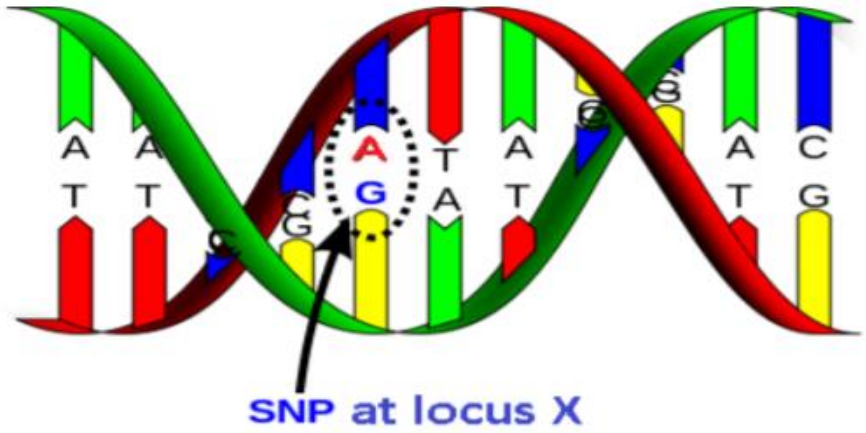
ARCHANA BHARDWAJ

(GWAS : via Multiple approaches)

GBIO0002

Important genetic terms

- Given position in the genome (i.e. locus) has several associated alleles (A and G) which produce genotypes r_A/r_G



- Haplotypes

  - Combination of alleles at different loci

GWAS main philosophy

- **GWAS = Genome Wide Association Studies**
- **IDEA = GWAS involve scan for large number of genetic markers across the whole genome of many individuals to find specific genetic variations associated with the disease and/or other phenotype**
- **Find the genetic variation(s) that contribute(s) and explain(s) complex diseases**

GWAS visually

- GWAS tries to uncover links between genetic basis of the disease

- Which set of SNPs explain the phenotype?

Genotype	Phenotype
ATGCAGTT	control
TTGCAGTT	control
CTGCAGTT	control
ATGCGGTT	case
TTGCGGTT	case
CTGCCGTT	case

SNP

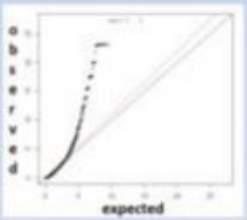
GWAS workflow

Large cohort (>1000) of cases and controls

Get genome information with SNP arrays

Find deviating from expected haplotypes
visualize SNP-SNP interactions using HapMap

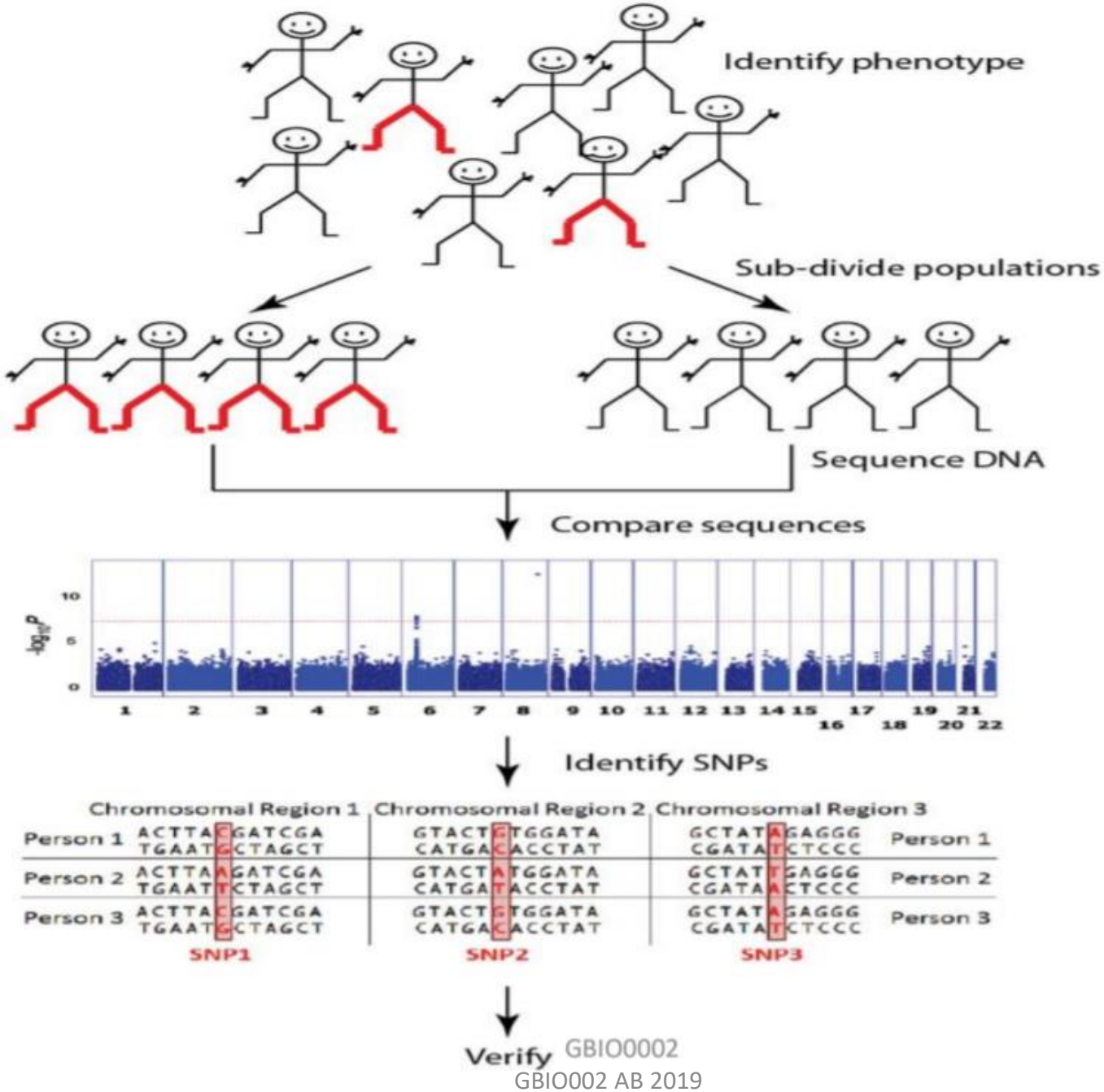
Detection of potential association signals and their fine mapping (e.g. detection of LD, stratification effect)



Replication of detected association in new cohot / subset for validation purposes

	AT	AG	Total
cases observed	35	65	100
contorls observed	125	25	100
Totals	160	90	200

Biological / clinical validation


Identify phenotype
Sub-divide populations
Sequence DNA
Compare sequences
Identify SNPs
Chromosomal Region 1
Chromosomal Region 2
Chromosomal Region 3
Person 1
Person 2
Person 3
SNP1
SNP2
SNP3
Verify

Genotype coding

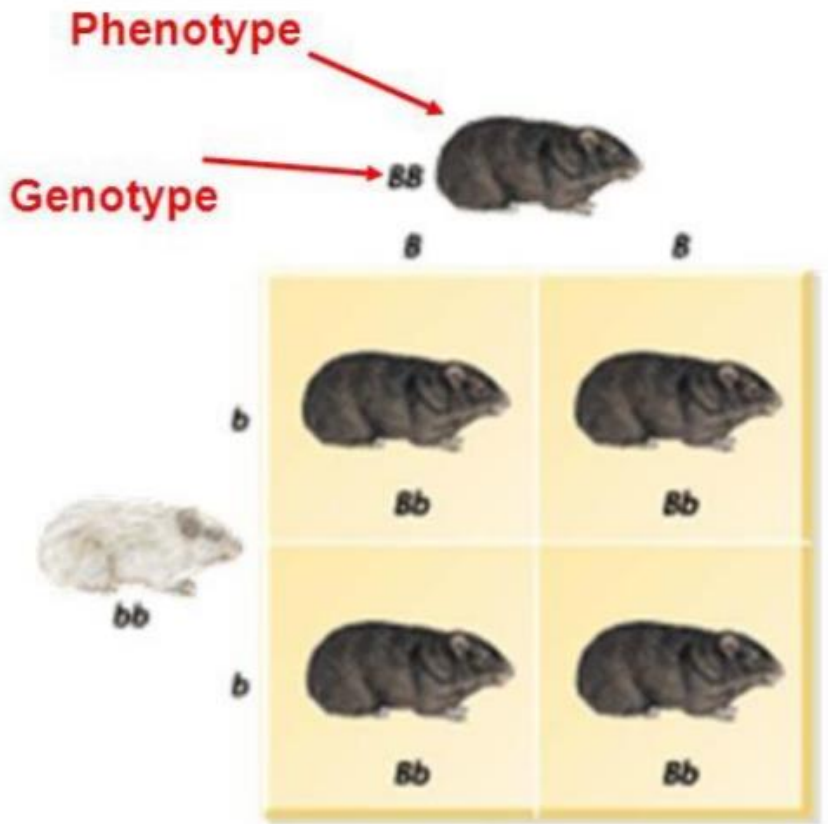
- For given bi-allelic marker/SNP/loci there could be total of 3 possible genotypes given alleles A and a

Genotype	Coding
AA	0
Aa	1
aa	2

Note: A is major allele and a is minor

Relationship between Genotypes and Phenotypes

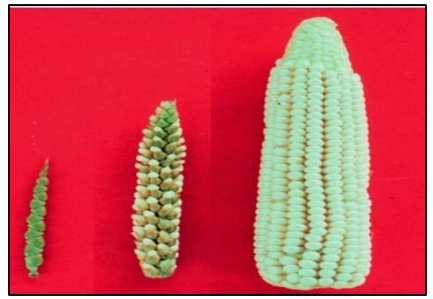
- **Genotype:** Indicates the alleles that the organism has inherited regarding a particular trait.
- **Phenotype:** The actual visible trait of the organism.



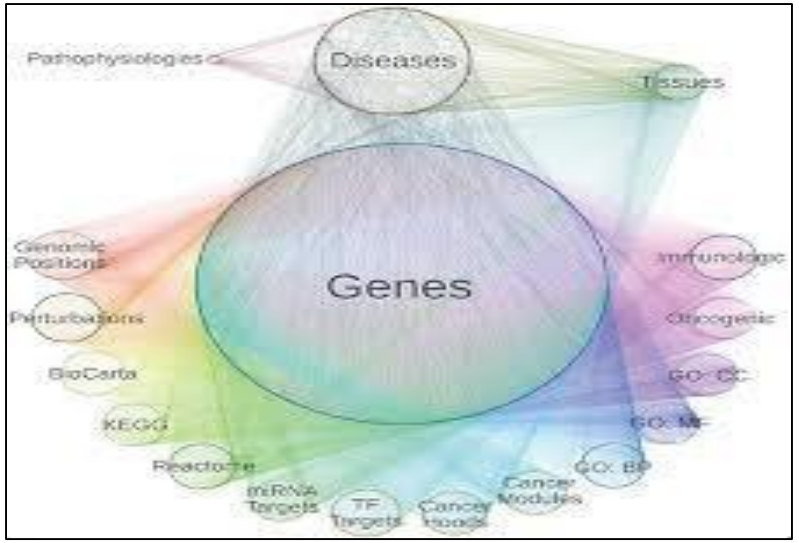
Uses of GWAS

➢Identify genes that are responsible for traits of interest:

- Humans
- Animals
- Plants

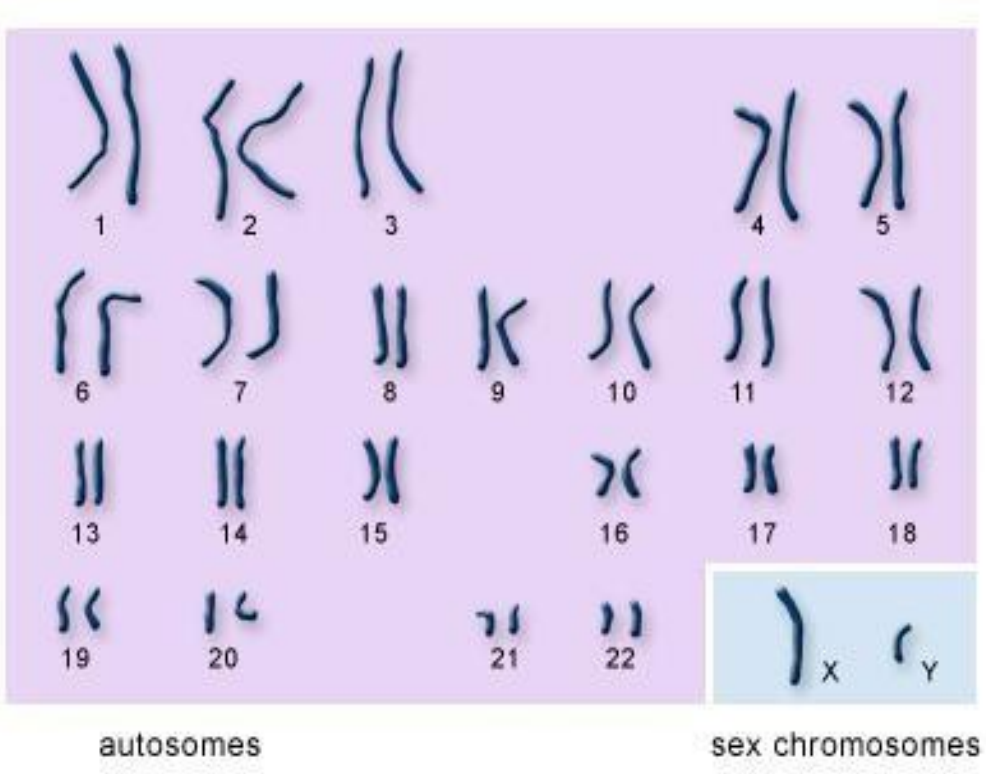


➢Understanding biological mechanisms related to the trait of interest



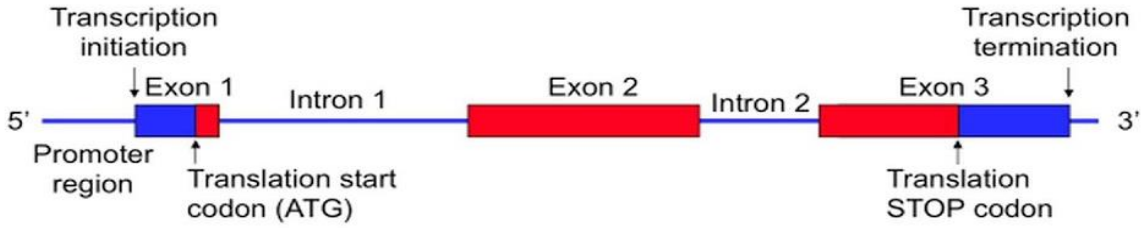
Human Genome Statistics

- Number of Chromosomes : 23 pairs
- Genome Size : 3,079,843,747 Base pairs
- No of Genes : 32,185

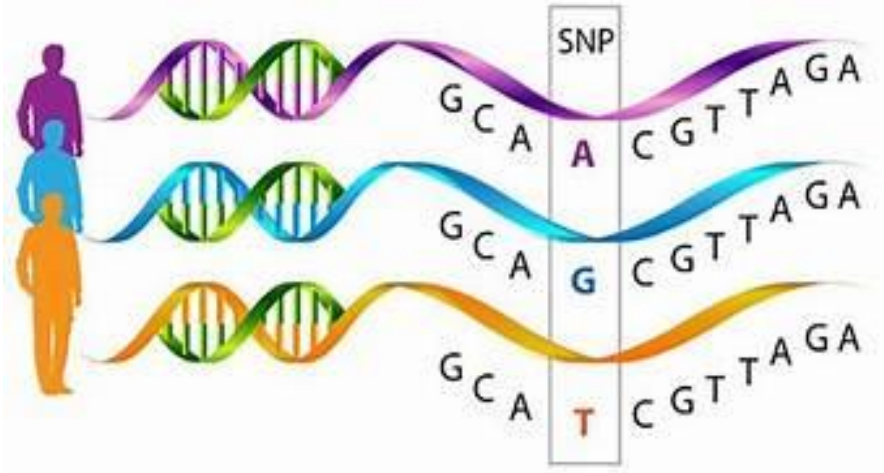


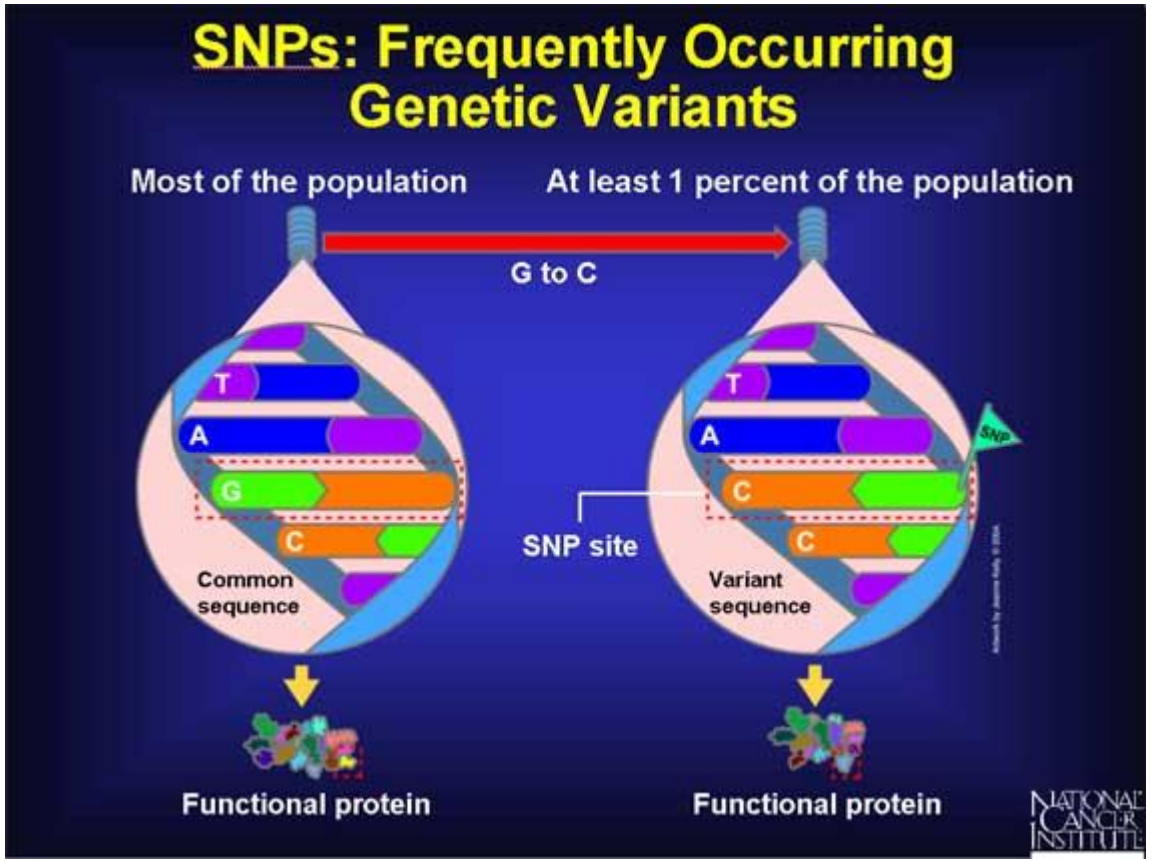
Gene: This is a sequence of nucleotides in the DNA that codes for a molecule (e.g., a protein)

