r-forge.r-project.org/>
- 460 packages available on BioConductor (03.05.2011) – <http://www.bioconductor.org/>
- from genetics to social sciences and from geology to cryptography

## Installing packages

```
> install.packages("GenABEL")  
> install.packages("DatABEL",  
+ repos="http://R-Forge.R-project.org")
```

## Loading packages

```
> require("GenABEL") # Within functions  
> library("GenABEL")
```

### How to get help

```
> vignette("GenABEL") # Package level
> demo(graphics)
> help(qtscore)        # Function level
> ?qtscore
> ??qtscore           # Extensive search
```

# Introduction to R

## Your own function

### Function for generating random genotypes

```
> generateGenotypes <- function(num.markers = 1, missing = F) {  
+   if (missing) {  
+     genotypes <- sample(1:5, num.markers, replace = T)  
+   }  
+   else {  
+     sample(1:4, num.markers, replace = T) -> genotypes  
+   }  
+   genotypes[genotypes == 1] <- "A"  
+   genotypes[genotypes == 2] <- "T"  
+   genotypes[genotypes == 3] <- "C"  
+   genotypes[genotypes == 4] <- "G"  
+   genotypes[genotypes == 5] <- "X"  
+   genotypes  
+ }
```

```
> generateGenotypes(5)  
[1] "T" "T" "G" "G" "G"  
> generateGenotypes(num.markers = 5)  
[1] "G" "T" "A" "G" "G"  
> generateGenotypes(10, T)  
[1] "C" "C" "A" "T" "A" "A" "G" "A" "G" "A"  
> generateGenotypes(missing = T, num.markers = 10)  
[1] "G" "A" "X" "A" "A" "C" "X" "A" "A" "C"
```



# Introduction to GenABEL

What GenABEL is?

GenABEL project – <http://www.genabel.org>

*The mission of the GenABEL project is to provide a framework for collaborative, sustainable, transparent, open-source based development of statistical genomics methodology. We aim to streamline methodology discussion, development, implementation, dissemination and maintenance; through the community.*

GenABEL is developed by a Team led Dr. Yurii Aulchenko, Erasmus MC, Rotterdam.

# Introduction to GenABEL

The multitude of ABEL packages... (after: [www.genabel.org](http://www.genabel.org))

- **GenABEL** – genome-wide association analysis for quantitative, binary and time-till-event traits.
- **MetABEL** – meta-analysis of genome-wide SNP association results GWAS for quantitative, binary and time-till-event trait.
- **ProbABEL** – genome-wide association analysis of imputed data.
- **PredictABEL** – assess the performance of risk models for binary outcomes.
- **DatABEL** – file-based access to large matrices stored on HDD in binary format.
- **ParallABEL** – generalized parallelization of GWAS.
- **MixABEL** – more mixed models GWAS; experimenting with GSL, multiple input formats, iterator, parallelization through threads.

### Data representation in GenABEL

- **\*.raw** – genotype data GenABEL internal binary format.
- **\*.dat** – phenotype data, e.g., as in PLINK.

Binary format = compression, e.g. for 170K SNP chip 200 individuals: data.ped – 144.4MB vs. data.raw – 32.9MB.

### \*.dat file format

```
id sex age bt1 ct ct1
"289982" 0 30.33 NA NA 3.93
"325286" 0 36.514 1 0.49 3.61
"357273" 1 37.811 0 1.65 5.30
```

### Importing from different data formats

- `convert.snp.text` – convert from text format.
- `convert.snp.ped` – convert from PED format.
- `convert.snp.tped` – convert from TPED format.
- `convert.snp.illumina` – convert from Illumina format.

### Loading data

```
> data <- load.gwa.data("dataset/phenotype.dat",  
+                      "dataset/genotype.raw",  
+                      makemap = T)
```

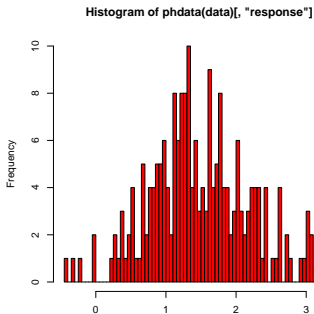
If coords are chromosome specific, you can make them genome-wise by: `makemap = T`.

### Examine the phenotype data

```
> nids(data)
[1] 207
> nsnps(data)
[1] 174375
> phdata(data)[2,]
      id sex bt      ct group response
dog225 dog225  1  0 1.925575      3 1.569402
> phdata(data)[1:5, "sex"]
[1] 1 1 0 1 0
```

### Get summary for trait response

```
> summary(phdata(data)[,"response"])  
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's  
-0.4325  0.9996  1.4090  1.4860  1.9480  3.3860  1.0000  
  
> hist(phdata(data)[,"response"],  
+ breaks = 100,  
+ col = "red")
```



### Examine the genotype data 1st individual, markers 3-5

```
> gtdata(data)[1,3:5]
@nids = 1
@nsnps = 3
@nbytes = 1
@idnames = dog224
@snpnames = BICF2P1383091 TIGRP2P259 BICF2P186608
@chromosome = 1 1 1
@coding = 04 01 01
@strand = 00 00 00
@map = 3212349 3249189 3265742
@male = 1
@gtps =
80 40 40
```

