

Advanced Gene Mapping Course

November 7-11, 2022
The Rockefeller University
New York, NY

Computer Exercises

Table of Contents

| | |
|---|-----|
| Getting Started..... | I |
| Launching the Exercises..... | II |
| Genome-wide Association Analysis - Quality Control – PLINK*..... | 1 |
| Genome-wide Association Analysis - MDS and PCA – PLINK*..... | 10 |
| Sequence Data QC and Association Analysis – VAT*..... | 15 |
| Association Analysis of Sequence Data using PLINK/SEQ (PSEQ)*..... | 32 |
| REGENIE (Jupyter Notebook in JupyterLab)..... | N/A |
| TWAS (MR-JTI) (Jupyter Notebook in JupyterLab)..... | N/A |
| LD-Clumping (Jupyter Notebook in JupyterLab)..... | N/A |
| Fine-mapping (SuSiE) (Jupyter Notebook in JupyterLab)..... | N/A |
| Mendelian Randomization (MR-base)..... | 39 |
| Pleiotropy..... | 51 |
| FASTLMM..... | 68 |
| GCTA..... | 78 |
| Epistasis (PLINK & CASSI)..... | 97 |
| Power and Sample Size Calculations – Cochran Armitage Test for Trend..... | 104 |
| Polygenic Risk Score (LDpred2) (Jupyter Notebook in JupyterLab)..... | N/A |
| AnnoVar..... | 106 |

*Jupyter Notebook is also available

Getting Started

Please view the following videos to install Docker to your computer

For MAC

<https://www.youtube.com/watch?v=DRCDNB1xZ-w>

For Windows PC

<https://www.youtube.com/watch?v=sxv45NCSFMk>

For Linux

<https://www.youtube.com/watch?v=3K-sGzxyK0>

How to install and run course exercises

<https://www.youtube.com/watch?v=OgHvRVtIIog>

For more detail, please read our course wiki

<https://github.com/statgenetics/statgen-courses/wiki/How-to-launch-course-tutorials#alternative-to-cloud-server-use-your-own-computer>

Please go to <https://statgen.us/Tutorials> and install the following tutorials for the course:

Launching Exercises

Please use the following commands to launch the exercises. Please note that all exercises (except those that are web based) can be performed either in command line console or JupyterLab environment. Exercises developed in Jupyter Notebook (within a JupyterLab environment) that have graphics will not have this output if the command line version is used. Please to not attempt to run Command line and JupyterLab at the same time on your computer, since it will create a file conflict.

- PLINK
 - Command line*: `./statgen-setup login --tutorial plink`
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial plink`
- VAT
 - Command line*: `./statgen-setup login --tutorial vat`
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial vat`
- PSEQ
 - Command line*: `./statgen-setup login --tutorial pseq`
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial pseq`
- REGENIE
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial regenie`
- TWAS (MR-JTI)
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial twas`
- LD-Clumping
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial clumping`
- Fine-mapping (SuSiE)
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial finemap`
- Mendelian Randomization (MR-base)
 - Web-based
- Pleiotropy
 - Command Line*: `./statgen-setup login --tutorial pleiotropy`
- FASTLMM & GCTA
 - Command Line*: `./statgen-setup login --tutorial fastlmm-gcta`
- Epistasis (PLINK and CASSI)
 - Command Line*: `./statgen-setup login --tutorial epistasis`
- Power and Sample Size Estimation (pages)
 - Web-based
- Polygenic Risk Score (LDpred2)
 - Jupyter Notebook in JupyterLab: `./statgen-setup launch --tutorial ldpred2`
- Annovar
 - Jupyter Notebook in JupyterLab: `./statgen-setup login --tutorial annovar`

*Run **get-data** command if you don't see the data for the exercise already loaded (hint use **ls** command). Before you start the exercise, you will need to `cd` into the work directory (`cd ~/work`).

Genome-wide Association Analysis - Data Quality Control

Copyright © 2022 Merry-Lynn McDonald, Isabelle Schrauwen & Suzanne M. Leal

Introduction

In this exercise, you will learn how to perform data quality control (QC) by removing markers and samples that fail QC quality control criteria. You will also examine your samples for individuals that are related to each other and/or are duplicate samples. Each sample will also be tested for excess homozygosity and heterozygosity of genotype data. Each SNP will be tested for deviations from Hardy-Weinberg Equilibrium. These exercises will be carried out using PLINK1.9 and R.

1. Using PLINK

PLINK can upload data in different formats please see the PLINK documentation (<https://www.cog-genomics.org/plink/1.9/input>) for additional details. The data for this exercise is in PLINK/LINKAGE file format. There are two files: a pedfile (GWAS.ped) and a map file (GWAS.map). Please examine these files and the PLINK documentation. Please note the commands must be given in the directory where the data resides.

Navigate via the command prompt to the directory which contains the files for the exercise. Type **plink** in the command prompt and make note of the output. Next type:

```
plink --file GWAS
```

Note, that PLINK outputs a file called **plink.log** that contains the same output which you see on the screen. To see all options, type `plink --help` for more information. Determine how many samples there are in your data set and fill in Oval 1 of the flowchart below.

2. Data Quality Control

a. Removing Samples and SNPs with Missing Genotypes.

You will exclude samples that are missing more than 10% of their genotype calls. These samples are likely to have been generated using low quality DNA and can also have higher than average genotyping error rates.

```
plink --file GWAS --mind 0.10 --recode --out GWAS_clean_mind
```

Examine **GWAS_clean_mind.log** to see how many samples are excluded based on this criterion and fill in Box 1.

Create two versions of your dataset, one with SNPs with a minor allele frequencies (MAFs) $\geq 5\%$ and the other with SNPs with a MAFs $< 5\%$.

You will now remove SNPs with MAFs $\geq 5\%$ that are missing $> 5\%$ of their genotypes and then remove SNPs with MAFs $< 5\%$ that are missing $> 1\%$ of their genotypes. SNPs which are missing genotypes can have higher error rates than those SNP markers without missing data.

```
plink --file GWAS_clean_mind --maf 0.05 --recode --out MAF_greater_5
```

```
plink --file GWAS_clean_mind --exclude MAF_greater_5.map --recode --out MAF_less_5
```

```
plink --file MAF_greater_5 --geno 0.05 --recode --out MAF_greater_5_clean
```

Fill in Box 2a.

```
plink --file MAF_less_5 --geno 0.01 --recode --out MAF_less_5_clean
```

Fill in Box 2b.