Gene Structure



IMPORTANT FINDING





Let us identify signal (in from of SNPs) from GWAS DATA

PLINK SESSION

- Data Preparation
- Quality Control
- Clustering
- GWAS

PLINK: Introduction

- PLINK is whole genome association analysis tool
- PLINK has a well-documented manual to explain all features
- PLINK is available for Linux, Mac OS and MS-DOS
- gPLINK is the other version of PLINK that provides the graphical user interface
- Command line version is faster than graphical PLINK

PLINK: Download

- To download PLINK:
- http://zzz.bwh.harvard.edu/plink/download.shtml#download
- Uncompress the PLINK-1.07-dos.zip
- Click on the folder. There are two files
  - – test.map contains the marker information
  - – test.ped contains genotype data and sample information

PLINK: File Formats(1/2)

- MAP Format

Each line of the MAP file describes a single marker and must contain exactly 4 columns:

- chromosome (1-22, X, Y or 0 if unplaced)
- rs# or snp identifier
- Genetic distance (morgans)
- Base-pair position (bp units)

Column1 Column2 Column3 Column4

```
1 snp1 0 1
1 snp2 0 2
```

- **A centimorgan, also known and written as a genetic map unit (gmu), is, at heart, a unit of probability. One cM is equal to the distance of two genes that gives a recombination frequency of one percent.**

PLINK: File Formats(2/2)

- PED Format. This file is a white-space (space or tab) delimited file: the first six columns are mandatory:

Family ID, Individual ID,Paternal ID,Maternal ID,Sex (1=male; 2=female; other=unknown),phenotype

Column1	Column2	Column3	Column4	Column5	Column6				
1	1	0	0	1	1	A	A	G	T
2	1	0	0	1	1	A	C	T	G
3	1	0	0	1	1	C	C	G	G
4	1	0	0	1	2	A	C	T	T
5	1	0	0	1	2	C	C	G	T
6	1	0	0	1	2	C	C	T	T

- **Binary format:**
  -> BED, BIM, and FAM

- **Transposed text format :**
  and TFAM

We will be using BED, BIM and FAM in coming session

PLINK: File Formats (Example)

*.ped

FID	IID	PID	MID	Sex	P	rs1	rs2	rs3
1	1	0	0	2	1	CT	AG	AA
2	2	0	0	1	0	CC	AA	AC
3	3	0	0	1	1	CC	AA	AC

*.map

Chr	SNP	GD	BPP
1	rs1	0	870000
1	rs2	0	880000
1	rs3	0	890000

*.fam

FID	IID	PID	MID	Sex	P
1	1	0	0	2	1
2	2	0	0	1	0
3	3	0	0	1	1

*.bed

Contains binary version of the SNP info of the *.ped file. (not in a format readable for humans)

*.bim

Chr	SNP	GD	BPP	Allele 1	Allele 2
1	rs1	0	870000	C	T
1	rs2	0	880000	A	G
1	rs3	0	890000	A	C

Covariate file

FID	IID	C1	C2	C3
1	1	0.00812835	0.00606235	-0.000871105
2	2	-0.0600943	0.0318994	-0.0827743
3	3	-0.0431903	0.00133068	-0.000276131

Legend			
FID	Family ID	rs{x}	Alleles per subject per SNP
IID	Individual ID	Chr	Chromosome
PID	Paternal ID	SNP	SNP name
MID	Maternal ID	GD	Genetic distance (morgans)
Sex	Sex of subject	BPP	Base-pair position (bp units)
P	Phenotype	C{x}	Covariates (e.g., Multidimensional Scaling (MDS) components)

PLINK: Data input (map and ped)

- **Type command :**

  plink --file test –noweb

Provide you detail information of SNPs count, individuals count and check for the missingness and frequency test

```
Options in effect:
        --file test
        --noweb

2 (of 2) markers to be included from [ test.map ]
6 individuals read from [ test.ped ]
6 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
3 cases, 3 controls and 0 missing
6 males, 0 females, and 0 of unspecified sex
Before frequency and genotyping pruning, there are 2 SNPs
6 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 1
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 2 SNPs
After filtering, 3 cases, 3 controls and 0 missing
After filtering, 6 males, 0 females, and 0 of unspecified sex
```

PLINK: Data input (bed format)

- **Convert map and ped into binary format**

  - *plink.exe --file test --make-bed --out test --noweb*

- **Read binary file in plink**

  - *plink.exe –bfile test*

```
Reading map (extended format) from [ test.bim ]
2 markers to be included from [ test.bim ]
Reading pedigree information from [ test.fam ]
6 individuals read from [ test.fam ]
6 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
3 cases, 3 controls and 0 missing
6 males, 0 females, and 0 of unspecified sex
Reading genotype bitfile from [ test.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 2 SNPs
6 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 1
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 2 SNPs
After filtering, 3 cases, 3 controls and 0 missing
After filtering, 6 males, 0 females, and 0 of unspecified sex

Analysis finished: Sun Oct 20 18:49:10 2019
```

Comparision

```
2 (of 2) markers to be included from [ test.map ]
6 individuals read from [ test.ped ]
6 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
3 cases, 3 controls and 0 missing
6 males, 0 females, and 0 of unspecified sex
Before frequency and genotyping pruning, there are 2 SNPs
6 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 1
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 2 SNPs
After filtering, 3 cases, 3 controls and 0 missing
After filtering, 6 males, 0 females, and 0 of unspecified sex
```

TEXT output

```
Reading map (extended format) from [ test.bim ]
2 markers to be included from [ test.bim ]
Reading pedigree information from [ test.fam ]
6 individuals read from [ test.fam ]
6 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
3 cases, 3 controls and 0 missing
6 males, 0 females, and 0 of unspecified sex
Reading genotype bitfile from [ test.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 2 SNPs
6 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 1
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 2 SNPs
After filtering, 3 cases, 3 controls and 0 missing
After filtering, 6 males, 0 females, and 0 of unspecified sex

Analysis finished: Sun Oct 20 18:49:10 2019
```

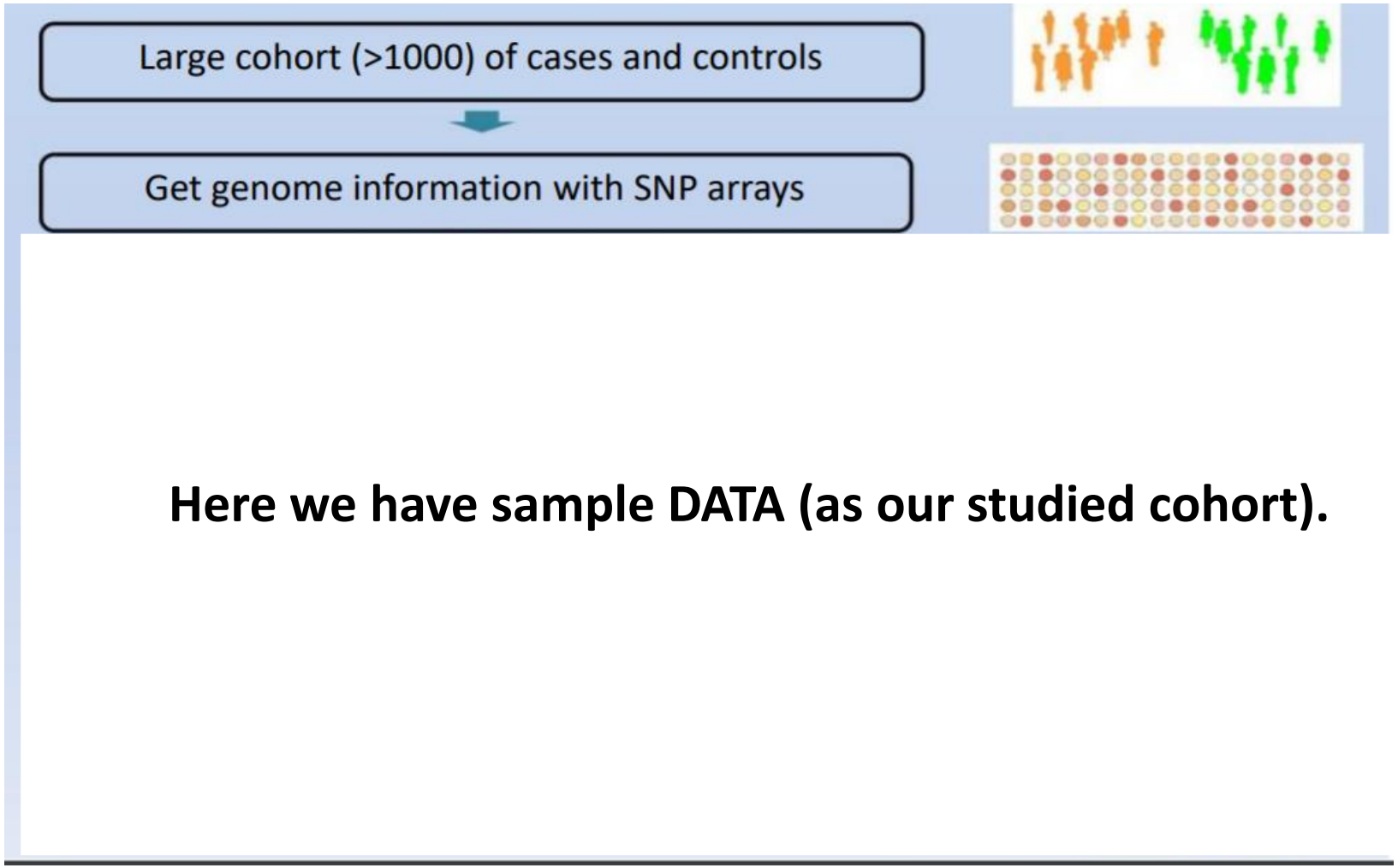
BED OUTPUT

SIMILAR Conclusion

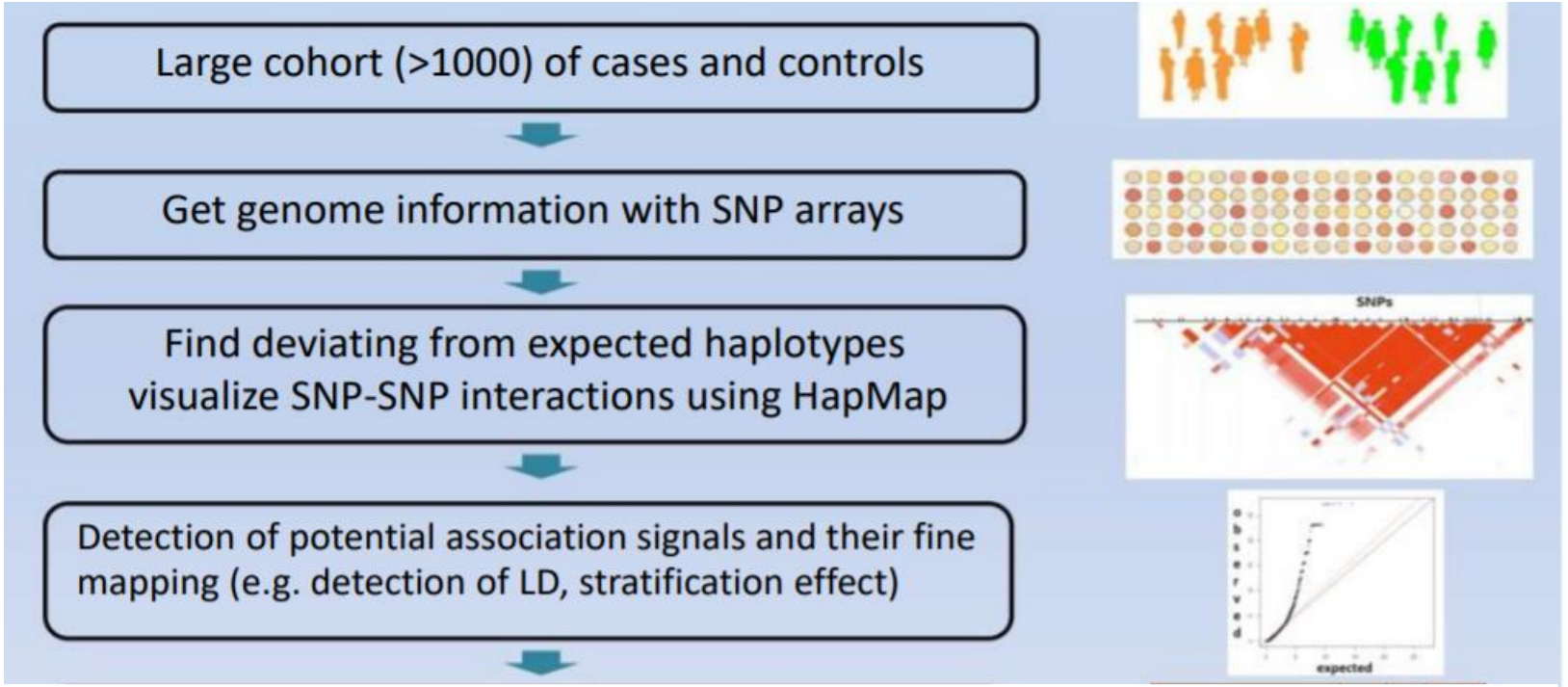
Example data

- Download the example data from the course website **(PLINK FOLDER)**
  - **TSI_JPT_chr20_case_control.bed**
  - **TSI_JPT_chr20_case_control.bim**
  - **TSI_JPT_chr20_case_control.fam**
  - **TSI_JPT_chr20_pheno_header.txt**
  - **TSI_JPT_chr20_pheno.txt**

By looking into file extension, BED FORMAT



Here we have sample DATA (as our studied cohort).

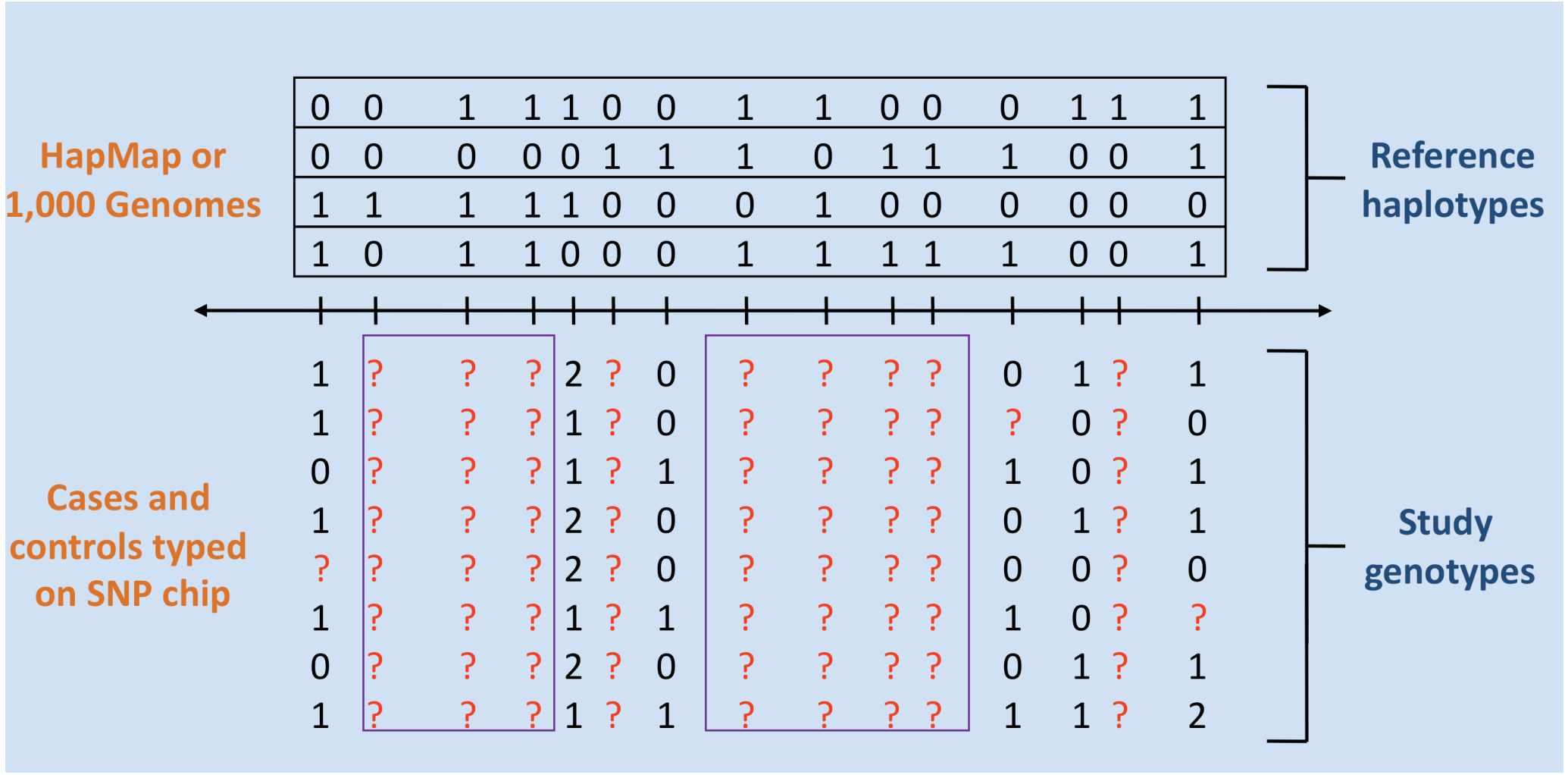


Detection of LD, population stratification (comes under Filteration step)
Lets Perform Quality filteration

Quality control processes

- Missing genotype
- Hardy-Weinberg Equilibrium
- Minor Allele frequency
- Linkage disequilibrium pruning

Missing genotype



Missing Genotypes

- To generate a list genotyping/missingness rate statistics:

  - *plink.exe --bfile TSI_JPT_chr20_case_control --missing --noweb*

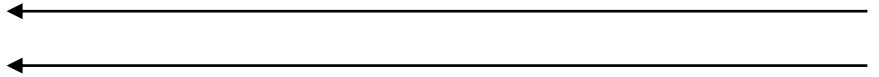
    - **plink.imiss**
    - **plink.lmiss**

    } **OUTPUT in running directory**

- It provides the detail missingness by individual and by SNP (locus), respectively.