## Get summary for markers 2 and 3

```
> summary(gtdata(data))[2:3,]
```

	Chromosome	Position	Strand	A1	A2	NoMeasured	CallRate	Q.2	P.11
BICF2G630707846	1	3082514	u	1	2	206	0.995169	0.0	206
BICF2P1383091	1	3212349	u	A	G	207	1.000000	0.5	0
	P.12	P.22	Pexact	Fmax	Plrt				
BICF2G630707846	0	0	1.000000e+00	0	1.000000e+00				
BICF2P1383091	207	0	1.240547e-61	-1	2.282010e-64				



## Is the binary trait bt correlated with sex?

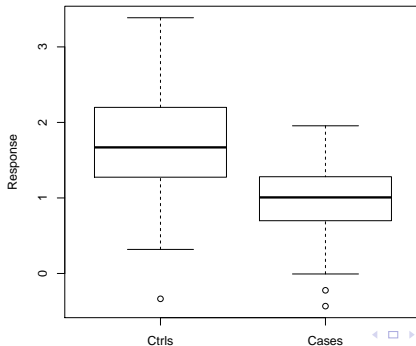
```
> tab <- table(phdata(data)$bt, phdata(data)$sex)
> fisher.test(tab)

Fisher's Exact Test for Count Data

data:  tab
p-value = 1
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.5322721 1.9110534
sample estimates:
odds ratio
 1.010358
```

## Is the binary trait bt related to response?

```
> boxplot(phdata(data)$response ~ phdata(data)$bt,  
+ names = c("Ctrls", "Cases"),  
+ ylab = "Response")
```



## Do a simple QC?

```
> qc1 <- check.marker(data, call = 0.95,  
+ perid.call = 0.95,  
+ maf = 1e-08,  
+ p.lev = 1e-08)  
> ...  
> data.clean <- data[qc1$idok, qc1$snpok]
```

## Do a simple association test

```
> an <- qtscore(bt ~ response, data,
+ trait.type="binomial", times=1)
> summary(an, top=5)
```

Summary for top 5 results, sorted by P1df

	Chromosome	Position	Strand	A1	A2	N	effB	se_effB
BICF2P506952	1	90475257	u	A	G	204	-0.002009766	0.0003574521
BICF2G630348662	3	339590471	u	T	C	204	-0.002009766	0.0003574521
TIGRP2P51678	3	339878416	u	C	T	204	-0.002009766	0.0003574521
BICF2G630348969	3	340378977	u	T	G	204	-0.002009766	0.0003574521
BICF2P628966	3	340641323	u	C	T	204	-0.002009766	0.0003574521

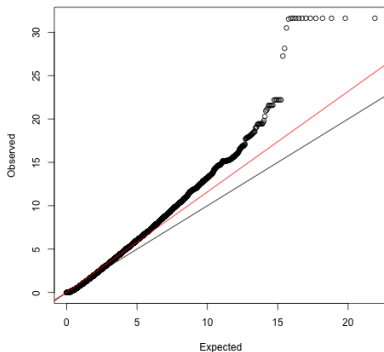
	chi2.1df	P1df	effAB	effBB	chi2.2df	P2df
BICF2P506952	31.61223	1.882407e-08	0	NA	31.61223	1.882407e-08
BICF2G630348662	31.61223	1.882407e-08	0	NA	31.61223	1.882407e-08
TIGRP2P51678	31.61223	1.882407e-08	0	NA	31.61223	1.882407e-08
BICF2G630348969	31.61223	1.882407e-08	0	NA	31.61223	1.882407e-08
BICF2P628966	31.61223	1.882407e-08	0	NA	31.61223	1.882407e-08

	Pc1df
BICF2P506952	1.767672e-07
BICF2G630348662	1.767672e-07
TIGRP2P51678	1.767672e-07
BICF2G630348969	1.767672e-07
BICF2P628966	1.767672e-07

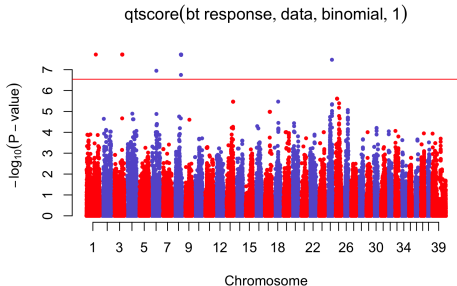
### What is $\lambda$ , show Q-Q plot...

```
> estlambda(an[, "P1df"])  
$estimate  
[1] 1.159153  
  
$se  
[1] 0.0003918843
```



### What about a Manhattan plot?

```
> plot(an,  
+ col = c("red", "slateblue"),  
+ pch = 19,  
+ cex = .5,  
+ df = "1")  
> bonferroni <- -log10(0.05 / nsnps(data))  
> abline(h=bonferroni, col = "red")
```



# GenABEL – why?

Easiness of comparing different approaches...

## Load your data one time and enjoy:

- Simple association tests.
- Genomic control.
- PCA-based correction – Eigenstrat.
- Mixed models.
- Structured association.
- Any combination of the above!

Thank You! and:

- Leif Andersson
- Yurii Aulchenko
- Örjan Carlborg
- Dirk Jan de Koonig
- Kerstin Lindblad-Toh
- Xia Shen
- Katarina Tengvall



## Use account:

- login: Kurs\_LUPAonStatistic
- password: LupaStat2011
- DO NOT TRY TO LOGIN TO VMWARE (Windows) - you will block the whole account!!!

## Website:

<http://www.computationalgenetics.se/LUPA2011>