Merge the two files.

```
plink --file MAF_greater_5_clean --merge MAF_less_5_clean.ped MAF_less_5_clean.map --
recode --out GWAS_MAF_clean
```

A more stringent criterion for missing data is used, samples missing >3% of their genotypes are removed.

```
plink --file GWAS_MAF_clean --mind 0.03 --recode --out GWAS_clean2
```

Fill in Box 3.

b. Checking Sex

Error of the reported sex of an individual can occur. Information from the SNP genotypes can be used to verify the sex of individuals, by examining homozygosity (F) on the X chromosome for every individual. F is expected to be <0.2 in females and >0.8 in males. To check sex run

```
plink --file GWAS_clean2 --check-sex --out GWAS_sex_checking
```

Use R to examine the GWAS_sex_checking.sexcheck file and determine if there are individuals whose recorded sex is inconsistent with genetic sex.

```
R
sexcheck = read.table("GWAS_sex_checking.sexcheck", header=T)
names(sexcheck)
sex_problem = sexcheck[which(sexcheck$STATUS=="PROBLEM"),]
sex_problem
q()
```

NA20530 and NA20506 were coded as a female (2) and from the genotypes appear to be males (1). In addition, 3 individuals (NA20766, NA20771 and NA20757) do not have enough information to determine if they are males or females and PLINK reports sex = 0 for the genotyped sex. Fill in the table below:

Table 1: Sex check

| FID | IID | PEDSEX | SNPSEX | STATUS | F |
|---------|---------|--------|--------|--------|---|
| NA20506 | NA20506 | | | | |
| NA20530 | NA20530 | | | | |
| NA20766 | NA20766 | | | | |
| NA20771 | NA20771 | | | | |
| NA20757 | NA20757 | | | | |

Reasons for these kinds of discrepancies, include the records are incorrect, incorrect data entry, sample swap, unreported Turner or Klinefelter syndromes. Additionally, if a sufficient number of SNPs have not been genotyped on the X chromosome it can be difficult to accurately predict the sex of an individual. In this dataset, there are only 194 X chromosomal SNPs. If you cannot validate the sex of the individual they should be removed. For this exercise, we are going to assume that when the sex was checked, we found it was incorrectly recorded (i.e. these samples were male). Therefore, this error could simply be corrected.

Question 1: Why do you expect the homozygosity rate to be higher on the X chromosome in males than females?

c. Duplicate Samples

The following PLINK command can be used to check for duplicate samples:

```
plink --file GWAS_clean2 --genome --out duplicates
```

Open the **duplicates.genome** file in R with the following command:

```
dups = read.table("duplicates.genome", header = T)
```

We are interested in the Pi-Hat (the estimated proportion IBD sharing) value. You may notice that there is more than one duplicate (Pi-Hat= ~ 1). Also, examine the output for pairs of individuals with high Pi-Hat values which can indicate they are related. The amount of allele sharing [Z(0), Z(1) and Z(2)] across all SNPs provides information on the type of relative pair.

```
problem_pairs = dups[which(dups$PI_HAT > 0.4),]  
problem_pairs
```

Table 2: Duplicate and Related Individuals

| FID1 | IID1 | FID2 | IID2 | Z(0) | Z(1) | Z(2) | PI_HAT |
|------|------|------|------|------|------|------|--------|
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

FID1- Family ID for 1st individual; IID1 - Individual ID for 1st individual; FID2- Family ID for 2nd individual; IID2 - Individual ID for 2nd individual; Z(0)- P(IBD=0); Z(1)- P(IBD=1); Z(2)- P(IBD=2); PI_HAT-P(IBD=2)+0.5*P(IBD=1) (proportion IBD)

Question 2: How many duplicate pairs do you find (**hint: Pi-Hat = ~ 1**)? Do pairs with a **Pi-Hat = ~ 1** have to be duplicate samples? What is another explanation? What proportion would you expect a parent/ child to share IBD? Can you find any such relationship?

Note: Pi-hat can be inflated and individuals appear to be related to each other if you have samples from different populations. This explains why we observe pairs of individuals with Pi-hat > 0.05 since three distinct populations were analyzed. Additionally, this phenomenon can be observed if a subset(s) of samples have higher genotyping/sequencing error rates, which creates two or more “populations” and the individuals within these “populations” incorrectly appear to be related.

Using this R script please observe how many sample pairs have pi-hat > 0.05 :

```
problem_pairs = dups[which(dups$PI_HAT > 0.05),]  
myvars = c("FID1", "IID1", "FID2", "IID2", "PI_HAT")  
problem_pairs[myvars]
```

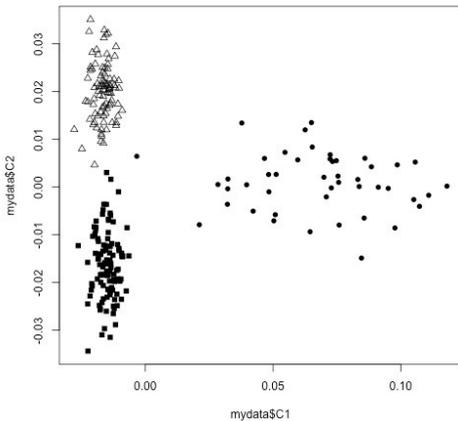
Create the following txt file:

```
1344 NA12057  
1444 NA12739  
M033 NA19774
```

name it ‘IBS_excluded.txt’ and save it to the folder with your PLINK data. Give the command:

```
plink --file GWAS_clean2 --remove IBS_excluded.txt --recode --out GWAS_clean3
```

Fill in Box 4 and Oval 3.



As part of QC usually the data is examined for outliers by plotting the first and second principal or multidimensional scaling (MDS) components. Using a subset of markers that have been trimmed to remove LD ($r^2 < 0.5$). Principal components analysis (PCA) and MDS will be performed in the second part of the exercise to detect outliers and control for populations substructure. Outlier can be due to study subjects coming from different populations e.g. European- and African-Americans or batch effects. If it is suspected that outliers are due to study subjects having been sampled from different populations than data from HapMap can be included to elucidate population membership, e.g. for a study of European-Americans if African-American study subjects are included they would cluster between the European and African HapMap samples. If you perform this type of analysis you should remove the HapMap samples and re-estimate the MDS or PC components before adjusting for population substructure

or stratification. For this exercise data is used from HapMap Phase III which consists of CEU (Europeans from Utah), MEX (Mexicans from Los Angeles) and TSI (Tuscans from Italy). Three clusters can be observed that consist of the three data sets but no extreme outliers are observed. This data set is being used for demonstration purposes. Different populations should be analyzed separately and the results can be combined using meta-analysis. In part two of this exercise MDS and PC components will be constructed and analyzed.

d. Hardy-Weinberg Equilibrium (HWE):

To test for HWE we will test separately in each ancestry group and by case-control status. Therefore, we will need to use information on ancestry and cases-control status. Please note that this should be tested in the 3 different populations separately (CEU, MEX, TSI), but due to the small sample sizes, we tested it in the 3 populations together for example purposes. It should also be noted if the sample sizes are small it is difficult to detect a deviation from HWE.

```
plink --file GWAS_clean3 --pheno pheno.txt --pheno-name Aff --hardy
```

Using R examine the file **plink.hwe** and look for SNPs with p-values of 10^{-7} or smaller.

```
hardy = read.table("plink.hwe", header = T)
names(hardy)
hwe_prob = hardy[which(hardy$P < 0.0000009),]
hwe_prob
```

Using a criterion of $p < 10^{-7}$ to reject the null hypothesis of HWE, how many SNPs fail HWE in the controls? Fill out Oval 5 and Box 4. Using the same criteria, how many SNPs fail HWE in the controls? Complete Table 2 with this information.