Lets read log file

```
Reading map (extended format) from [ TSI_JPT_chr20_case_control.bim ]
36302 markers to be included from [ TSI_JPT_chr20_case_control.bim ]
Reading pedigree information from [ TSI_JPT_chr20_case_control.fam ]
174 individuals read from [ TSI_JPT_chr20_case_control.fam ]
174 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
88 cases, 86 controls and 0 missing
88 males, 86 females, and 0 of unspecified sex
Reading genotype bitfile from [ TSI_JPT_chr20_case_control.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 36302 SNPs
174 founders and 0 non-founders found
Writing individual missingness information to [ plink.imiss ]
Writing locus missingness information to [ plink.lmiss ]
Total genotyping rate in remaining individuals is 0.998399
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 36302 SNPs
After filtering, 88 cases, 86 controls and 0 missing
After filtering, 88 males, 86 females, and 0 of unspecified sex
```



Clustering based on Missing Genotypes

- **Systematic batch effects that induce missingness in parts of the sample will induce correlation between the patterns of missing data that different individuals display**

- **One approach to detecting correlation in these patterns, that might possibly idenity such biases, is to cluster individuals based on their identityby-missingness (IBM).**

  - *plink.exe --bfile TSI_JPT_chr20_case_control --cluster-missing --noweb*

- **which creates the files:**

  - ***plink.matrix.missing***
  - ***plink.cluster3.missing***

which have similar formats to the corresponding IBS clustering files.

Missing Rate Per Person

- The initial step in all data analysis is to exclude individuals with too much missing genotype data. This option is set as follows:

➢ *plink.exe --bfile TSI_JPT_chr20_case_control --mind 0.1 --noweb*

which means exclude with more than 10% missing genotypes.

- A line in the terminal output will appear, indicating how many individuals were removed due to low genotyping. If any individuals were removed, a file called **plink.irem** will be created, listing the Family and Individual IDs of these removed individuals.

Missing Rate Per SNP

- Subsequent analyses can be set to automatically exclude SNPs on the basis of missing genotype rate, with the --geno option: the default is to include all SNPS (i.e. --geno 1).
- To include only SNPs with a 90% genotyping rate (10% missing) use

➢ *plink.exe --bfile TSI_JPT_chr20_case_control --geno 0.1 --noweb*

- As with the --maf option, these counts are calculated after removing individuals with high missing genotype rates.

Hardy-Weinberg Equilibrium (1/2)

▪To generate a list of genotype counts and Hardy-Weinberg test statistics for each SNP, use the option:

• *plink.exe --bfile TSI_JPT_chr20_case_control --hardy --noweb*

which creates a file: **plink.hwe.** The file has the following format

SNP SNP identifier
TEST Code indicating sample
A1 Minor allele code
A2 Major allele code
GENO Genotype counts:11/12/22
O(HET) observed hetrozygosity
E(HET) Expected hetrozygosity
P H-W p-value

Hardy-Weinberg Equilibrium (2/2)

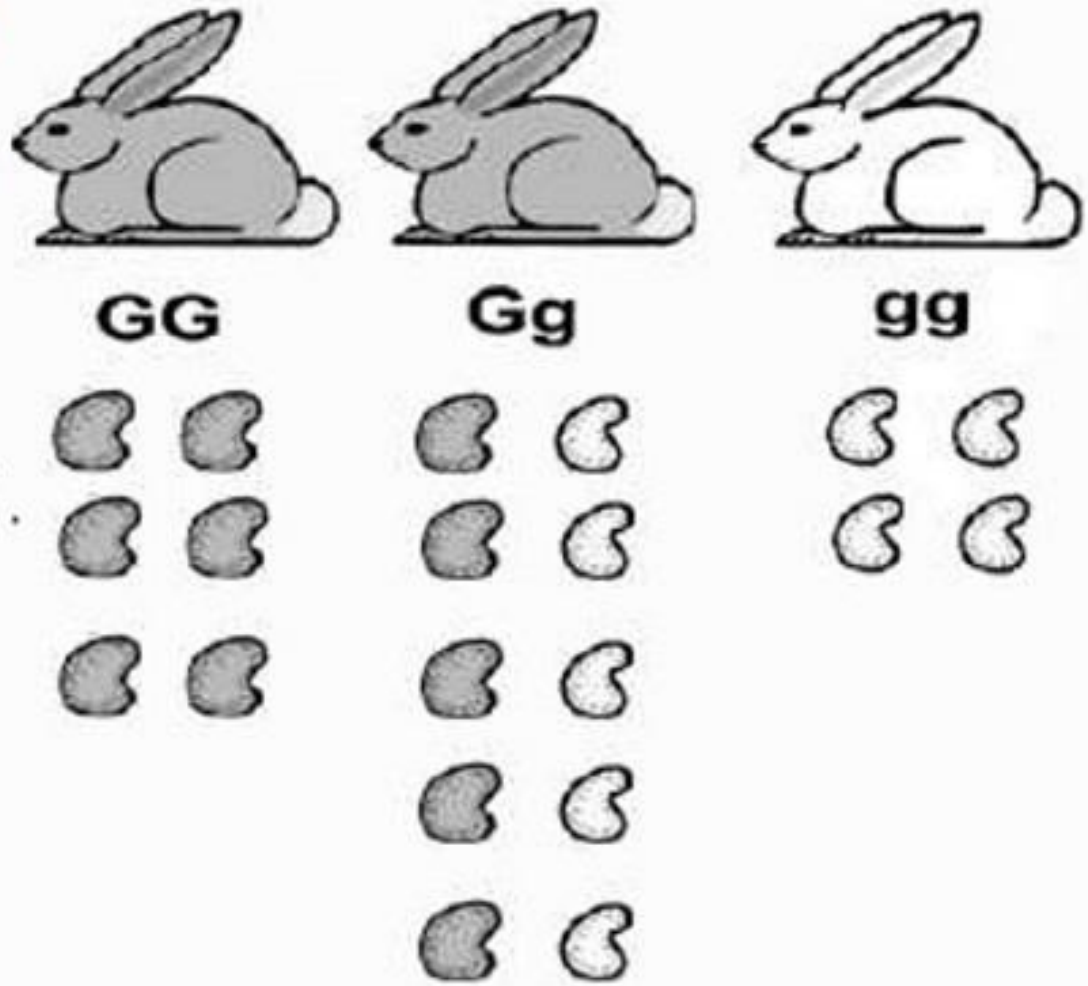
- To exclude markers that failure the Hardy-Weinberg test at a specified significance threshold, use the option:
  - plink.exe --bfile TSI_JPT_chr20_case_control --hwe 0.001 --noweb
- By default this filter uses an exact test. The standard asymptotic (1 df genotypic chi-squared test) can be requested with the --hwe2 option instead of --hwe.
- The following output will appear in the console window and in **plink.log,** detailing how many SNPs failed the Hardy-Weinberg test, for the sample as a whole, and (when PLINK has detected a disease phenotype) for cases and controls separately:

```
Reading map (extended format) from [ TSI_JPT_chr20_case_control.bim ]
36302 markers to be included from [ TSI_JPT_chr20_case_control.bim ]
Reading pedigree information from [ TSI_JPT_chr20_case_control.fam ]
174 individuals read from [ TSI_JPT_chr20_case_control.fam ]
174 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
88 cases, 86 controls and 0 missing
88 males, 86 females, and 0 of unspecified sex
Reading genotype bitfile from [ TSI_JPT_chr20_case_control.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 36302 SNPs
174 founders and 0 non-founders found
18 markers to be excluded based on HWE test ( p <= 0.001 )
        20 markers failed HWE test in cases
        18 markers failed HWE test in controls
Total genotyping rate in remaining individuals is 0.998399
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 36284 SNPs
After filtering, 88 cases, 86 controls and 0 missing
After filtering, 88 males, 86 females, and 0 of unspecified sex
```

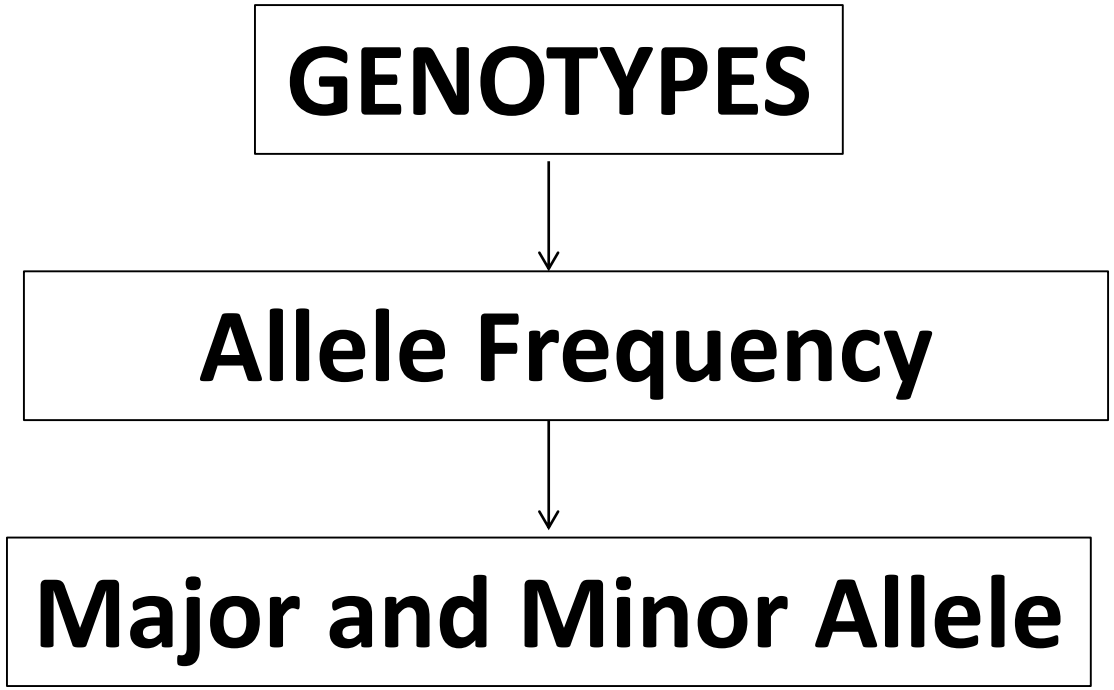
Input data

Allele Frequency

how often an form of a gene shows up in a population over several generations



the number of copies of a particular allele divided by the number of copies of all alleles at the genetic place in a population.


GENOTYPES
Allele Frequency
Major and Minor Allele

Allele Frequency

- To generate a list of minor allele frequencies (MAF) for each SNP, based on all founders in the sample:
  - *plink.exe --bfile TSI_JPT_chr20_case_control --freq --noweb*

- This will create a file: **plink.frq** with five columns:

  CHR Chromosome
  SNP SNP identifier
  A1 Allele 1 code (minor allele)
  A2 Allele 2 code (major allele)
  MAF Minor allele frequency
  NCHROBS Non-missing allele count

Minor Allele Frequency

- Once individuals with too much missing genotype data have been excluded, subsequent analyses can be set to automatically exclude SNPs on the basis of MAF (minor allele frequency):
  - *plink --bfile TSI_JPT_chr20_case_control --maf 0.05 --noweb*
- It means only include SNPs with MAF >= 0.05. The default value is 0.01. This quantity is based only on founders (i.e. individuals for whom the paternal and maternal individual codes and both 0).

option is appropriately counts alleles for X chromosome SNPs.

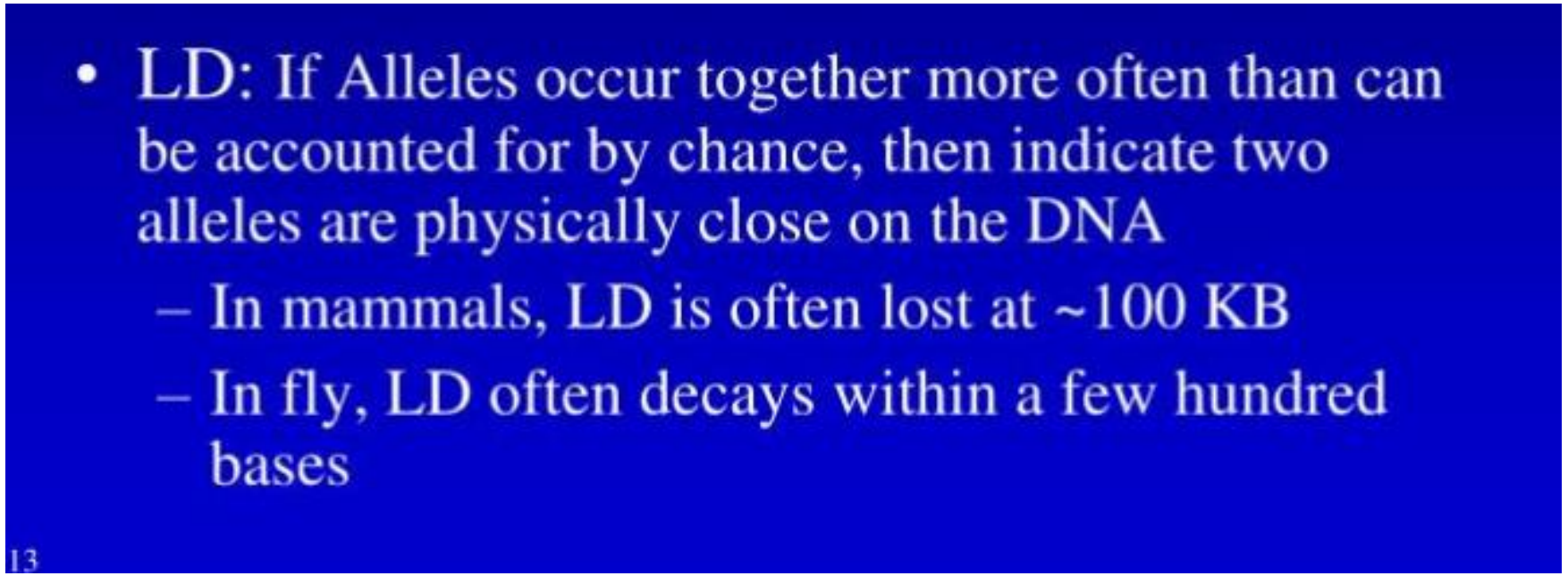
```
Reading map (extended format) from [ TSI_JPT_chr20_case_control.bim ]
36302 markers to be included from [ TSI_JPT_chr20_case_control.bim ]
Reading pedigree information from [ TSI_JPT_chr20_case_control.fam ]
174 individuals read from [ TSI_JPT_chr20_case_control.fam ]
174 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
88 cases, 86 controls and 0 missing
88 males, 86 females, and 0 of unspecified sex
Reading genotype bitfile from [ TSI_JPT_chr20_case_control.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 36302 SNPs
174 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 0.998399
0 SNPs failed missingness test ( GENO > 1 )
7826 SNPs failed frequency test ( MAF < 0.05 )
After frequency and genotyping pruning, there are 28476 SNPs
After filtering, 88 cases, 86 controls and 0 missing
After filtering, 88 males, 86 females, and 0 of unspecified sex
```

Input data

Count SNPs under MAF < 0.01 ?

```
Reading map (extended format) from [ TSI_JPT_chr20_case_control.bim ]
36302 markers to be included from [ TSI_JPT_chr20_case_control.bim ]
Reading pedigree information from [ TSI_JPT_chr20_case_control.fam ]
174 individuals read from [ TSI_JPT_chr20_case_control.fam ]
174 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
88 cases, 86 controls and 0 missing
88 males, 86 females, and 0 of unspecified sex
Reading genotype bitfile from [ TSI_JPT_chr20_case_control.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 36302 SNPs
174 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 0.998399
0 SNPs failed missingness test ( GENO > 1 )
4625 SNPs failed frequency test ( MAF < 0.01 )
After frequency and genotyping pruning, there are 31677 SNPs
After filtering, 88 cases, 86 controls and 0 missing
After filtering, 88 males, 86 females, and 0 of unspecified sex
```

Input data



Linkage disequilibrium (LD): This is a measure of non- random association between alleles at different loci at the same chromosome in a given population. SNPs are in LD when the frequency of association of their alleles is higher than expected under random assortment. LD concerns patterns of correlations between SNPs.

Linkage disequilibrium pruning (1/2)

- Sometimes it is useful to generate a pruned subset of SNPs that are in approximate linkage equilibrium with each other. This can be achieved via two commands:

--indep which prunes based on the variance inflation factor (VIF), which recursively removes SNPs within a sliding window;

-- indep-pairwise which is similar, except it is based only on pairwise genotypic correlation.

- The VIF pruning routine is performed:

Linkage disequilibrium pruning (2/2)

- Each is a simlpe list of SNP IDs; both these files can subsequently be specified as the argument for a --extract or --exclude command.

- The parameters for --indep are: window size in SNPs (e.g. 50), the number of SNPs to shift the window at each step (e.g. 5), the VIF threshold. The VIF is 1/(1-R^2) where R^2 is the multiple correlation coefficient for a SNP being regressed on all other SNPs simultaneously.

- That is, this considers the correlations between SNPs but also between linear combinations of SNPs.

➢ *plink.exe --bfile TSI_JPT_chr20_case_control --indep 50 5 2 --noweb*

```
Reading map (extended format) from [ TSI_JPT_chr20_case_control.bim ]
36302 markers to be included from [ TSI_JPT_chr20_case_control.bim ]
Reading pedigree information from [ TSI_JPT_chr20_case_control.fam ]
174 individuals read from [ TSI_JPT_chr20_case_control.fam ]
174 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
88 cases, 86 controls and 0 missing
88 males, 86 females, and 0 of unspecified sex
Reading genotype bitfile from [ TSI_JPT_chr20_case_control.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 36302 SNPs
174 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 0.998399
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 36302 SNPs
After filtering, 88 cases, 86 controls and 0 missing
After filtering, 88 males, 86 females, and 0 of unspecified sex
Performing LD-based pruning...
Writing pruned-in SNPs to [ plink.prune.in ]
Writing pruned-out SNPs to [ plink.prune.out ]
Scanning from chromosome 20 to 20
```

Input data

```
Scan region on chromosome 20 from [ rs4814683 ] to [ rs6011394 ]
For chromosome 20, 29618 SNPs pruned out, 6684 remaining
```

How many snp in LD with window size “100”, “200” ?

clustering

plink.exe --bfile TSI_JPT_chr20_case_control --cluster --noweb

which generates four output files:

- **plink.cluster0**
- **plink.cluster1**
- **plink.cluster2**
- **plink.cluster3**

that contain similar information but in different formats. The
The *.cluster0 file contains some information on the clustering process. This file can be safely ignored by most users.
The *.cluster1 file contains information on the final solution, listed by cluster.
The *.cluster2 file contains the same information but listed one line per individual

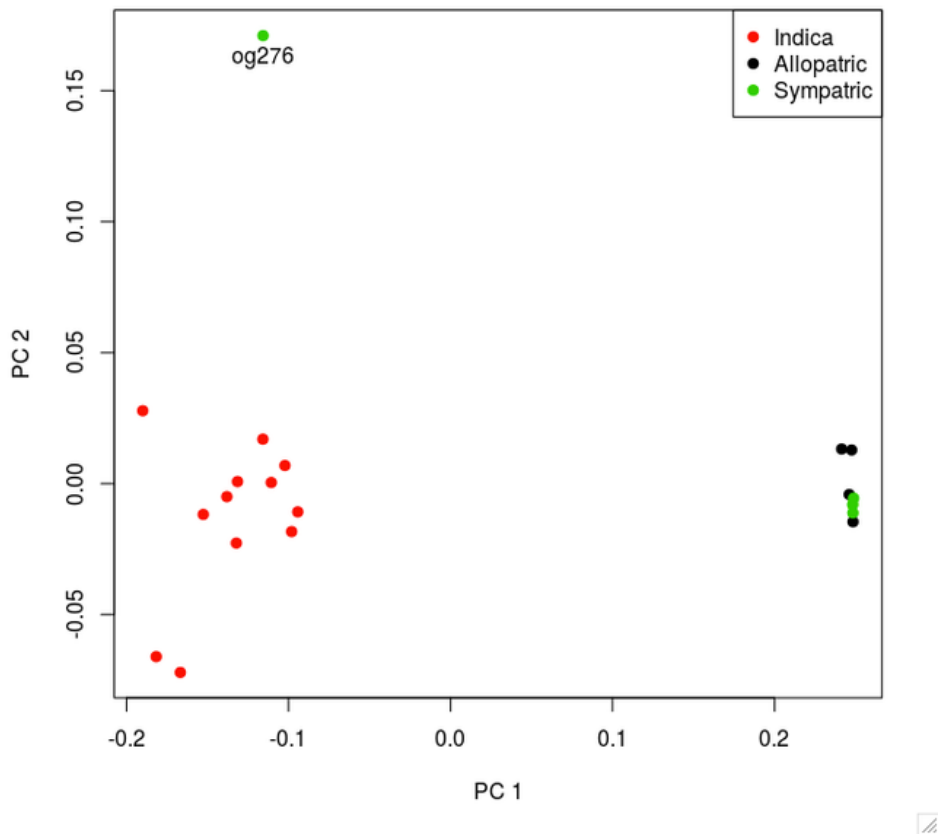
The *.cluster3 file is in the same format as cluster2 (one line per individual) but contains all solutions (i.e. every step of the clustering from moving from N clusters each of 1 individual (leftmost column after family and individual ID) to 1 cluster (labelled 0) containing all N individuals (the final, rightmost column)

Plink.cluster1

```
SOL-0     NA18946_NA18946 NA18955_NA18955 NA19087_NA19087 NA19002_NA19002
```

There is only one cluster.

What if we have more than one cluster?



Three cluster

We will perform this analysis in other R package

Association Analysis

- Case/control
- Multiple-testing correction

Basic case/control association test

To perform a standard case/control association analysis, use the option:

plink.exe --bfile TSI_JPT_chr20_case_control --assoc --noweb

which generates a file

plink.assoc

which contains the fields:

CHR	Chromosome
SNP	SNP ID
BP	Physical position (base-pair)
A1	Minor allele name (based on whole sample)
F_A	Frequency of this allele in cases
F_U	Frequency of this allele in controls
A2	Major allele name
CHISQ	Basic allelic test chi-square (1df)
P	Asymptotic p-value for this test
OR	Estimated odds ratio (for A1, i.e. A2 is reference)

plink.exe --bfile TSI_JPT_chr20_case_control –assoc --noweb

```
Reading map (extended format) from [ TSI_JPT_chr20_case_control.bim ]
36302 markers to be included from [ TSI_JPT_chr20_case_control.bim ]
Reading pedigree information from [ TSI_JPT_chr20_case_control.fam ]
174 individuals read from [ TSI_JPT_chr20_case_control.fam ]
174 individuals with nonmissing phenotypes
Assuming a disease phenotype (1=unaff, 2=aff, 0=miss)
Missing phenotype value is also -9
88 cases, 86 controls and 0 missing
88 males, 86 females, and 0 of unspecified sex
Reading genotype bitfile from [ TSI_JPT_chr20_case_control.bed ]
Detected that binary PED file is v1.00 SNP-major mode
Before frequency and genotyping pruning, there are 36302 SNPs
174 founders and 0 non-founders found
Total genotyping rate in remaining individuals is 0.998399
0 SNPs failed missingness test ( GENO > 1 )
0 SNPs failed frequency test ( MAF < 0 )
After frequency and genotyping pruning, there are 36302 SNPs
After filtering, 88 cases, 86 controls and 0 missing
After filtering, 88 males, 86 females, and 0 of unspecified sex
Writing main association results to [ plink.assoc ]

Analysis finished: Sun Oct 20 20:27:13 2019
```

Input data

Adjustment for multiple testing

To generate a file of adjusted significance values that correct for all tests performed and other metrics, use the option:

plink.exe --bfile TSI_JPT_chr20_case_control --assoc --adjust --noweb

which generates the file

plink.adjust

which contains the fields

CHR	Chromosome number
SNP	SNP identifer
UNADJ	Unadjusted p-value
GC	Genomic-control corrected p-values
BONF	Bonferroni single-step adjusted p-values
HOLM	Holm (1979) step-down adjusted p-values
SIDAK_SS	Sidak single-step adjusted p-values
SIDAK_SD	Sidak step-down adjusted p-values
FDR_BH	Benjamini & Hochberg (1995) step-up FDR control
FDR_BY	Benjamini & Yekutieli (2001) step-up FDR control

This file is sorted by significance value rather than genomic location, the most significant results being at the top.

1. Open plink.assoc.adjusted
2. Sort column « BONF »
3. Count SNPs has p value 0.05, 0.01?

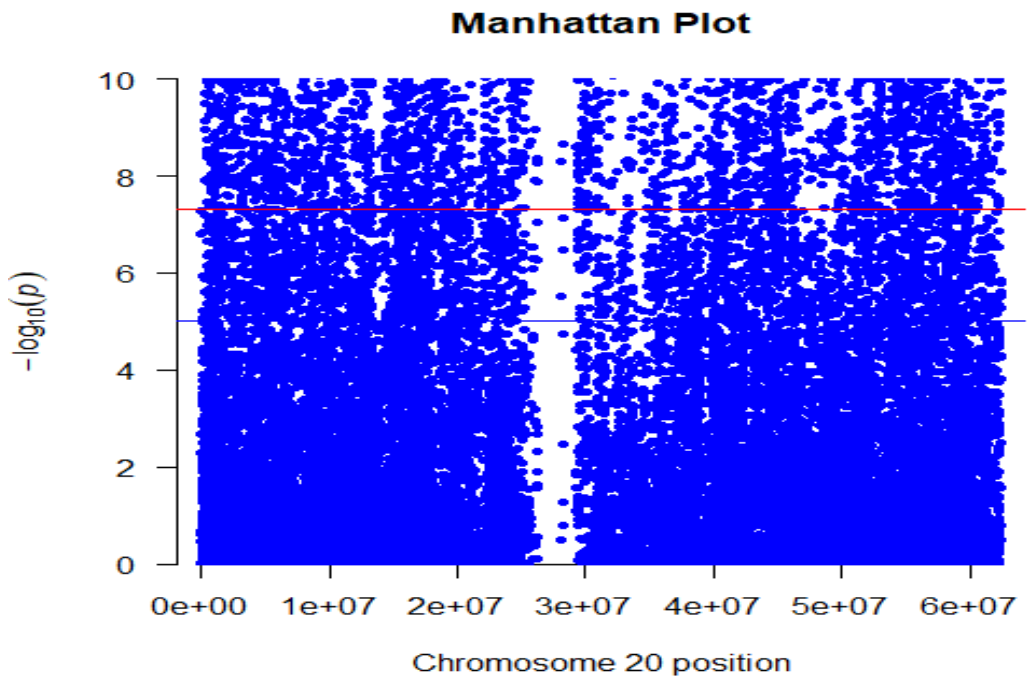
Let us visualize GWAS result

LETS INSTALL R Pakcage

1. Open R window
2. Install.packages(qqman)
3. Load in library

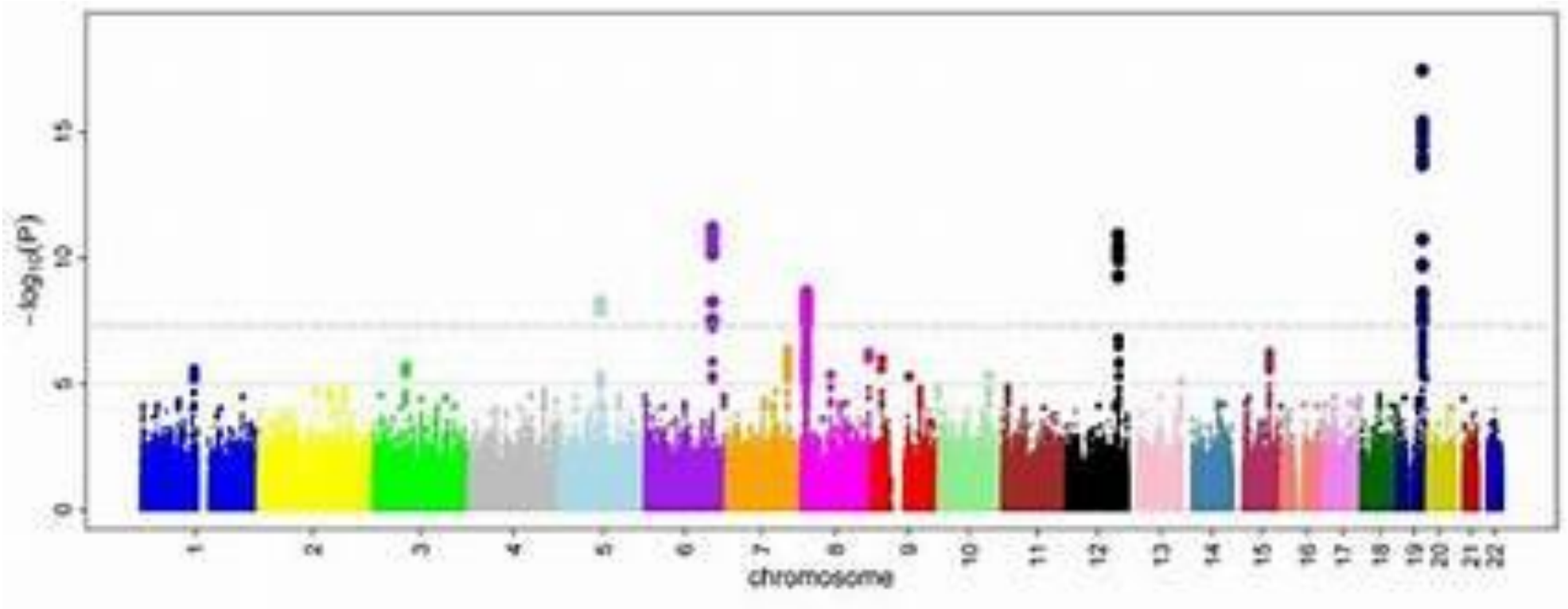
library(qqman)

- *gwas <- data.frame(read.table(file="plink.assoc",header=TRUE))*
- *manhattan(gwas, main = "Manhattan Plot", ylim = c(0, 10),col="blue")*



It has data from Chr 20

Manhattan (multi chromosome view)



To work on cluster, we have Storage issue.

We will be working on small dataset.

It will have SNPs information for 1 chromosome.

R Session (multiple packages)

- **Data Preparation**
- **Preprocessing (similar to PLINK)**
- **ANALYSIS : (similar to PLINK)**
  - **Principal Component Analysis**
  - **Genome-Wide Association**
  - **Functional insights into candidate markers**

Introduction to GenABEL (1/2)

- This library allows to do complete GWAS workflow
- GWAS data and corresponding attributes (SNPs, phenotype, sex, etc.) are stored in data object gwas.data-class
- The object attributes could be accessed with @

  - phenotype data: gwaa_object@phdata
  - number of people in study: gwaa_object@gtdata@nids

Introduction to GenABEL(2/2)

- number of SNPs: gwaa_object@gtdata@nsnps

- SNP names: gwaa_object@gtdata@snpnames

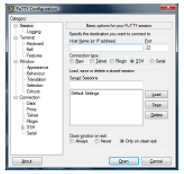
- Chromosome labels:gwaa_object@gtdata@chromosome

- SNPs map positions: gwaa_object@gtdata@map

- **GenABEL is an R library developed to facilitate Genome-Wide Association analysis of binary and quantitative traits.**
- **Features of GenABEL :**
  - **specific facilities for storage and manipulation of large data**
  - **QC**
  - **Maximum Likelihood estimation of linear, logistic and Cox regression on Genome-wide scale**
  - **Specific functions to analyze and display the results**

CONNECT TO SERVER (For Windows)

1. GO to Link https://putty.org/ Click to download

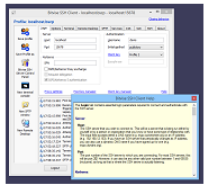


Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. PuTTY is open source software that is available with source code and is developed and supported by a group of volunteers.

You can download PuTTY here.

Below suggestions are independent of the authors of PuTTY. They are *not* to be seen as endorsements by the PuTTY project.

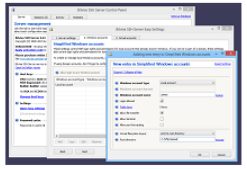


Bitvise SSH Client

Bitvise SSH Client is an SSH and SFTP client for Windows. It is developed and supported professionally by Bitvise. The SSH Client is robust, easy to install, easy to use, and supports all features supported by PuTTY, as well as the following:

- graphical SFTP file transfer;
- single-click Remote Desktop tunneling;
- auto-reconnecting capability;
- dynamic port forwarding through an integrated proxy;
- an FTP-to-SFTP protocol bridge.

Bitvise SSH Client is **free to use**. You can download it here.