Table 3: Hardy-Weinberg Equilibrium

| Cases | | | Controls | | |
|-------|--------|---------------|----------|---------------|--------|
| SNP | Pvalue | Population(s) | SNP | Population(s) | Pvalue |
| | | | | | |

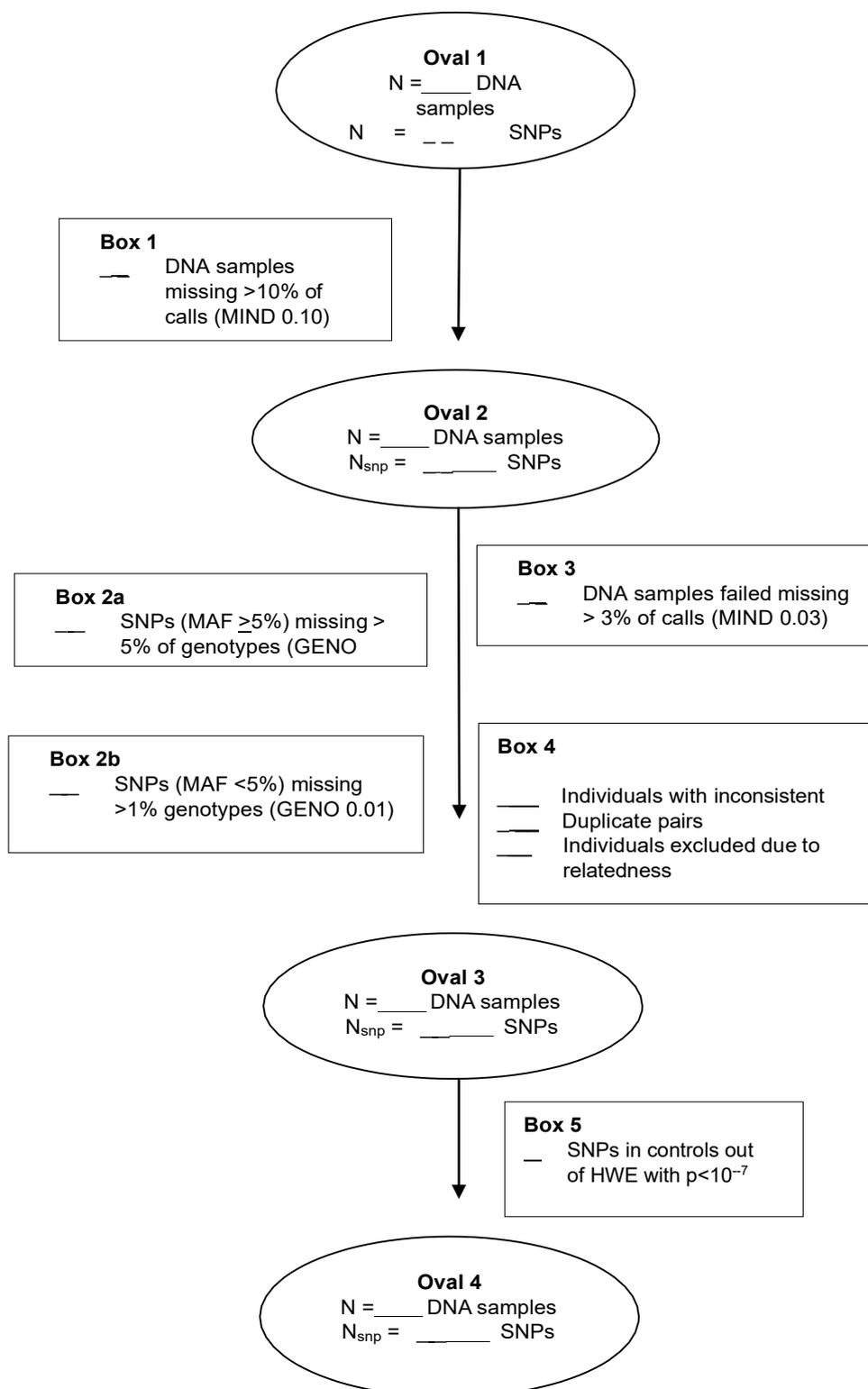
Create a text file called HWE_out.txt with the following SNP in it:

rs2968487

and type the following command:

```
plink --file GWAS_clean3 --exclude HWE_out.txt --recode --out GWAS_clean4
```

There are a number of SNPs with HWE p-values in the range of 10^{-5} to 10^{-6} in the controls. Based on above criterion they will not be excluded however, if they reach genome-wide significance during association testing they SNPs should be further investigated to ensure there is no genotyping error. You can now fill in Box 5 and Oval 4.



Answers to Questions:

Oval 1 and 2 also and Box 1 information:

```
PLINK v1.90b4.9 64-bit (13 Oct 2017)
Options in effect:
  --file GWAS
  --mind 0.10
  --out GWAS_clean_mind
  --recode

Random number seed: 1515434515
16384 MB RAM detected; reserving 8192 MB for main workspace.
Scanning .ped file... done.
Performing single-pass .bed write (6424 variants, 248 people) [Oval 1].
--file: GWAS_clean_mind-temporary.bed + GWAS_clean_mind-temporary.bim +
GWAS_clean_mind-temporary.fam written.
6424 variants loaded from .bim file.
248 people (125 males, 123 females) loaded from .fam.
1 person removed due to missing genotype data (--mind) [Box 1].
ID written to GWAS_clean_mind.irem .
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 247 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Warning: 6 het. haploid genotypes present (see GWAS_clean_mind.hh ); many
commands treat these as missing.
Total genotyping rate in remaining samples is 0.996863.
6424 variants and 247 people pass filters and QC [Oval 2].
Note: No phenotypes present.
--recode ped to GWAS_clean_mind.ped + GWAS_clean_mind.map ... done.
```

Box 2a information:

```
PLINK v1.90b4.9 64-bit (13 Oct 2017)
Options in effect:
  --file MAF_greater_5
  --geno 0.05
  --out MAF_greater_5_clean
  --recode

Random number seed: 1515435189
16384 MB RAM detected; reserving 8192 MB for main workspace.
Scanning .ped file... done.
Performing single-pass .bed write (5868 variants, 247 people).
--file: MAF_greater_5_clean-temporary.bed + MAF_greater_5_clean-temporary.bim +
MAF_greater_5_clean-temporary.fam written.
5868 variants loaded from .bim file.
247 people (125 males, 122 females) loaded from .fam.
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 247 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Warning: 6 het. haploid genotypes present (see MAF_greater_5_clean.hh ); many
commands treat these as missing.
Total genotyping rate is 0.996858.
2 variants removed due to missing genotype data (--geno) [Box2a].
5866 variants and 247 people pass filters and QC.
Note: No phenotypes present.
--recode ped to MAF_greater_5_clean.ped + MAF_greater_5_clean.map ... done.
```

Box 2b information:

```
PLINK v1.90b4.9 64-bit (13 Oct 2017)
Options in effect:
  --file MAF_less_5
  --geno 0.01
  --out MAF_less_5_clean
  --recode
```

```
Random number seed: 1515435255
16384 MB RAM detected; reserving 8192 MB for main workspace.
Scanning .ped file... done.
```

```

Performing single-pass .bed write (556 variants, 247 people).
--file: MAF_less_5_clean-temporary.bed + MAF_less_5_clean-temporary.bim +
MAF_less_5_clean-temporary.fam written.
556 variants loaded from .bim file.
247 people (125 males, 122 females) loaded from .fam.
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 247 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Total genotyping rate is 0.996913.
59 variants removed due to missing genotype data (--geno) [Box2b] .
497 variants and 247 people pass filters and QC.
Note: No phenotypes present.
--recode ped to MAF_less_5_clean.ped + MAF_less_5_clean.map ... done.

```

Box 3 information:

PLINK v1.90b4.9 64-bit (13 Oct 2017)

Options in effect:

```

--file GWAS_MAF_clean
--mind 0.03
--out GWAS_clean2
--recode

```

Random number seed: 1515435827

16384 MB RAM detected; reserving 8192 MB for main workspace.

Scanning .ped file... done.

Performing single-pass .bed write (6363 variants, 247 people).

```

--file: GWAS_clean2-temporary.bed + GWAS_clean2-temporary.bim +
GWAS_clean2-temporary.fam written.

```

6363 variants loaded from .bim file.

247 people (125 males, 122 females) loaded from .fam.

0 people removed due to missing genotype data (--mind) **[Box 3]** .

Using 1 thread (no multithreaded calculations invoked).

Before main variant filters, 247 founders and 0 nonfounders present.

Calculating allele frequencies... done.

Warning: 6 het. haploid genotypes present (see GWAS_clean2.hh); many commands treat these as missing.

Total genotyping rate is 0.99716.

6363 variants and 247 people pass filters and QC.

Note: No phenotypes present.

```

--recode ped to GWAS_clean2.ped + GWAS_clean2.map ... done.

```

Answer to Question 1: Why do you expect the homozygosity rate to be higher on the X chromosome in males than females?

Because males only have one allele for each SNP on the X chromosome they will appear homozygous.

Table 1: Sex check

| FID | IID | PEDEX | SNPSEX | STATUS | F |
|---------|---------|-------|--------|---------|--------|
| NA20506 | NA20506 | 2 | 1 | PROBLEM | 1 |
| NA20530 | NA20530 | 2 | 1 | PROBLEM | 1 |
| NA20766 | NA20766 | 2 | 0 | PROBLEM | 0.2292 |
| NA20771 | NA20771 | 2 | 0 | PROBLEM | 0.2234 |
| NA20757 | NA20757 | 2 | 0 | PROBLEM | 0.2141 |

Table 2: Duplicate and Related Individuals

| FID1 | IID1 | FID2 | IID2 | Z(0) | Z(1) | Z(2) | PI_HAT |
|------|---------|-------|---------|--------|--------|--------|--------|
| M033 | NA19774 | M041 | NA25000 | 0.0000 | 0.0000 | 1.0000 | 1.00 |
| 1344 | NA12057 | 13291 | NA25001 | 0.0000 | 0.0025 | 0.9975 | 1.00 |
| 1444 | NA12739 | 1444 | NA12749 | 0.0026 | 0.9807 | 0.0168 | 0.51 |
| 1444 | NA12739 | 1444 | NA12748 | 0.0026 | 0.9949 | 0.0025 | 0.50 |

FID1- Family ID for 1st individual; IID1 - Individual ID for 1st individual; FID2- Family ID for 2nd individual; IID2 - Individual ID for 2nd individual; Z(0)- P(IBD=0); Z(1)- P(IBD=1); Z(2)- P(IBD=2); PI_HAT-P(IBD=2)+0.5*P(IBD=1) (proportion IBD)

Question 2: How many duplicate pairs do you find (**hint: Pi-Hat = ~1**)? Do pairs with a **Pi-Hat = ~1** have to be duplicate samples? What is another explanation? What proportion would you expect a parent/ child to share IBD? Can you find any such relationship?

There are two duplicate pairs and also a trio (two parents and a child). Parent/child relationships will have a Pi_Hat value of ~0.5, but so will sibpairs. We can tell that this is a parent child relationship by examine Z(0), Z(1) and Z(2). We will retain only one sample from each duplicate pair and the parents NA12749 and NA12748. If you perform mixed-model analysis related individuals can be retained in the sample.

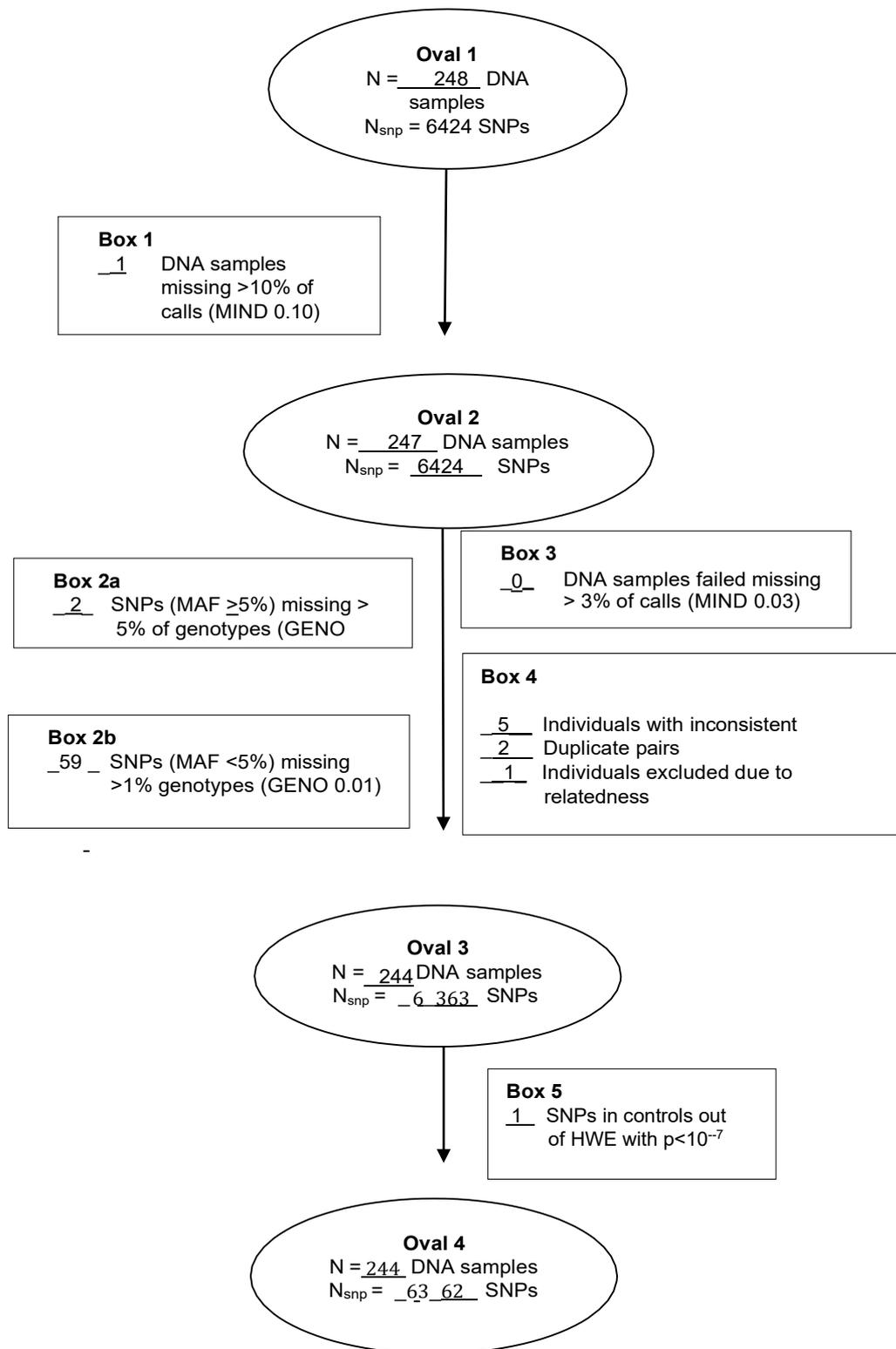
Oval 3 information

```
PLINK v1.90b4.9 64-bit (13 Oct 2017)
Options in effect:
  --file GWAS_clean2
  --out GWAS_clean3
  --recode
  --remove IBS_excluded.txt
Random number seed: 1515440989
16384 MB RAM detected; reserving 8192 MB for main workspace.
Scanning .ped file... done.
Performing single-pass .bed write (6363 variants, 247 people).
--file: GWAS_clean3-temporary.bed + GWAS_clean3-temporary.bim +
GWAS_clean3-temporary.fam written.
6363 variants loaded from .bim file.
247 people (125 males, 122 females) loaded from .fam.
--remove: 244 people remaining.
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 244 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Warning: 6 het. haploid genotypes present (see GWAS_clean3.hh ); many commands
treat these as missing.
Total genotyping rate in remaining samples is 0.997225.
6363 variants and 244 people pass filters and QC [Oval 3].
Note: No phenotypes present.
--recode ped to GWAS_clean3.ped + GWAS_clean3.map ... done.
```