Bitvise SSH Server

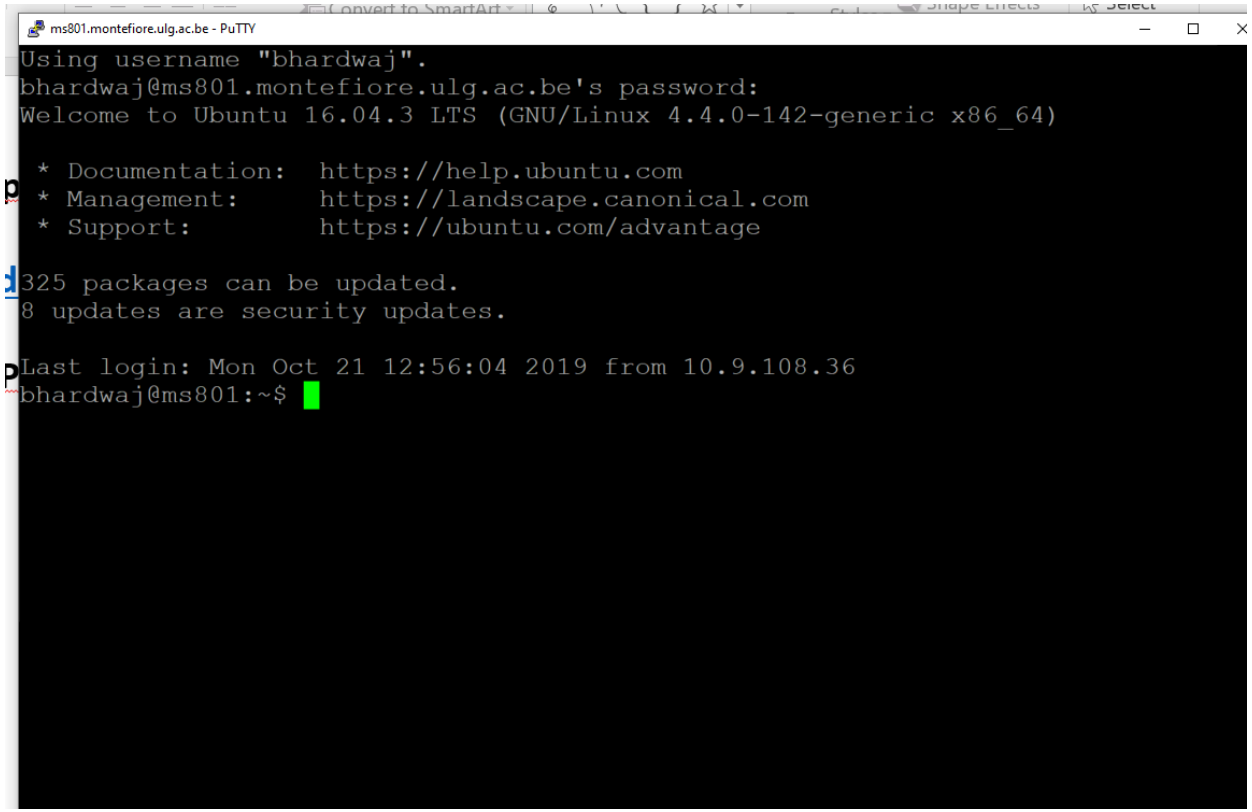
Bitvise SSH Server is an SSH, SFTP and SCP server for Windows. It is robust, easy to install, easy to use, and works well with a variety of SSH clients, including Bitvise SSH Client, OpenSSH, and PuTTY. The SSH Server is developed and supported professionally by Bitvise.

You can download Bitvise SSH Server here.

CONNECT TO SERVER

- Open putty
- User id : username@ms801.montefiore.ulg.ac.be
- Enter Password

ms801 to ms818



CONNECT TO SERVER (For Unix)

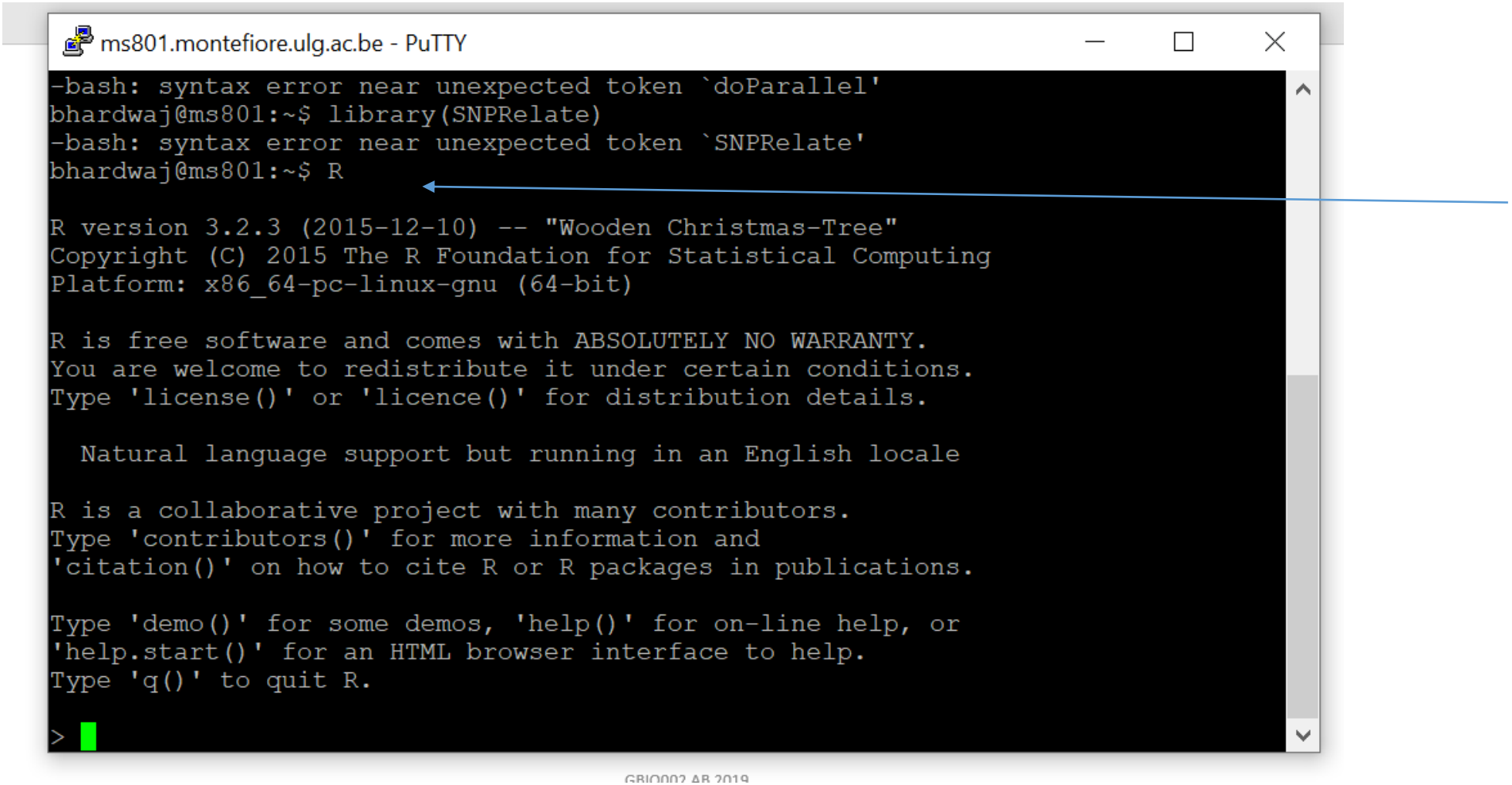
Open terminal

- **Enter command**

  ssh username@ms801.montefiore.ulg.ac.be

You will be connected to the server

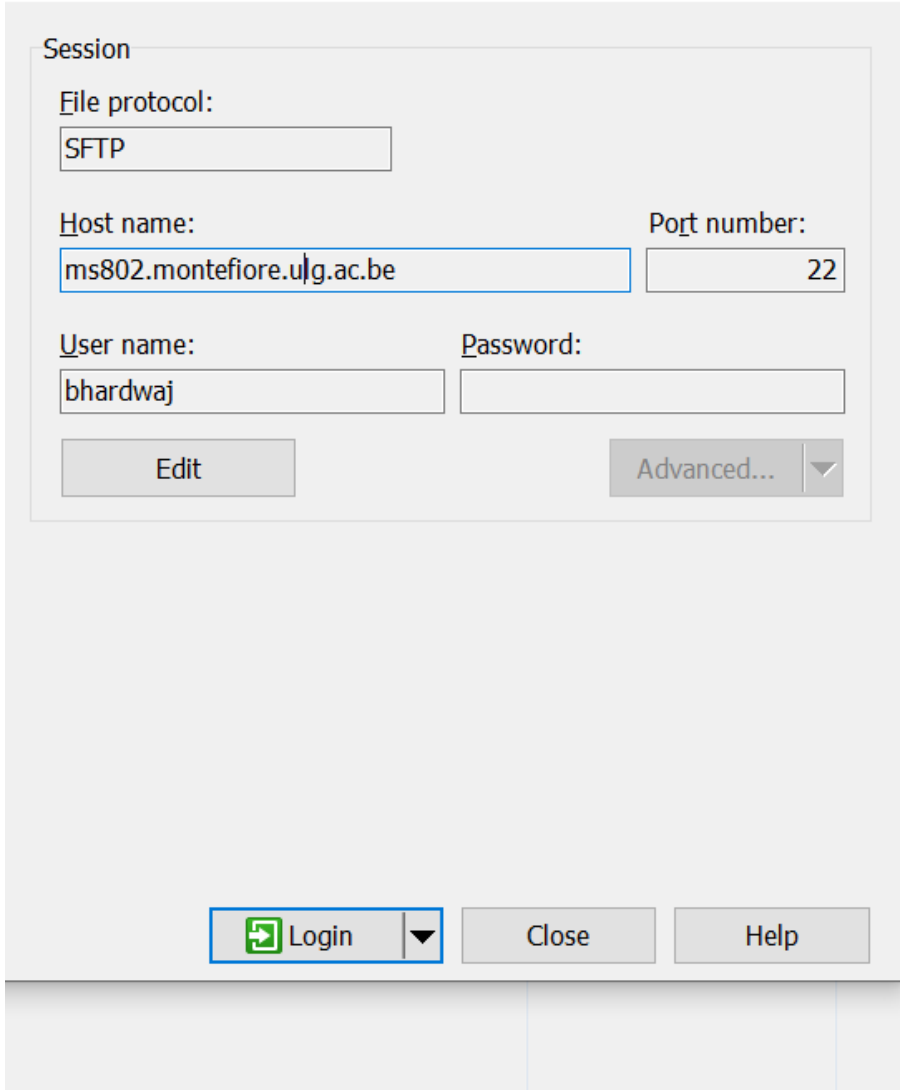
GO TO « R »


ms801.montefiore.ulg.ac.be - PuTTY
-bash: syntax error near unexpected token `doParallel'
bhardwaj@ms801:~$ library(SNPRelate)
-bash: syntax error near unexpected token `SNPRelate'
bhardwaj@ms801:~$ R
R version 3.2.3 (2015-12-10) -- "Wooden Christmas-Tree"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (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.
>

TRANSFER FILES TO SERVER

https://winscp.net/eng/download.php

Folder : CMI



READY TO WORK !!

REQUIRED PACKAGES

- *library(GenABEL)*
- *library(dplyr)*
- *library(snpStats)*
- *library(doParallel)*
- *library(SNPRelate)*

Already installed on server .
No need to install it.

If not installed, use following command

```
if (!requireNamespace("BiocManager", quietly = TRUE))

install.packages("BiocManager")

 BiocManager::install("snpStats")
BiocManager::install("dplyr")
BiocManager::install("doParallel")
BiocManager::install("SNPRelate")
```

STUDY DESIGN

- **Data Download : Directory under name of CMI**
- **Background : It has data from three population.**
- **TASK : What are significant SNPs in context to specific trait.**

Files names :

gwas_indian.fam
gwas_indian.bim
gwas_indian.bed

Population A

gwas_chinese.fam
gwas_chinese.bim
gwas_chinese.bed

Population B

gwas_malay.fam
gwas_malay.bim
gwas_malay.bed

Population C

GWASfunction.R
conversionTable.RData
122Chinese_282lipids.txt
120Indian_282lipids.txt
117Malay_282lipids.txt

Lipids « omics » data

Read all files in R (Population data and lipids information)

Lets look into Data

- **PLINK-converted .bed, .fam and .bim Illumina files from each of the three ethnic groups.**

Exercise

- Read data in PLINK
- How many SNPs present in each File ?
- How many indivduals belong to three datasets? Give count of males and females?

From PLINK to R package

> library(snpStats)

- **We will use the function read.plink from the package snpStats and work on the resulting objects throughout the rest of the session.**

```
pathC <- paste("/home/bhardwaj/22-OCT/gwas_chinese", c(".bed", ".bim", ".fam"), sep = "")
SNP_C <- read.plink(pathM[1], pathM[2], pathM[3])

pathI <- paste("/home/bhardwaj/22-OCT/gwas_indian", c(".bed", ".bim", ".fam"), sep = "")
SNP_I <- read.plink(pathI[1], pathI[2], pathI[3])

pathM <- paste("/home/bhardwaj/22-OCT/gwas_malay", c(".bed", ".bim", ".fam"), sep = "")
SNP_M <- read.plink(pathC[1], pathC[2], pathC[3])
```

- **Check no of input columns**

```
if( ncol(SNP_C$genotypes) != ncol(SNP_I$genotypes)) {
   stop("Different number of columns in input files detected. This is not allowed.")
}
if( ncol(SNP_I$genotypes) != ncol(SNP_M$genotypes)) {
   stop("Different number of columns in input files detected. This is not allowed.")
}
```

- **Append the $genotypes element from each object row-wise (we've checked columns are equal in number)**

```
SNP <- rbind(SNP_M$genotypes, SNP_I$genotypes, SNP_C$genotypes)
```

Data merge (1/2)

- **Merge the three SNP datasets**

SNP <- rbind(SNP_M$genotypes, SNP_I$genotypes, SNP_C$genotypes)

Genotype will be merged

- **Lets merge all 3 maps are based on the same ordered set of SNPs)**

map <- SNP_M$map

```
> head(map)
           chromosome    snp.name cM position allele.1 allele.2
rs4477212           1   rs4477212 NA    82154     <NA>        A
kgp15717912         1 kgp15717912 NA   534247     <NA>        C
kgp7727307          1  kgp7727307 NA   569624     <NA>        C
kgp15297216         1 kgp15297216 NA   723918     <NA>        G
rs3094315           1   rs3094315 NA   752566        G        A
rs3131972           1   rs3131972 NA   752721        A        G
>
```

Data merge (2/2)

- **Get colnames from map data variable**

  colnames(map) <- c("chr", "SNP", "gen.dist", "position", "A1", "A2")

- **Lets merge all 3 fam are based on the same ordered set of SNPs)**

  SNP_M$fam<- rbind(SNP_M$fam, SNP_Ifam, SNP_Cfam)

```
>  head(SNP_M$fam)
          pedigree    member father mother sex affected
M11060903 M11060903 M11060903     NA     NA   1       NA
M11072004 M11072004 M11072004     NA     NA   2       NA
M11071302 M11071302 M11071302     NA     NA   1       NA
M11072317 M11072317 M11072317     NA     NA   1       NA
M11071307 M11071307 M11071307     NA     NA   2       NA
M11073011 M11073011 M11073011     NA     NA   2       NA
>
```

Lets Read another Dataset :
Lipids information of three population

- **Next we import and merge the three lipid data sets (stored as .txt)**

lipidsMalay <- read.delim("/home/bhardwaj/22-OCT/117Malay_282lipids.txt", row.names = 1)

lipidsIndian <- read.delim("/home/bhardwaj/22-OCT/120Indian_282lipids.txt", row.names = 1)

lipidsChinese <- read.delim("/home/bhardwaj/22-OCT/122Chinese_282lipids.txt", row.names = 1)

all(Reduce(intersect, list(colnames(lipidsMalay), colnames(lipidsIndian), colnames(lipidsChinese))) == colnames(lipidsMalay))

> dim(lipidsChinese)
[1] 122 283

- **merge the three lipid data sets (stored as .txt)**

lip <- rbind(lipidsMalay, lipidsIndian, lipidsChinese)

Country Information of samples

```
country <- sapply(list(SNP_M, SNP_I, SNP_C), function(k){
    nrow(k$genotypes)
})
```

```
> country
[1] 110 105 108
```

```
origin <- data.frame(sample.id = rownames(SNP),
                     Country = factor(rep(c("M", "I", "C"), country)))
```

sample.id	Country
1 M11060903	M
2 M11072004	M
3 M11071302	M
4 M11072317	M
5 M11071307	M
6 M11073011	M

Matched data in GWAS and Lipids: Country Information of samples

```
matchingSamples <- intersect(rownames(lip), rownames(SNP))
SNP <- SNP[matchingSamples,]
lip <- lip[matchingSamples,]
origin <- origin[match(matchingSamples, origin$sample.id),]
```

Save session :

```
genData <- list(SNP = SNP, MAP = map, LIP = lip)
save(genData, origin, file = "PhenoGenoMap.RData")
```

Get GDS structure

- **Merge the three SNP datasets**

```
SNP_M$genotypes <- rbind(SNP_M$genotypes, SNP_I$genotypes, SNP_C$genotypes)
colnames(map) <- c("chr", "SNP", "gen.dist", "position", "A1", "A2")
SNP_M$fam<- rbind(SNP_M$fam, SNP_I$fam, SNP_C$fam)
```

- **Rename SNPs present in the conversion table into rs IDs**

```
load("conversionTable.RData")
mappedSNPs <- intersect(SNP_M$map$SNP, names(conversionTable))
newIDs <- conversionTable[match(SNP_M$map$SNP[SNP_M$map$SNP %in% mappedSNPs],
names(conversionTable))]
SNP_M$map$SNP[rownames(SNP_M$map) %in% mappedSNPs] <- newIDs
```

write.plink("convertGDS", snps = SNP_M$genotypes)

```
Writing FAM file to convertGDS.fam
Writing extended MAP file to convertGDS.bim
Writing BED file to convertGDS.bed (SNP-major mode)
NULL
```

What are quality check we can do

- **Based on MAF ? (As performed in plink)**
- **Based on HWE ? (As performed in plink)**

Lets do it one by one and analyse the data.