Table 3: Hardy Weinberg Equilibrium

| Fail Cases | | Fail Controls | |
|------------|--------|---------------|------------|
| SNP | pvalue | SNP | pvalue |
| None | | rs2968487 | 2.262e-007 |

```
PLINK v1.90b4.9 64-bit (13 Oct 2017)
Options in effect:
  --exclude HWE_out.txt
  --file GWAS_clean3
  --out GWAS_clean4
  --recode
Random number seed: 1515442367
16384 MB RAM detected; reserving 8192 MB for main workspace.
Scanning .ped file... done.
Performing single-pass .bed write (6363 variants, 244 people).
--file: GWAS_clean4-temporary.bed + GWAS_clean4-temporary.bim +
GWAS_clean4-temporary.fam written.
6363 variants loaded from .bim file.
244 people (123 males, 121 females) loaded from .fam.
--exclude: 6362 variants remaining.
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 244 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Warning: 6 het. haploid genotypes present (see GWAS_clean4.hh ); many commands
treat these as missing.
Total genotyping rate is 0.997229.
6362 variants and 244 people pass filters and QC [Oval 4].
Note: No phenotypes present.
--recode ped to GWAS_clean4.ped + GWAS_clean4.map ... done
```



Genome-Wide Association Exercise

Association Analysis Controlling for Population Substructure

Copyrighted © 2022 Merry-Lynn N. McDonald, Isabelle Schrauwen & Suzanne M. Leal

1. Population Stratification and Association Testing

The dataset from part I of this exercise which you performed data quality control (QC) on was obtained from HapMap Phase III data. It contains CEU founders (Caucasians from Utah), MEX founders (Mexicans from Los Angeles) and TSI (Tuscans from Italy). The CEU pedigree identifiers begin with only numbers e.g., 1347, the MEX pedigree identifiers all start with M e.g., M017 and the TSI pedigree identifiers all start with NA e.g., NA0217. Before we start testing for association, we want to know if there are outliers. Even after removing the outliers when association analysis is performed population substructure and admixture may need to be controlled. If not, we risk observing an association, which is due to a difference in genotype frequencies in cases and controls, because of population substructure/admixture and not because of linkage disequilibrium (LD) between tagSNP(s) and the functional variant(s). We are going to use multidimensional scaling (MDS) and principal components analysis (PCA) within the PLINK software to generate 10 components. **Disclaimer: You usually should not analyze data from European-Americans, Mexican-Americans and Italians together even if you control for population stratification. They can be analyzed separately and the data combined using meta-analysis.**

Note: For a GWAS study instead of this toy study, you will have a denser set of markers of which some will be in LD. You should first prune your SNPs to obtain a subset in linkage equilibrium/weak LD ($R^2 < 0.5$) prior to performing MDS or PCA analysis on the data. Although for association analysis is performed on the entire data set will be analyzed only this a subset of SNPs which are not in LD will be used to construct PCA and MDS components. For more information on how to do this in PLINK see <https://www.cog-genomics.org/plink/1.9/ld>.

```
plink --file GWAS_clean4 --genome --cluster --mds-plot 10
```

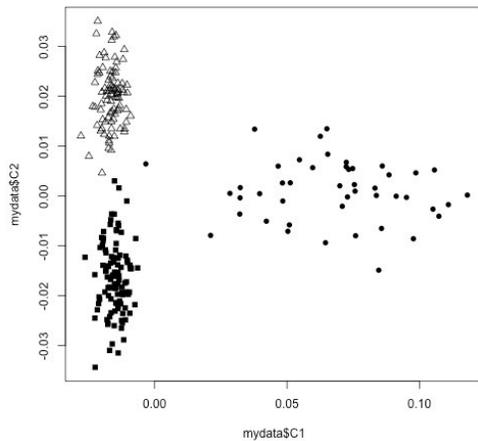
This command outputs the file **plink.mds** that contains the subject IDs and values for the 10 components we just generated. There is another file in your folder called `mds_components.txt`. This file is identical to your **plink.mds** file with the exception that a group column which codes CEU individuals as 1, MEX individuals as 2 and TSI individuals as 3. This is done so when we plot the MDS components in R you can see which group the points belong to and judge how well does the data cluster, e.g., are there outliers. The following commands will generate a jpeg image file containing the mds plot (filename=`mds.jpeg`) in your current working directory. Open R and use the following command:

```
mydata = read.table("mds_components.txt", header=T)
```

```
mydata$pch[mydata$Group==1 ] <--15  
mydata$pch[mydata$Group==2 ] <--16  
mydata$pch[mydata$Group==3 ] <--2
```

```
jpeg("mds.jpeg", height=500, width=500)  
plot(mydata$C1, mydata$C2 ,pch=mydata$pch)  
dev.off()
```

Visualizing population structure using MDS is useful for identifying subpopulations, population stratification and systematic genotyping or sequencing errors, and can also be used to detect individual outliers that may need to be removed, e.g. European-Americans included in a study of African-Americans. MDS coordinates help with



visualizing genetic distances and population substructure. PLINK also offers another dimension reduction, `--pca`, for PCA, the PC components which can also be used for visualizing data to detect outliers in the same manner which was performed using MDS. Additionally, covariates either from either MDS or PCA can be used in a regression model to aid in correcting for population substructure and admixture.

We will now continue performing the analysis using PLINK but will use PCA instead of MDS. We will generate PCs and determine how many PC covariates should be included in the regression model. When SNPs are tested for an association with a trait analysis can be performed, first by including no PC components, then one PC component and then two PC components and so on. Please note that as each PC component is added all the SNPs are analyzed, e.g. a

complete GWAS is performed. Examining λ can aid in determining how many PC components should be included in the analysis. If there is no population stratification or other biases, then λ should equal 1 or ~ 1 . We will use λ to determine how many PC components from our analysis will be added to the logistic regression model. First, estimate λ without adjusting for any PC components:

```
plink --file GWAS_clean4 --pheno pheno.txt --pheno-name Aff --logistic --adjust --out unadj
```

Generated the first 10 PCA values:

```
plink --file GWAS_clean4 --genome --cluster --pca 10 header
```

Eigenvectors are written to `plink.eigenvec`, and top eigenvalues are written to `plink.eigenval`. The 'header' modifier adds a header line to the `.eigenvec` file(s).

And then find out what λ is when we adjust for the first component:

```
plink --file GWAS_clean4 --pheno pheno.txt --pheno-name Aff --covar plink.eigenvec --covar-name PC1 --logistic --adjust --out PC1
```

And the first and second components:

```
plink --file GWAS_clean4 --pheno pheno.txt --pheno-name Aff --covar plink.eigenvec --covar-name PC1-PC2 --logistic --adjust --out PC1-PC2
```

and so forth for all 10 components in the `.log` file completing the table:

Table 1

| | Un-adjusted | PC 1 | PC 1-2 | PC 1-3 | PC 1-4 | PC 1-5 | PC 1-6 | PC 1-7 | PC 1-8 | PC 1-9 | PC1-10 |
|-----------|-------------|------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| λ | | | | | | | | | | | |

The number closest to 1.0, with the least number of PC components, would be the best for adjusting without overfitting and introducing unnecessary noise. You can check your table against the one provided in the answers section.

Go to the **assoc.logistic file that corresponds to that number of components** and make a note of how you named the **.assoc.logistic** file for it and when you did not adjust for any components. Then go back to the R program to load the results and create a jpeg image file containing QQ plots for the adjusted and unadjusted results (using a modified script from <http://www.broad.mit.edu/node/555>) as follows:

```
broadqq <--function(pvals, title)
{
  observed <-- sort(pvals)
  lobs <-- --(log10(observed))

  expected <-- c(1:length(observed))
  lexp <-- --(log10(expected / (length(expected)+1)))

  plot(c(0,7), c(0,7), col="red", lwd=3, type="l", xlab="Expected (-logP)", ylab="Observed (-logP)",
xlim=c(0,max(lobs)), ylim=c(0,max(lobs)), las=1, xaxs="i", yaxs="i", bty="l", main = title)
  points(lexp, lobs, pch=23, cex=.4, bg="black") }

jpeg("qqplot_compare.jpeg", height=1000, width=500)
par(mfrow=c(2,1))
aff_unadj<--read.table("unadj.assoc.logistic", header=TRUE)
aff_unadj.add.p<--aff_unadj[aff_unadj$TEST=="ADD"),]$P
broadqq(aff_unadj.add.p,"Some Trait Unadjusted")
aff_C1C2<--read.table("PC1-PC2.assoc.logistic", header=TRUE)
aff_C1C2.add.p<--aff_C1C2[aff_C1C2$TEST=="ADD"),]$P
broadqq(aff_C1C2.add.p, "Some Trait Adjusted for PC1 and PC2")
dev.off()
```

Now look for SNPs with genome-wide significance using the following R commands:

```
gws_unadj = aff_unadj[which(aff_unadj$P < 0.0000001),]
gws_unadj
gws_adjusted = aff_C1C2[which(aff_C1C2$P < 0.0000001),]
gws_adjusted
```

Note: These are the uncorrected p-values for multiple testing. The p-values which have been corrected using various multiple testing methods can be found in the .adjusted file.

A common question when you have a finding with genome-wide significance in a GWAS is “Is the SNP in a known gene?” One way to look this information up is annotate variants in batch (please look at the annotating exercise for more information). You can do this using the Ensembl Variant Predictor. Go to the website:

http://grch37.ensembl.org/Homo_sapiens/Tools/VEP (GRCh37 version)

Type the rs number(s) of the SNP(s) with genome-wide significance in “Either paste data”, leave all options default and press run. In a few minutes you can view the results of your query.

Question 1: Did this study have a finding with genome-wide significance after adjusting for population substructure? Did you notice any difference in the p-values before and after adjustment for substructure? How many PC components should you include in the regression model. Please also, complete the tables below.

Table 2. SNPS with genome-wide significance unadjusted for substructure:

| CHR | SNP | BP | A1 | TEST | NMISS | OR | STAT | P |
|-----|-----|----|----|------|-------|----|------|---|
| | | | | | | | | |
| | | | | | | | | |

Table 3. SNPs with genome-wide significance adjusted for components 1 and 2:

| CHR | SNP | BP | A1 | TEST | NMISS | OR | STAT | P |
|-----|-----|----|----|------|-------|----|------|---|
| | | | | | | | | |
| | | | | | | | | |

Question 2: Why would you not want to include in your analysis individuals from different ethnic backgrounds even if you control for population substructure?

Question 3. Are any SNPs with genome-wide significance in known genes?

Answers and Output

| Table 1 | Un- adjusted | PC1 | PC1- 2 | PC1 -3 | PC1 -4 | PC1 -5 | PC1 -6 | PC1 -7 | PC1 -8 | PC1 -9 | PC1- 10 |
|---------------|-----------------|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|
| lambda | 1.121 | 1.085 | 1.026 | 1.033 | 1.040 | 1.050 | 1.043 | 1.021 | 1.036 | 1.043 | 1.051 |

Answer to Question 1:

Question 1:

Did this study have a finding with genome-wide significance after adjusting for population substructure? How many PC components should you include in the regression model. Did you notice any difference in the p-values before and after adjustment for substructure?

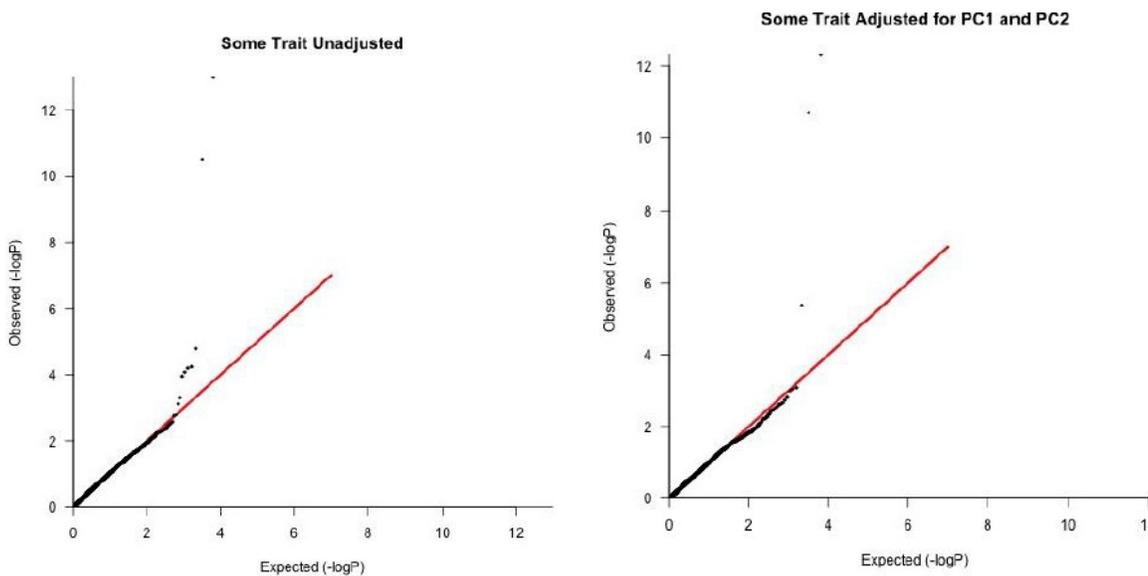
Yes, see tables below. It is best to include to two PC components in the analysis, however the lambda is still inflated. Since we are analyzing three unique populations inclusion of PCs did not adequately control for substructure. If you compare the QQ plots below you can see that for this dataset the most significant SNPs were changed minimally when we adjusted for substructure but some of the moderately significant SNPs became less significant after adjustment. However, in some situations the p-values can become smaller.