Pre-Processing or QC

- In a nutshell, the pre-processing of the data consists in discarding SNPs with call rate < 1 or MAF < 0.1 discarding samples with call rate < 100%, IBD kinship coefficient > 0.1 or inbreeding coefficient $|F| > 0.1$

- Call rate is the proportion of SNPs (or samples) that were genotyped. For example, a call rate of 0.95 for a particular SNP (sample) means 5% of the values are missing. (as performed in PLINK)

- Minor-allele frequency (MAF) denotes the proportion of the least common allele for each SNP.

- Of course, it is harder to detect associations with rare variants and this is why we select against low MAF values. Most GWA studies typically report MAF thresholds of 0.05. (as performed in PLINK)

Load Packages

```
library(snpStats)
library(doParallel)
library(SNPRelate)
library(GenABEL)
library(dplyr)
source("GWASfunction.R")
load("PhenoGenoMap.RData")
```

Minor Allele Frequency

- **Lets use SNP call rate of 100%, MAF of 0.1 (very stringent)** **(as perfomed in PLINK)**

```
maf <- 0.1    # selected parameter

callRate <- 1 # selected parameter

SNPstats <- col.summary(genData$SNP)
```

SNPstats Detail Information

	Calls	Call.rate	Certain.calls	RAF	MAF	P.AA	P.AB	P.BB	z.HWE
rs3094315	319	1	1	0.860501567398119	0.139498432601881	0.0188087774294671	0.241379310344828	0.739811912225705	0.0968676724046835
rs3131972	319	1	1	0.766457680250784	0.233542319749216	0.0313479623824451	0.404388714733542	0.564263322884012	2.31429230658791
kgp15275285	319	1	1	0.855799373040752	0.144200626959248	0.0438871473354232	0.200626959247649	0.755485893416928	-3.34227919125702
kgp5225889	319	1	1	0.877742946708464	0.122257053291536	0.0188087774294671	0.206896551724138	0.774294670846395	-0.642784289569623
rs11240777	319	1	1	0.69435736677116	0.30564263322884	0.0689655172413793	0.473354231974922	0.457680250783699	2.05783717257905
kgp8975187	319	1	1	0.713166144200627	0.286833855799373	0.0626959247648903	0.448275862068966	0.489028213166144	1.70937237845253
kgp5074587	319	1	1	0.811912225705329	0.188087774294671	0.0376175548589342	0.300940438871473	0.661442006269592	-0.262046989104515

Lets Filter based on MAF

maf_call <- with(SNPstats, MAF > maf & Call.rate == callRate)

TRUE : Passed MAF
FALSE : Failed MAF

```
> maf_call
  [1] FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
 [13] FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 [25] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
 [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE
 [61] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE
 [73] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE
 [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [97] FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[109] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[133] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[157] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[169] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[193] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[205]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[229] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[241] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE
[253]  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE
[265] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[277] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE
[289] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[301] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[313] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[325] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[337] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[349] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[361] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[373] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[385] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[397] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE
[409] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE
[421]  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[433] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE
[445] FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE
[457] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE
```

genData$SNP <- genData$SNP[,maf_call]
genData$MAP <- genData$MAP[maf_call,]
SNPstats <- SNPstats[maf_call,] # ***Read output***
head(SNPstats)

Hardy-Weinberg equilibrium (HWE 1/2)

- **Next, we need to consider samples that exhibit excessive heterozygosity – technically speaking, deviations from the Hardy-Weinberg equilibrium (HWE),**

- **Overall, large and small |F| might indicate poor sample quality or inbreeding, respectively. We will exclude samples with |F|> 0.1.**

- **Lets run Sample call rate & heterozygosity**

callMat <- !is.na(genData$SNP)

Sampstats <- row.summary(genData$SNP)

```
> head(Sampstats)
          Call.rate Certain.calls Heterozygosity       hetF
M11051101         1             1      0.3308424  0.09523034
M11050404         1             1      0.3165761  0.13424505
M11051105         1             1      0.3750000 -0.02552947
M11051901         1             1      0.4028533 -0.10170104
M11052507         1             1      0.3519022  0.03763720
M11050514         1             1      0.3491848  0.04506858
>
```

Hardy-Weinberg equilibrium (HWE 2/2)

hetExp <- callMat %% (2 * SNPstats$MAF * (1 - SNPstats$MAF))* ***# Hardy-Weinberg heterozygosity (expected)***

*hetObs <- with(Sampstats, Heterozygosity * (ncol(genData$SNP)) * Call.rate)* ***# Hardy-Weinberg heterozygosity (observed value)***

Sampstats$hetF <- 1-(hetObs/hetExp) ***(# hetF prediction)***

Use sample call rate of 100%, het threshold of 0.1 (very stringent)

- *het <- 0.1 # Set cutoff for inbreeding coefficient;*
- *het_call <- with(Sampstats, abs(hetF) < het & Call.rate == 1)*

```
> head(het_call)
          [,1]
M11051101  TRUE
M11050404 FALSE
M11051105  TRUE
M11051901 FALSE
M11052507  TRUE
M11050514  TRUE
```

- *genData$SNP <- genData$SNP[het_call,]*
- *genData$LIP <- genData$LIP[het_call,]*

Identity by Descent IBD

- **Finally, we will investigate relatedness among samples using the kinship coefficient based on identity by descent (IBD).**

- **These functions from the package SNPRelate require GDS files. For this reason we first need to aggregate the .bed, .fam and .bim files from the three populations into convertGDS.**

- **The function snpgdsBED2GDS2 creates the GDS necessary for this part of the analysis.**

- **To determine the kinship coefficient between pairs of samples we will use a subset of uncorrelated SNPs in order to have unbiased estimates.**

- **For this purpose, we will use linkage disequilibrium (LD) as a measure of correlation between markers. LD ranges from 0 to 1, the higher its value the more likely two SNPs co-segregate and therefore correlate.**

- **Here, we will utilize the subset of SNPs with LD < 0.2 to determine the IBD kinship coefficient.**

```
ld <- .2
kin <- .1
snpgdsBED2GDS(bed.fn = "convertGDS.bed", bim.fn = "convertGDS.bim",
  fam.fn = "convertGDS.fam", out.gdsfn = "myGDS", cvt.chr = "char")
genofile <- snpgdsOpen("myGDS", readonly = F)
gds.ids <- read.gdsn(index.gdsn(genofile, "sample.id"))
gds.ids <- sub("-1", "", gds.ids)
add.gdsn(genofile, "sample.id", gds.ids, replace = T)
geno.sample.ids <- rownames(genData$SNP)
```

Lets filter for LD (1/3)

- **Lets filter for LD**

snpSUB <- snpgdsLDpruning(genofile, ld.threshold = ld,
sample.id = geno.sample.ids,
snp.id = colnames(genData$SNP))

```
> snpSUB <- snpgdsLDpruning(genofile, ld.threshold = ld,
+                           sample.id = geno.sample.ids,
+                           snp.id = colnames(genData$SNP))
SNP pruning based on LD:
Excluding 3528 SNPs on non-autosomes
Excluding 0 SNP (monomorphic: TRUE, < MAF: NaN, or > missing rate: NaN)
Working space: 221 samples, 1472 SNPs
      Using 1 (CPU) core
      Sliding window: 500000 basepairs, Inf SNPs
      |LD| threshold: 0.2
Chromosome 0: 3.34%, 167/5000
167 SNPs are selected in total.
```

(as perfomed in PLINK)

Lets filter for LD (2/3)

```
snpset.ibd <- unlist(snpSUB, use.names = F)
# And now filter for MoM
ibd <- snpgdsIBDMoM(genofile, kinship = T,
sample.id = geno.sample.ids,
snp.id = snpset.ibd,
num.thread = 1)
```

```
> ibd <- snpgdsIBDMoM(genofile, kinship = T,
+                     sample.id = geno.sample.ids,
+                     snp.id = snpset.ibd,
+                     num.thread = 1)
IBD analysis (PLINK method of moment) on SNP genotypes:
Excluding 4833 SNPs on non-autosomes
Excluding 0 SNP (monomorphic: TRUE, < MAF: NaN, or > missing rate: NaN)
Working space: 221 samples, 167 SNPs
      Using 1 (CPU) core
PLINK IBD:      the sum of all working genotypes (0, 1 and 2) = 20713
PLINK IBD:      Mon Oct 21 16:40:51 2019       0%
PLINK IBD:      Mon Oct 21 16:40:51 2019       100%
```

Lets filter for LD (3/3)

ibdcoef <- snpgdsIBDSelection(ibd)
ibdcoef <- ibdcoef[ibdcoef$kinship >= kin,]

- **Lets filter samples**

```
related.samples <- NULL
while (nrow(ibdcoef) > 0) {
    # count the number of occurrences of each and take the top one
    sample.counts <- sort(table(c(ibdcoef$ID1, ibdcoef$ID2)), decreasing = T)
    rm.sample <- names(sample.counts)[1]
    cat("Removing sample", rm.sample, "too closely related to",
       sample.counts[1], "other samples.\n")

    # remove from ibdcoef and add to list
    ibdcoef <- ibdcoef[ibdcoef$ID1 != rm.sample & ibdcoef$ID2 != rm.sample,]
    related.samples <- c(as.character(rm.sample), related.samples)
}
```

After pre-processing, we are left with few samples . Note that your sample size might differ slightly as the LD pruning procedure is stochastic.

Filtered Data

```
genData$SNP <- genData$SNP[!(rownames(genData$SNP) %in% related.samples),]
genData$LIP <- genData$LIP[!(rownames(genData$LIP) %in% related.samples),]
```

Next :

What is the population structure in studied samples of india, chinese and malay population?

Principal Component Analysis

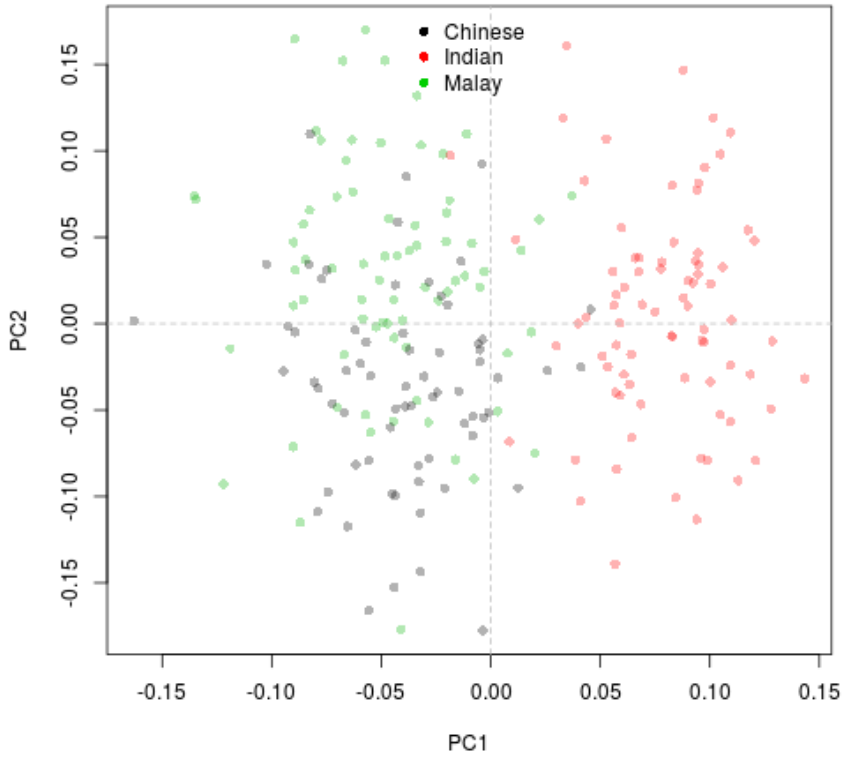
- **Now that we are done with the pre-processing, it might be a good idea to examine the largest sources of variation in the genotype data and look out for outliers or clustering patterns, using Principal Component Analysis (PCA).**

- **Because we are working with S4 objects, we will be using the PCA function from SNPRelate, snpgdsPCA. Let's plot the first two principal components (PCs).**

- **Lets run the PCA**

```
pca <- snpgdsPCA(genofile, sample.id = geno.sample.ids, snp.id = snpset.ibd,
num.thread = 1)

pctab <- data.frame(sample.id = pca$sample.id, PC1 = pca$eigenvect[,1], PC2 =
pca$eigenvect[,2], stringsAsFactors = F)

origin <- origin[match(pca$sample.id, origin$sample.id),]

pcaCol <- rep(rgb(0,0,0,.3), length(pca$sample.id)) # Set black for chinese
pcaCol[origin$Country == "I"] <- rgb(1,0,0,.3) # red for indian
pcaCol[origin$Country == "M"] <- rgb(0,.7,0,.3) # green for malay

png("PCApopulation.png", width = 500, height = 500)
plot(pctab$PC1, pctab$PC2, xlab = "PC1", ylab = "PC2", col = pcaCol, pch = 16)
abline(h = 0, v = 0, lty = 2, col = "grey")
legend("top", legend = c("Chinese", "Indian", "Malay"), col = 1:3, pch = 16, bty =
"n")
dev.off()
```

- **OUTPUT**



- **As expected, the filtered SNP markers clearly delineate the indian population clearly different from other two populations.**

- **The results also suggest that Chinese and Malay are closer to each other than to Indian (this observation would be much better addressed with e.g. hierarchical clustering).**

Genome-Wide Association

- **Note that the glm function is used to determine the significance of association between each SNP and the trait of interest. (as performed in PLINK)**

- **glm is much more versatile than lm since it conducts Gaussian, Poisson, binomial and multinomial regression / classification, depending on how your trait of interest is distributed (all lipids in the phenotype file are Gaussian).**

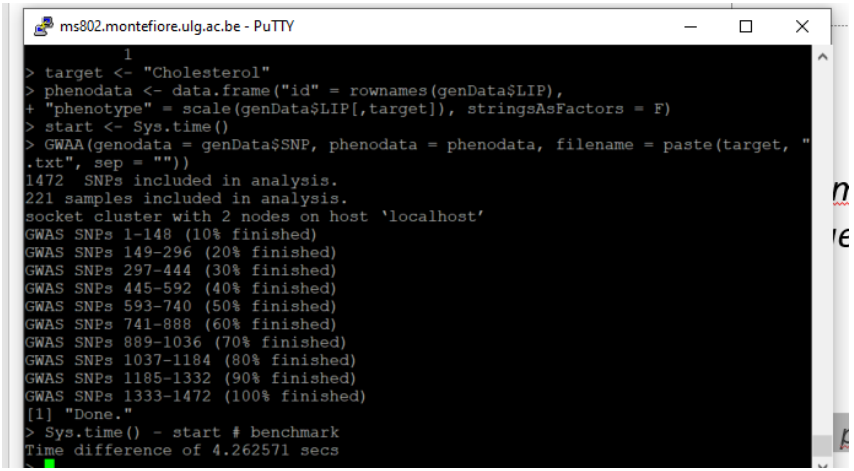
- **This GWA function will not create a variable, but rather write a .txt summary table listing the coefficient estimate \beta , t and the corresponding P-value for each SNP, alongside with the corresponding genomic coordinates.**

- **Choose trait for association analysis,**

```
target <- "Cholesterol"
phenodata <- data.frame("id" = rownames(genData$LIP),
"phenotype" = scale(genData$LIP[,target]), stringsAsFactors = F)
```

- **Conduct GWAS**

```
start <- Sys.time()
GWAA(genodata = genData$SNP, phenodata = phenodata, filename = paste(target,
".txt", sep = ""))
Sys.time() - start # benchmark
```

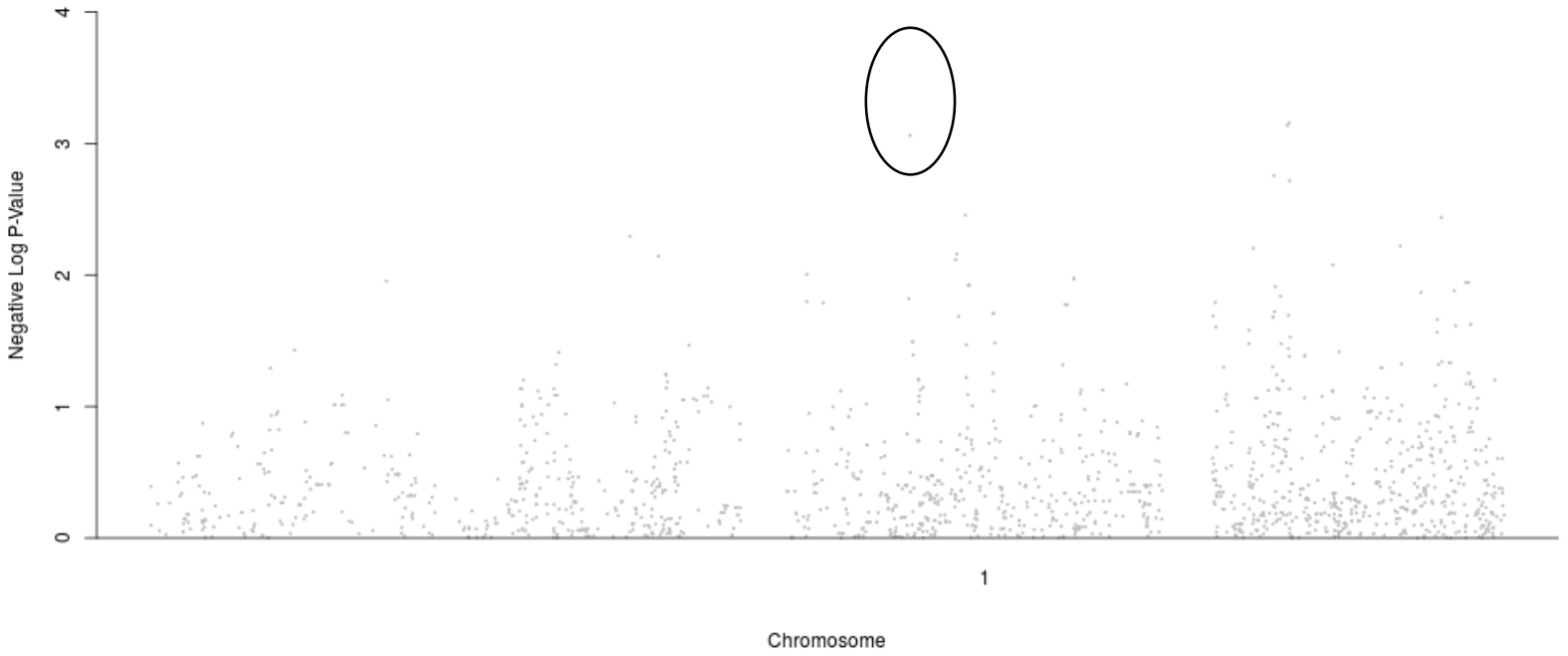


- **Once finished, we can visualize the results using the so-called Manhattan plots.**

- **All we need is to load the .txt summary table written in the previous step, add a column with -\log_{10} (P) and plot these significance estimates against the genomic coordinates of all SNPs.**

- **Lets draw Manhattan plot**

```
GWASout <- read.table(paste(target, ".txt", sep = ""), header = T, colClasses = c("character",
rep("numeric",4)))
GWASout$type <- rep("typed", nrow(GWASout))
GWASout$Neg_logP <- -log10(GWASout$p.value)
GWASout <- merge(GWASout, genData$MAP[,c("SNP", "chr", "position")])
GWASout <- GWASout[order(GWASout$Neg_logP, decreasing = T),]

png(paste(target, ".png", sep = ""), height = 500,width = 1000)
GWAS_Manhattan(GWASout)
dev.off()
```



We see that a total of one SNPs is indicating association (none passes the 'hard' threshold). **rs9509213**

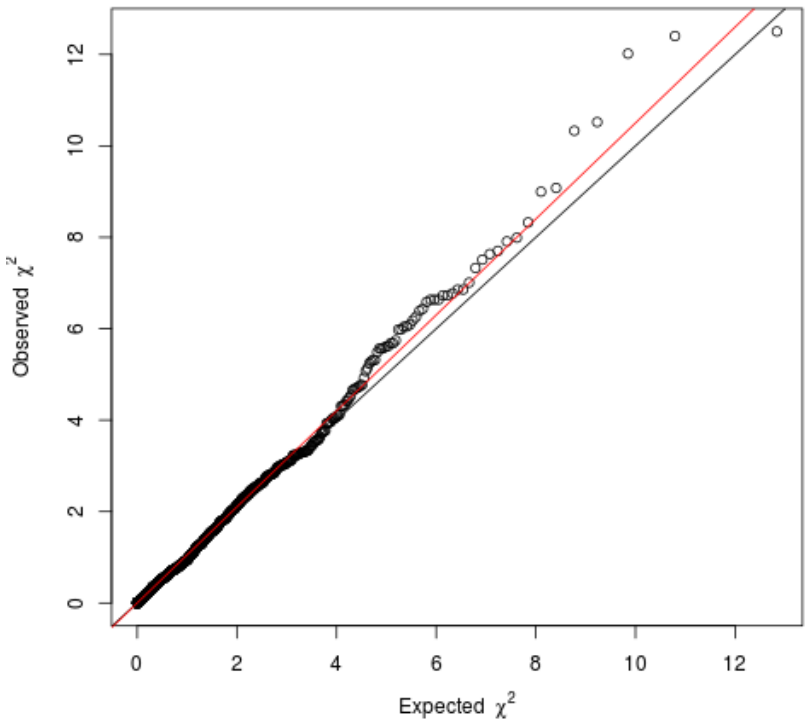
quantile-quantile (Q-Q) plot

- **Before proceeding with this one hits, it is helpful to constrast the distribution of the resulting P-values against that expected by chance, as to ensure there is no confounding systemic bias.**

- ***QQ plot using GenABEL estlambda function***

  - *png(paste(target, "_QQplot.png", sep = ""), width = 500, height = 500)*
  - *lambda <- estlambda(GWASout$t.value**2, plot = T, method = "median")*
  - *dev.off()*

- **The resulting Q-Q plot clearly depicts a trend line (\lambda = 0.99 , red) overlapping with x = y (black) and a slight deviation in the right tail, so we can be confident about our results.**



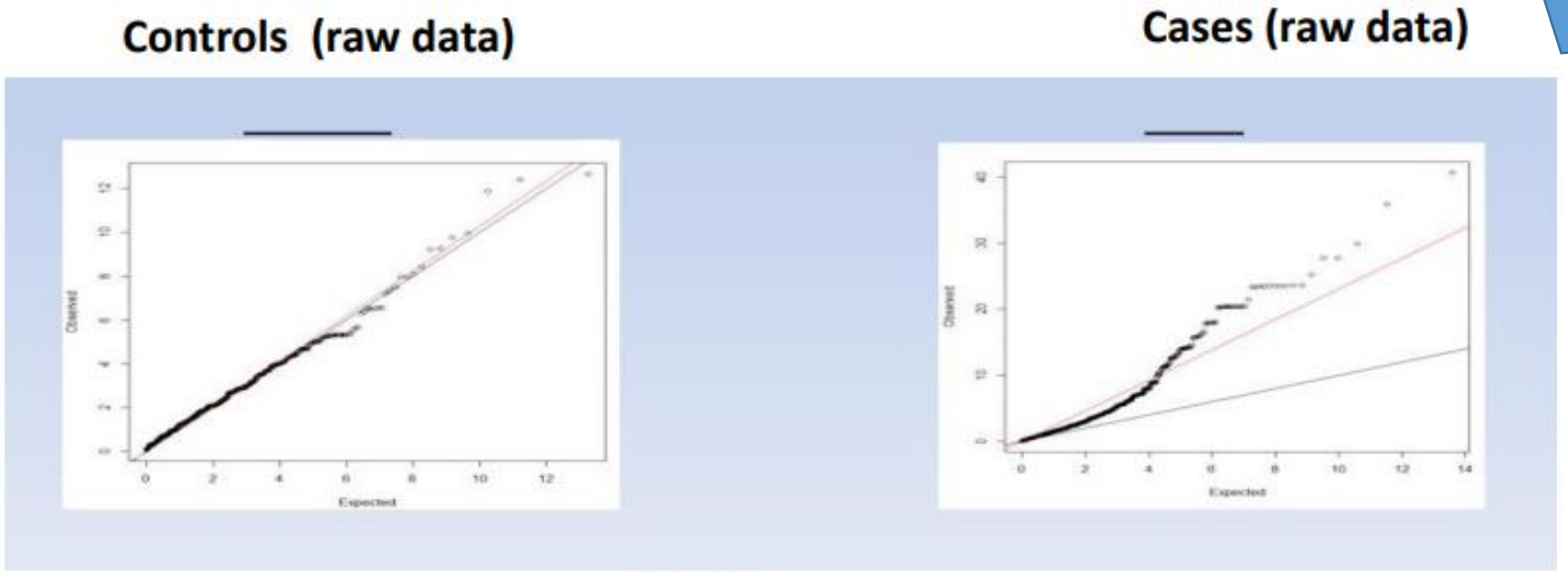
Lambda should range from 0 to 1 (ideal condition).

The resulting Q-Q plot clearly depicts a trend line (\lambda = 1 , red) ,overlapping with x = y (black) and a slight deviation in the right tail.

so we can be more confident about our results.

What if lambda value < 1 ?

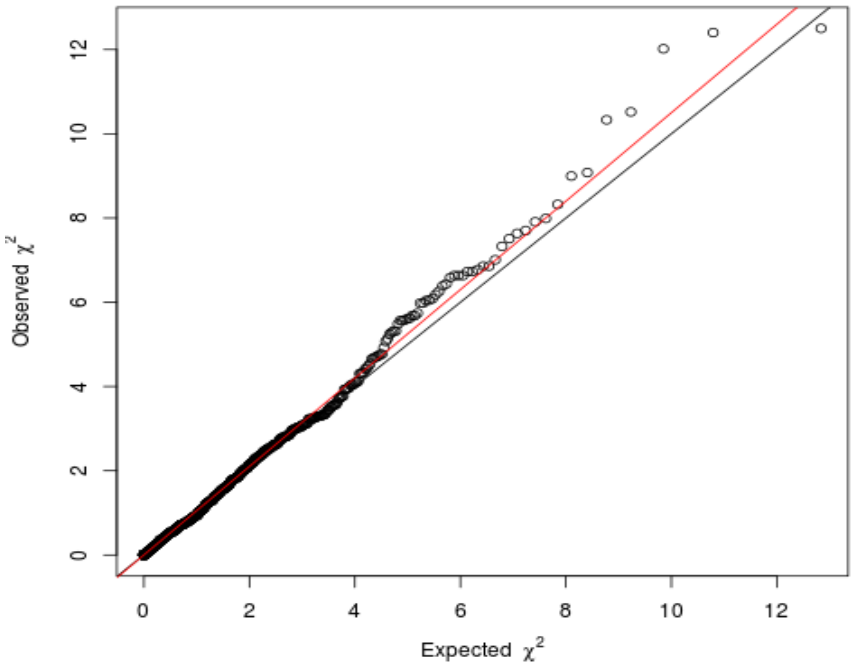
How QQ plot looks like ?



Q-Q plots:

One need to improve the data with more filteration steps

1. Considering the stringency in parameters
2. Looking for the clusters
3. Removal of missing data



(lambda = 1)

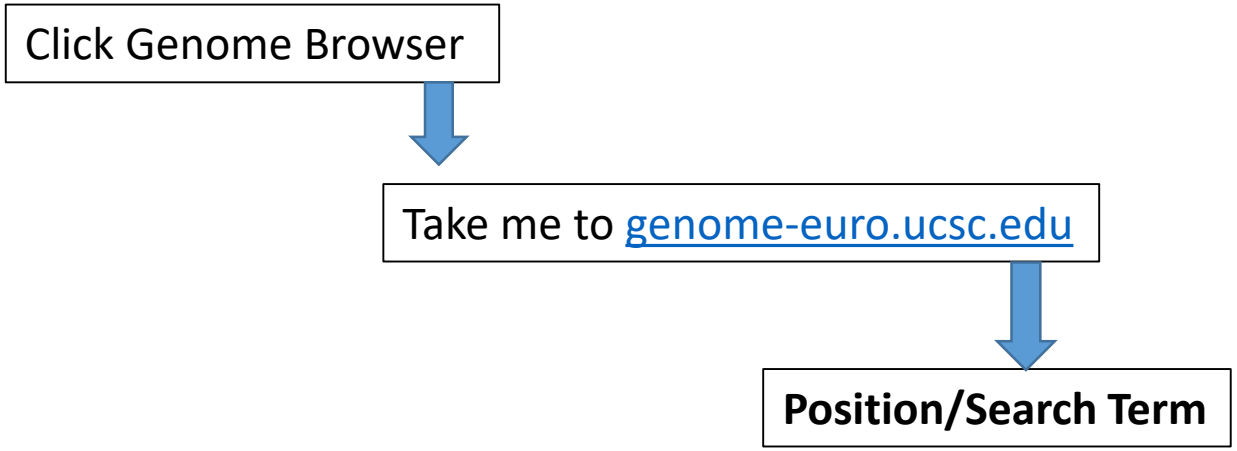
Reason for this lesser lamda :

In this study, we worked on the small sample size. If we include entire dataset, lamda could be increased

Functional insights into candidate markers

- **Next will try to find the functional relevance of these one candidate SNPs by searching for genes in their vicinity, using the USCS Genome Browser (enter Genome Browser, insert the SNP ID in the text box, enter and zoom out).**

- **I found that rs9509213 lands right on *CRYL1* (crystalline lambda 1, intron sequence), rendering it an interesting candidate for follow-up studies.**



USCS Genome Browser

UNIVERSITY OF CALIFORNIA SANTA CRUZ Genomics Institute | UCSC **Genome Browser**

Genomes | Genome Browser | Tools | Mirrors | Downloads | My Data | Projects | Help | About Us

Our tools

- **Genome Browser**
  interactively visualize genomic data
- **BLAT**
  rapidly align sequences to the genome
- **Table Browser**
  download data from the Genome Browser database
- **Variant Annotation Integrator**
  get functional effect predictions for variant calls
- **Data Integrator**
  combine data sources from the Genome Browser database
- **Gene Sorter**
  find genes that are similar by expression and other metrics
- **Genome Browser in a Box (GBiB)**
  run the Genome Browser on your laptop or server
- **In-Silico PCR**
  rapidly align PCR primer pairs to the genome
- **LiftOver**
  convert genome coordinates between assemblies
- **Track Hubs**
  import and view external data tracks

More tools...

Our story

On June 22, 2000, UCSC and the other members of the International Human Genome Project consortium completed the first working draft of the human genome assembly, forever ensuring free public access to the genome and the information it contains. A few weeks later, on July 7, 2000, the newly assembled genome was released on the web at http://genome.ucsc.edu, along with the initial prototype of a graphical viewing tool, the UCSC Genome Browser. In the ensuing years, the website has grown to include a broad collection of vertebrate and model organism assemblies and annotations, along with a large suite of tools for viewing, analyzing and downloading data.

What's new

Oct. 16, 2019 - **New GENCODE Genes tracks for human (V32) ...**

Oct. 11, 2019 - **New "group auto-scale" option**

Oct. 9, 2019 - **Expanded CRISPR track released for human (hg...**

More news...

Subscribe

*The UCSC Genome Browser is developed and maintained by the **Genome Bioinformatics Group**, a cross-departmental team within the **UCSC Genomics Institute**.*

UNIVERSITY OF CALIFORNIA SANTA CRUZ Genomics Institute | UCSC

Genome Browser Gateway

Genomes | Genome Browser | Tools | Mirrors | Downloads | My Data | Projects | Help | About Us

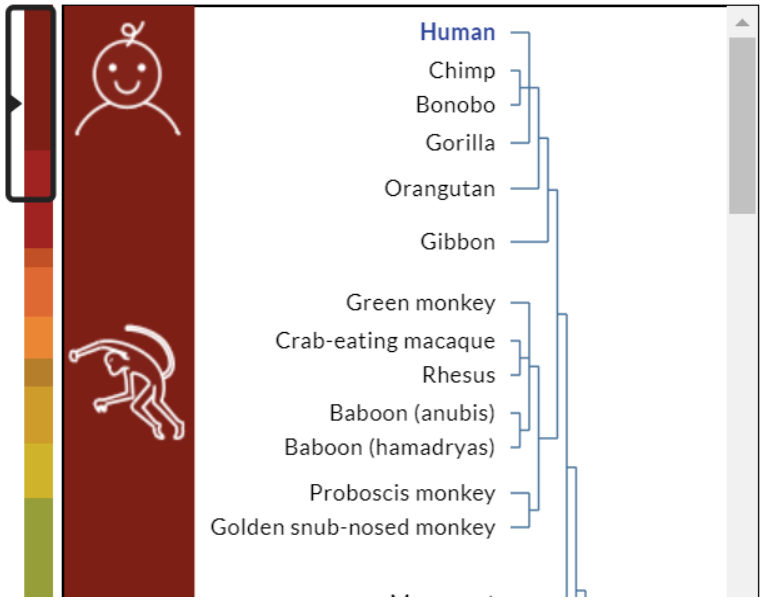
Browse/Select Species

POPULAR SPECIES

Human | Mouse | Rat | Zebrafish | Fruitfly | Worm | Yeast

Enter species or common name

REPRESENTED SPECIES



Find Position

Human Assembly

Dec. 2013 (GRCh38/hg38)

GO

Position/Search Term

Enter position, gene symbol or search terms

Current position: chr13:20,445,716-20,446,216

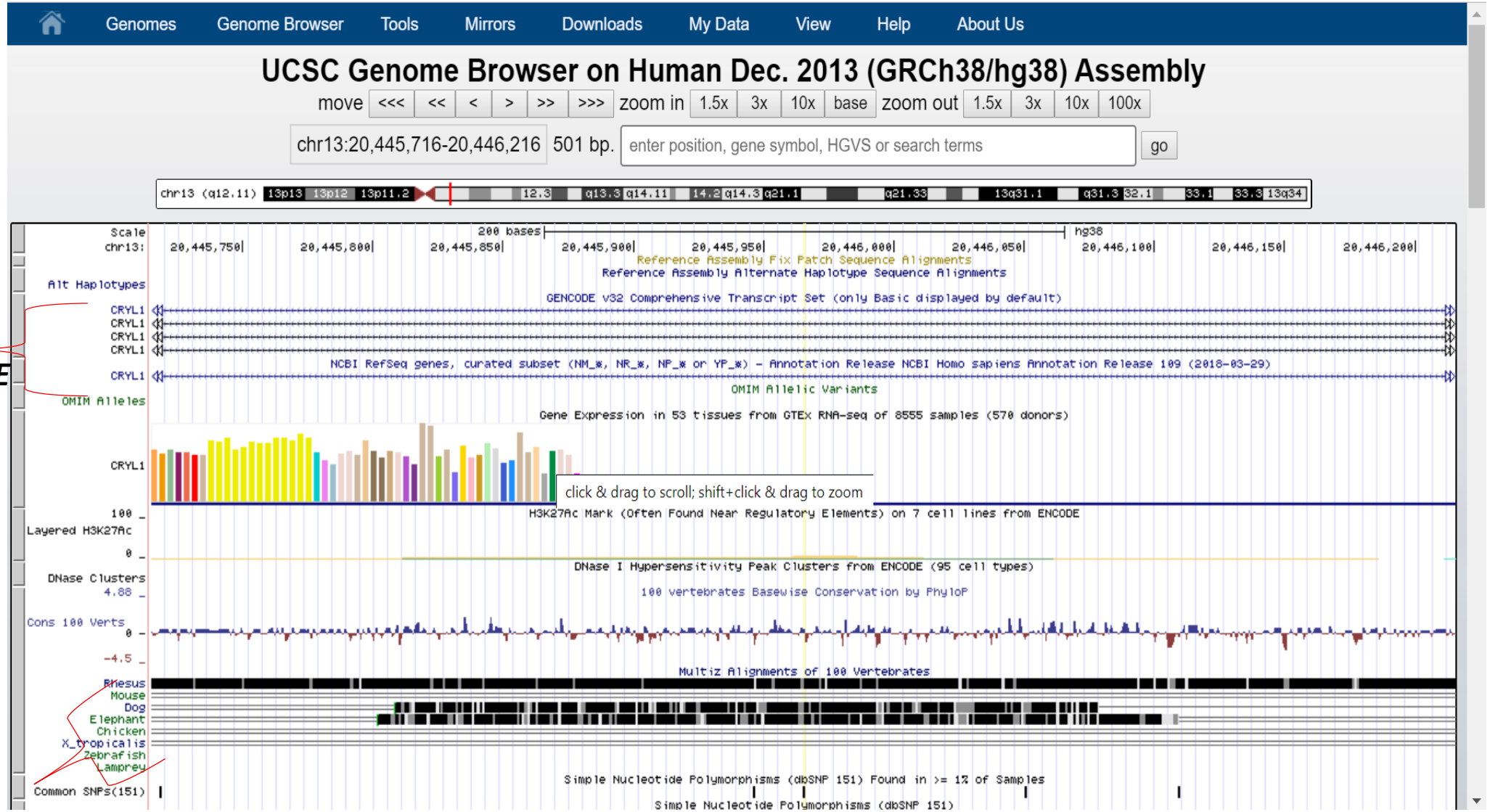
Human Genome Browser - hg38 assembly — view sequences