Table 2. SNPs with genome-wide significance unadjusted for substructure:

| CHR | SNP | BP | A1 | TEST | NMISS | OR | STAT | P |
|-----|------------|-----------|----|------|-------|---------|--------|----------|
| 8 | rs4571722 | 60326734 | T | ADD | 242 | 0.04126 | -7.436 | 1.04E-13 |
| 4 | rs10008252 | 179853616 | G | ADD | 244 | 0.1665 | -6.639 | 3.16E-11 |

Table 3. SNPs with genome-wide significance adjusted for components 1 and 2:

| CHR | SNP | BP | A1 | TEST | NMISS | OR | STAT | P |
|-----|------------|-----------|----|------|-------|---------|--------|----------|
| 8 | rs4571722 | 60326734 | T | ADD | 242 | 0.04382 | -7.237 | 4.59E-13 |
| 4 | rs10008252 | 179853616 | G | ADD | 244 | 0.13070 | -6.707 | 1.99E-11 |



Question 2: Why would you not want to include in your analysis individuals from different ethnic backgrounds even if you control for population substructure?

Firstly, you may not be able to adequately control for population substructure. Secondly, even if within the different populations the same genes are involved, for common variants LD structure can vary between populations, e.g., the tagSNPs in the different populations can have different allele frequencies, therefore the functional variant will not be tagged equally well in all populations and power can be reduced. It is also possible that different variants are associated, but for common variants, which are very old, usually this is not the cause. If a study involves individuals of different ancestry analysis can be performed separately and the results can be combined via meta-analysis. Studying individuals of different ancestry can be highly beneficial to fine map loci.

Question 3. Are any SNPs with genome-wide significance in known genes?

No, both rs457122 and rs10008252 are intergenic/intronic.

Association Analysis of Sequence Data using Variant Association Tools (VAT) for Complex Traits

Copyright (c) 2022 Gao Wang, Biao Li, Diana Cornejo Sánchez & Suzanne M. Leal

PURPOSE

Variant Association Tools [VAT, Wang et al (2014)] [1] was developed to perform quality control and association analysis of sequence data. It can also be used to analyze genotype data, e.g. exome chip data and imputed data. The software incorporates many rare variant association methods which include but not limited to Combined Multivariate Collapsing (CMC) [2], Burden of Rare Variants (BRV) [3], Weighted Sum Statistic (WSS) [4], Kernel Based Adaptive Cluster (KBAC) [5], Variable Threshold (VT) [6] and Sequence Kernel Association Test (SKAT) [7].

VAT inherits the intuitive command-line interface of Variant Tools (VTools) [8] with re-design and implementation of its infrastructure to accommodate the scale of dataset generated from current sequencing efforts on large populations. Features of VAT are implemented into VTools subcommand system.

RESOURCES

A list of all commands that are used in this exercise can be found at https://statgen.research.bcm.edu/index.php/Main_Page

Basic concepts to handle sequence data using vtools can be found at: <http://varianttools.sourceforge.net/Main/Concepts>

VAT Software documentation

<http://varianttools.sourceforge.net/Main/Documentation>

Genotype data

Exome genotype data was downloaded from the 1000 Genomes pilot data July 2010 release for both the CEU and YRI populations. Only the autosomes are contained in the datasets accompanying this exercise.

The data sets (CEU.exon:2010_03.genotypes.vcf.gz, YRI.exon:2010_03.genotypes.vcf.gz) are available from: ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/pilot_data/release/2010_07/exon/snps

Phenotype data

To demonstrate the association analysis, we simulated a quantitative trait phenotype (BMI). Please note that these phenotypes are NOT from the 1000 genome project.

Computation resources

Due to the nature of next-generation sequencing data, a reasonably powerful machine with high speed internet connection is needed to use this tool for real-world applications. For this reason, in this tutorial we will use a small demo dataset to demonstrate association analysis.

1 Data Quality Control, Annotation and Variant/sample Selection - Part I

1.1 Getting started

Please navigate to the exercise data directory and check the available subcommands by typing:

```
vtools -h
```

Subcommand system is used for various data manipulation tasks (to check details of each subcommand use `vtools name of subcommand -h`). This tutorial is mission oriented and focuses on a subset of the commands that are relevant to variant-phenotype association analysis, rather than introducing them systematically. For additional functionality, please refer to documentation and tutorials online.

Initialize a project

```
vtools init VATDemo
```

OUTPUT

```
INFO: variant tools 2.6.1 : Copyright (c) 2011 - 2014 Bo Peng
INFO: San Lucas FA, Wang G, Scheet P, Peng B (2012) Bioinformatics 28(3):421-422
INFO: Please visit http://varianttools.sourceforge.net for more information.
INFO: Creating a new project VATDemo
```

Command `vtools init` creates a new project in the current directory. A directory can only have one project. After a project is created, subsequent `vtools` calls will automatically load the project in the current directory. Working from outside of a project directory is not allowed.

Import variant and genotype data

Import all vcf files under the current directory:

```
vtools import *.vcf.gz --var_info DP filter --geno_info DP_geno --build hg18 -j1
```

OUTPUT

```
INFO: Importing variants and genotypes from CEU.exon.2010_03.genotypes.vcf.gz (1/2)
CEU.exon.2010_03.genotypes.vcf.gz: 100% [=====]
=====] 4,306 603.0/s in 00:00:07
INFO: 3,489 variants (3,489 new, 3,489 SNVs) from 3,500 lines are imported, with a total of 288,291 genotypes
from 90 samples.
INFO: Importing variants and genotypes from YRI.exon.2010_03.genotypes.vcf.gz (2/2)
YRI.exon.2010_03.genotypes.vcf.gz: 100% [=====]
=====] 5,967 547.2/s in 00:00:10
INFO: 5,175 variants (3,498 new, 5,175 SNVs) from 5,186 lines are imported, with a total of 513,911 genotypes
from 112 samples.
INFO: 8,664 variants (6,987 new, 8,664 SNVs) from 8,686 lines are imported, with a total of 802,202 genotypes
from 202 samples.
```

Command `vtools import` imports variants, sample genotypes and related information fields. The imported variants are saved to the master variant table for the project, along with their information fields.

The command above imports two vcf files sequentially into an empty `vtools` project. The second INFO message in the screen output shows that 3,489 variant sites are imported from the first vcf file, where 3,489 new means that all of them are new because prior to importing the first vcf the project was empty so there was 0 site. The fourth INFO message tells that 5,175 variant sites are imported from the second vcf file, but only 3,498 of them are new (which are not seen in the existing 3,489) because prior to importing the second vcf there were already 3,489 existing variant sites from first vcf.

Thus, $5,175 - 3,498 = 1,677$ variant sites are overlapped sites between first and second vcfs. The last INFO message summarizes that the sum of variant sites contained in both vcfs is $8,664 = 3,489 + 5,175$, where there are 6,987 variant sites after merging variants from both vcfs.

More details about `vtools import` command can be found at <http://varianttools.sourceforge.net/Vtools/Import>

Since the input VCF file uses hg18 as the reference genome while most modern annotation data sources are hg19-based, we need to *liftover* our project using hg19 in order to use various annotation sources in the analysis. `Vtools` provides a command which is based on the tool of USCS liftOver to map the variants from existing reference genome to an alternative build. More details about `vtools liftover` command can be found at <http://varianttools.sourceforge.net/Vtools/Liftover>

```
vtools liftover hg19 --flip
```

```

INFO: Downloading liftOver chain file from UCSC
INFO: Exporting variants in BED format
Exporting variants: 100% [=====]
=====] 6,987 59.3K/s in 00:00:00
INFO: Running UCSC liftOver tool
Updating table variant: 100% [=====]
=====] 6,987 157.6/s in 00:00:44

```

Import phenotype data

The aim of the association test is to find variants that modulate the phenotype BMI. We simulated BMI values for each of the individuals. The phenotype file must be in plain text format with sample names matching the sample IDs in the vcf file(s):

```
head phenotypes.csv
```

```
.phenotypes.csv
```

```

sample_name,panel,SEX,BMI
NA06984,ILLUMINA,1,36.353
NA06985,NA,2,21.415
NA06986,ABI_SOLID+ILLUMINA,1,26.898
NA06989,ILLUMINA,2,25.015
NA06994,ABI_SOLID+ILLUMINA,1,23.858
NA07000,ABI_SOLID+ILLUMINA,2,36.226
NA07037,ILLUMINA,1,32.513
NA07048,ILLUMINA,2,17.57
NA07051,ILLUMINA,1,37.142

```

The phenotype file includes information for every individual, the sample name, sequencing panel, sex and BMI. To import the phenotype data:

```
vtools phenotype --from_file phenotypes.csv --delimiter ","
```

```
OUTPUT
```

```

INFO: Adding phenotype panel of type VARCHAR(24)
INFO: Adding phenotype SEX of type INT
INFO: Adding phenotype BMI of type FLOAT
INFO: 3 field (3 new, 0 existing) phenotypes of 202 samples are updated.