UCSC Genome Browser assembly ID: hg38
Sequencing/Assembly provider ID: Genome Reference Consortium Human GRCh38.p12 (GCA_000001405.27)
Assembly date: Dec. 2013 initial release; Dec. 2017 patch release 12
Assembly accession: GCA_000001405.27
NCBI Genome ID: 51 (Homo sapiens (human))
NCBI Assembly ID: 5800238 (GRCh38.p12, GCA_000001405.27)
BioProject ID: PRJNA31257

Homo sapiens
(Graphic courtesy of CBSE)

Search the assembly:

- **By position or search term:** Use the "position or search term" box to find areas of the genome associated with many different attributes, such as a specific chromosomal coordinate range; mRNA, EST, or STS marker names;



GENE NAME

Genomic comparision with other species

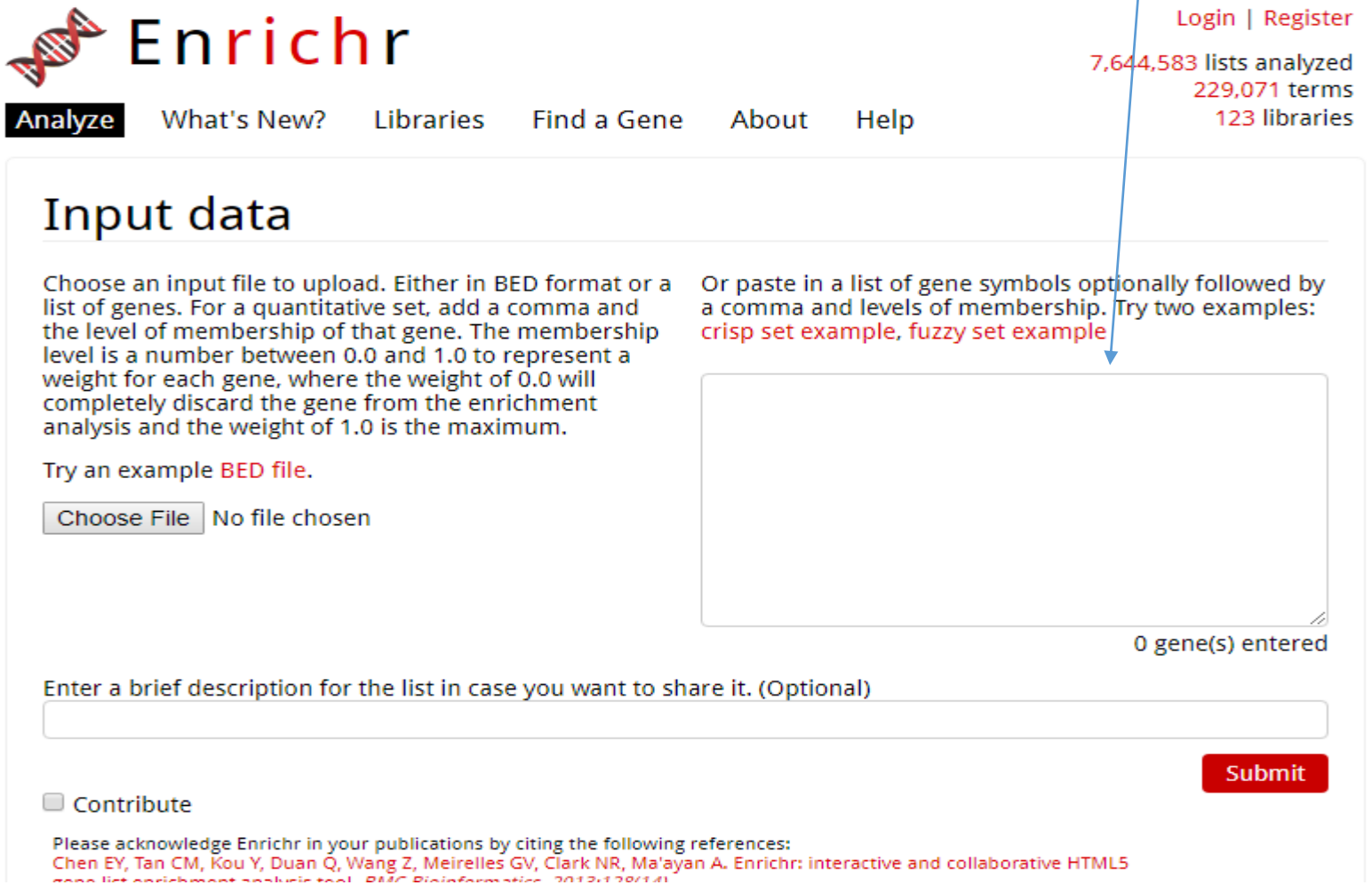
Exercice : Identify gene name of following SNPs

rs11083846
rs11636802
rs13397985
rs13401811
rs1679013
rs17483466
rs210142

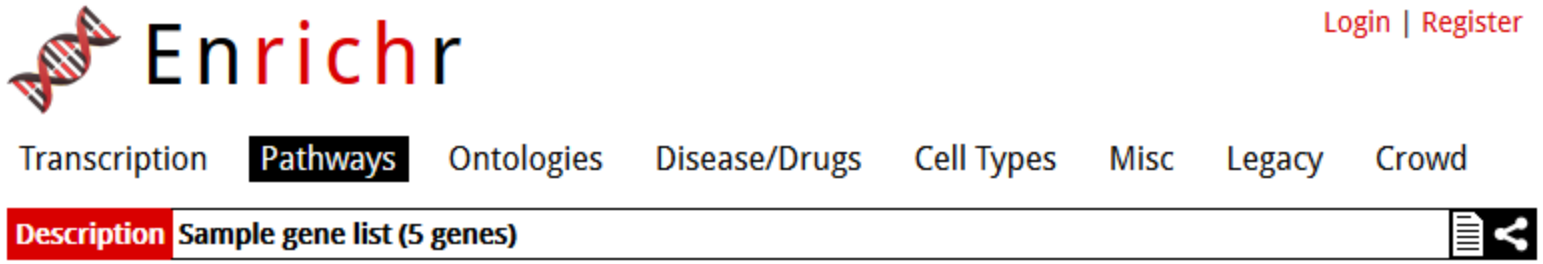
Advance Functional interpretation

Open browser https://amp.pharm.mssm.edu/Enrichr/

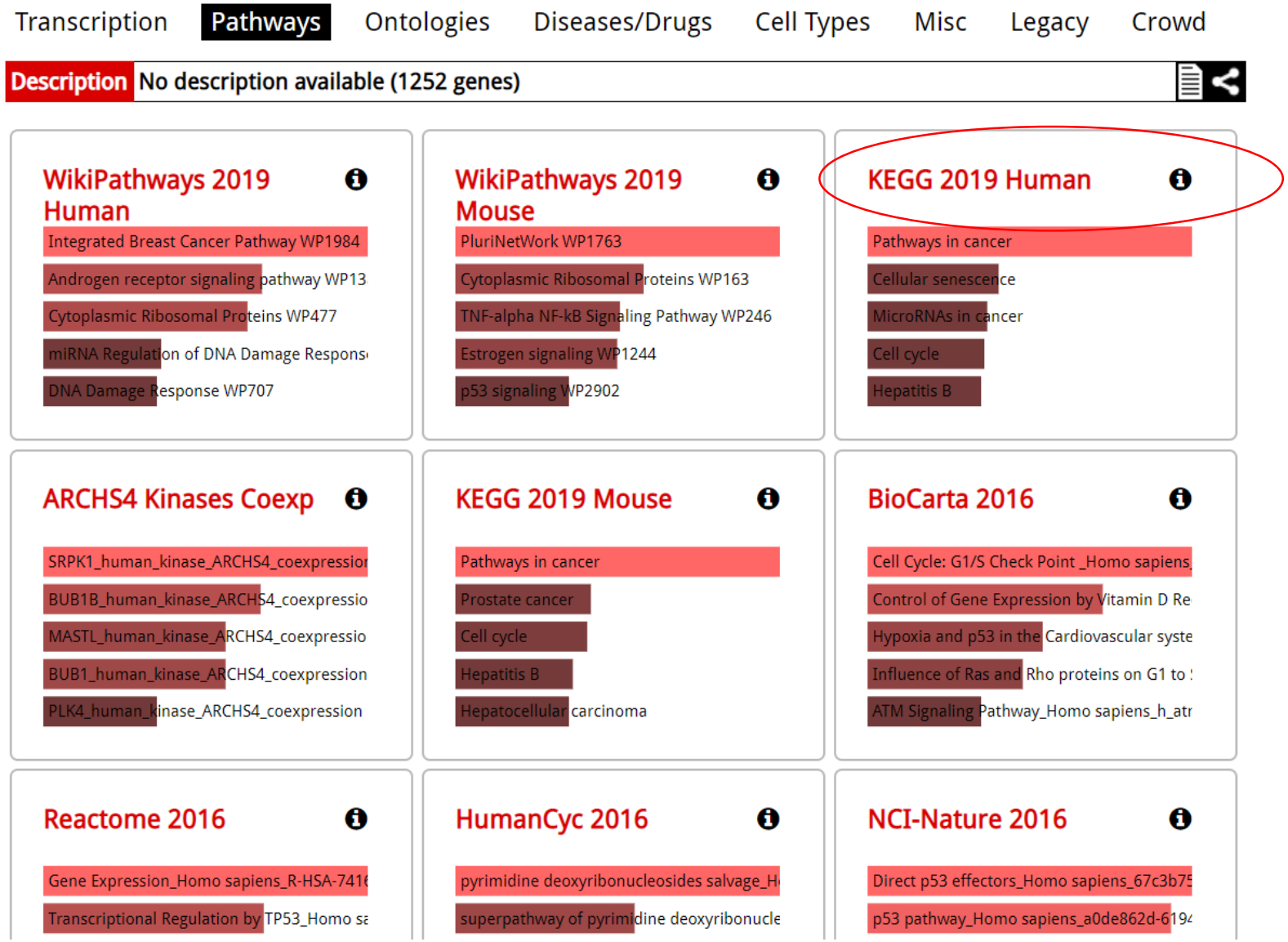
and paste gene list



https://amp.pharm.mssm.edu/Enrichr/



- **Click on pathways**
- **Click on ontologies**
  - **Cellular , Molecular and Biological functions??**

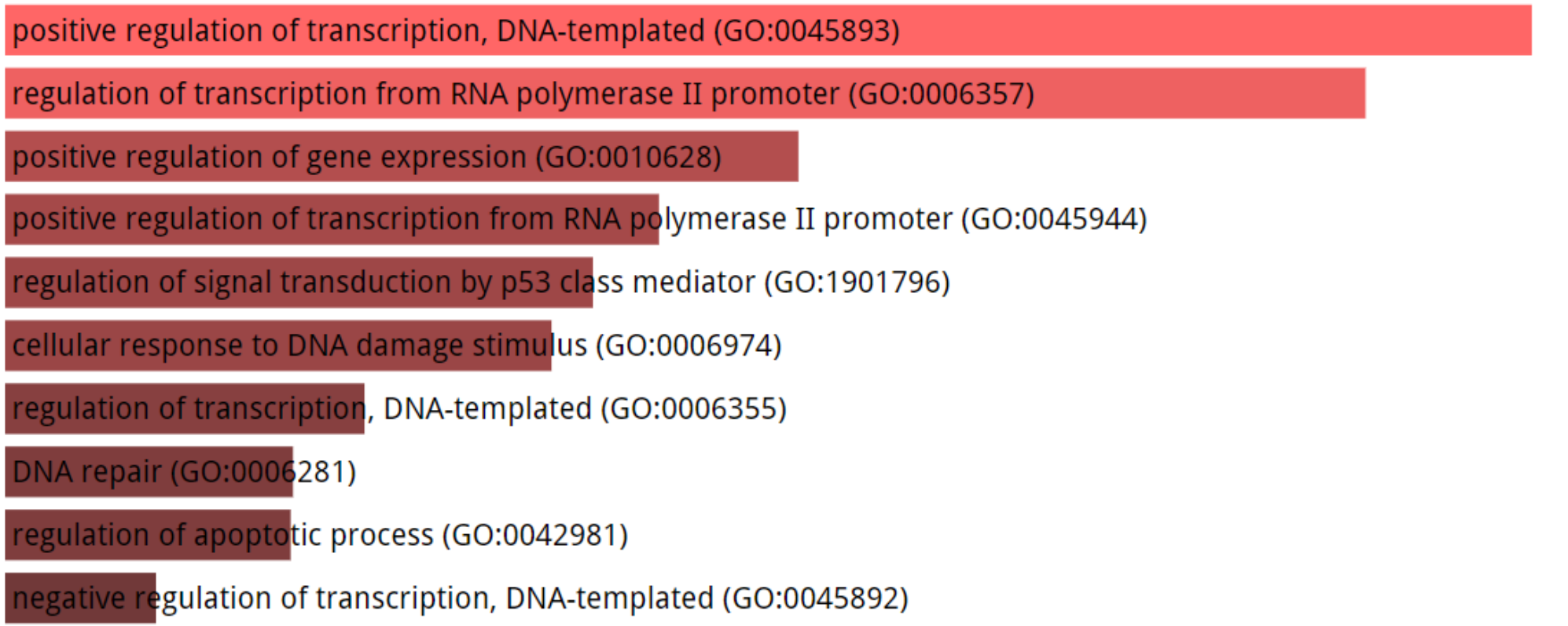

Transcription
Pathways
Ontologies
Diseases/Drugs
Cell Types
Misc
Legacy
Crowd
Description
No description available (1252 genes)
WikiPathways 2019 Human
Integrated Breast Cancer Pathway WP1984
Androgen receptor signaling pathway WP13
Cytoplasmic Ribosomal Proteins WP477
miRNA Regulation of DNA Damage Respons
DNA Damage Response WP707
WikiPathways 2019 Mouse
PluriNetWork WP1763
Cytoplasmic Ribosomal Proteins WP163
TNF-alpha NF-kB Signaling Pathway WP246
Estrogen signaling WP1244
p53 signaling WP2902
KEGG 2019 Human
Pathways in cancer
Cellular senescence
MicroRNAs in cancer
Cell cycle
Hepatitis B
ARCHS4 Kinases Coexp
SRPK1_human_kinase_ARCHS4_coexpressio
BUB1B_human_kinase_ARCHS4_coexpressio
MASTL_human_kinase_ARCHS4_coexpressio
BUB1_human_kinase_ARCHS4_coexpression
PLK4_human_kinase_ARCHS4_coexpression
KEGG 2019 Mouse
Pathways in cancer
Prostate cancer
Cell cycle
Hepatitis B
Hepatocellular carcinoma
BioCarta 2016
Cell Cycle: G1/S Check Point _Homo sapiens_
Control of Gene Expression by Vitamin D Re
Hypoxia and p53 in the Cardiovascular syste
Influence of Ras and Rho proteins on G1 to
ATM Signaling Pathway_Homo sapiens_h_atr
Reactome 2016
Gene Expression_Homo sapiens_R-HSA-741
Transcriptional Regulation by TP53_Homo sa
HumanCyc 2016
pyrimidine deoxyribonucleosides salvage_H
superpathway of pyrimidine deoxyribonucle
NCI-Nature 2016
Direct p53 effectors_Homo sapiens_67c3b75
p53 pathway_Homo sapiens_a0de862d-619

GO Biological Process 2018

Bar Graph | Table | Clustergram

Click the bars to sort. Now sorted by **p-value ranking**.

SVG PNG JPG



Let us try Unknown dataset

1. Download gene _ist .txt from course website.

2. Predict Gene ontology

3. Predict different pathways

4. Any idea, genes playing specific biological functional ??

Additional Reference

https://onlinelibrary.wiley.com/doi/full/10.1002/mpr.1608

FROM TODAY SESSION

- Performed different steps of GWAS and identified signals
- Performed annotation of identified signals (different ways)
  - Genomic information
  - PATHWAYS
  - Gene ontology

NEXT SESSION

POST GWAS : ADVANCE FUNCTIONAL ANNOTATION