```

Unlike `vtools import`, this command imports/adds properties to samples rather than to variants. More details about `vtools phenotype` command can be found at <http://varianttools.sourceforge.net/Vtools/Phenotype>

View imported data

Summary information for the project can be viewed anytime using the command `vtools show`, which displays various project and system information. More details about `vtools show` can be found at <http://varianttools.sourceforge.net/Vtools/Show>. Some useful data summary commands are:

```

vtools show project
vtools show tables
vtools show table variant
vtools show samples
vtools show genotypes
vtools show fields

```

1.2 Overview of variant and genotype data

Total number of variants

The number of imported variants may be greater than number of lines in the vcf file, because when a variant has two alternative alleles (e.g. A->T/C) it is treated as two separate variants.

```
vtools select variant --count
```

There are 6987 variants in our test data.

`vtools select table condition action` selects from a variant table `table` a subset of variants satisfying a specified condition, and perform an action of

- creating a new variant table if `--to table` is specified.
- counting the number of variants if `--count` is specified.
- outputting selected variants if `--output` is specified.

The `condition` should be a SQL expression using one or more fields in a project (displayed in `vtools show fields`). If the `condition` argument is unspecified, then all variants in the table will be selected. An optional `condition --samples [condition]` can also be used to limit selected variants to specific samples. More details about `vtools select` command can be found at <http://varianttools.sourceforge.net/Vtools/Select>

Genotype Summary

The command `vtools show genotypes` displays the number of genotypes for each sample and names of the available genotype information fields for each sample, e.g. `GT` - genotype; `DP` - genotype read depth. Such information is useful for the calculation of summary statistics of genotypes (e.g. depth of coverage).

```
vtools show genotypes > GenotypeSummary.txt
head GenotypeSummary.txt
```

| sample name | Filename | num genotypes | sample genotype fields |
|-------------|-----------------------------------|---------------|------------------------|
| NA06984 | CEU.exon.2010_03.genotypes.vcf.gz | 3162 | GT,DP geno |
| NA06985 | CEU.exon.2010_03.genotypes.vcf.gz | 3144 | GT,DP geno |
| NA06986 | CEU.exon.2010_03.genotypes.vcf.gz | 3437 | GT,DP geno |
| NA06989 | CEU.exon.2010_03.genotypes.vcf.gz | 3130 | GT,DP geno |
| NA06994 | CEU.exon.2010_03.genotypes.vcf.gz | 3002 | GT,DP geno |
| NA07000 | CEU.exon.2010_03.genotypes.vcf.gz | 3388 | GT,DP geno |
| NA07037 | CEU.exon.2010_03.genotypes.vcf.gz | 3374 | GT,DP geno |
| NA07048 | CEU.exon.2010_03.genotypes.vcf.gz | 3373 | GT,DP geno |
| NA07051 | CEU.exon.2010_03.genotypes.vcf.gz | 3451 | GT,DP geno |

Variant Quality Overview

The following command calculates summary statistics on the variant site depth of coverage (DP). Below is the command to calculate depth of coverage information for all variant sites.

```
vtools output variant "max(DP)" "min(DP)" "avg(DP)" "stdev(DP)" "lower_quartile(DP)" "upper_quartile(DP)" --header
```

| max DP | min DP | avg DP | stdev DP | lower quartile DP | upper quartile DP |
|--------|--------|---------------|---------------|-------------------|-------------------|
| 25490 | 13 | 6815.77028768 | 3434.28040091 | 4301 | 9143 |

In the test data, the maximum DP for variant sites is 25490, minimum DP 13, average DP about 6815, standard deviation of DP about 3434, lower quartile of DP 4301 and upper quartile of DP 9143.

The same syntax can be applied to other variant information or annotation information fields. The command `vtools output name of variant table` outputs properties of variants in a specified variant table. The properties include fields from annotation databases and variant tables, basically fields outputted from command `vtools show fields`, and SQL-supported functions and expressions. There are several freely available SQL resources on the web to learn more about SQL functions and expressions.

It is also possible to view variant-level summary statistic for variants satisfying certain filtering criteria using `vtools select name of variant table` command, for example to count only variants having passed all quality filters:

```
vtools select variant "filter='PASS'" --count
```

All 6987 variants have passed the quality filters. To combine variant filtering and summary statistics:

```
vtools select variant "filter='PASS'" -o "max(DP)" "min(DP)" "avg(DP)" "stdev(DP)" "lower_quartile(DP)" "upper_quartile(DP)" --header
```

The output information of command above will be the same as the previous `vtools` output command, since all variants have passed quality filter.

1.3 Data exploration

Variant level summaries

The command below will calculate:

- total: Total number of genotypes (GT) for a variant
- num: Total number of alternative alleles across all samples
- het: Total number of heterozygote genotypes 1/0
- hom: Total number of homozygote genotypes 1/1
- other: Total number of double-homozygotes 1/2
- min/max/meanDP: Summaries for depth of coverage and genotype quality across samples
- maf: Minor allele frequency
- Add calculated variant level statistics to fields, which can be shown by commands `vtools show fields` and `vtools show table variant`

```
vtools update variant --from_stat 'total=#(GT)' 'num=#(alt)' 'het=#(het)' 'hom=#(hom)' 'other=#(other)' 'minDP=min(DP_genotype)' 'maxDP=max(DP_genotype)' 'meanDP=avg(DP_genotype)' 'maf=maf()'
```

OUTPUT

Counting variants: 100%

```
[=====] 202 22.9/s in 00:00:08
=====] 6,987 22.0K/s in 00:00:00
INFO: Adding variant info field num with type INT
INFO: Adding variant info field hom with type INT
INFO: Adding variant info field het with type INT
INFO: Adding variant info field other with type INT
INFO: Adding variant info field total with type INT
INFO: Adding variant info field maf with type FLOAT
INFO: Adding variant info field minDP with type INT
INFO: Adding variant info field maxDP with type INT
INFO: Adding variant info field meanDP with type FLOAT
Updating variant: 100% [=====]
INFO: 6987 records are updated
```

```
vtools show fields
vtools show table variant
```

Command `vtools update` updates variant info fields (and to a lesser extent genotype info fields) by adding more fields or updating values at existing fields. It does not add any new variants or genotypes, and does not change existing variants, samples, or genotypes. Using three parameters `--from file`, `--from stat`, and `--set`, variant information fields could be updated from external file, sample genotypes, and existing fields. More details about `vtools update` command can be found at

<http://varianttools.sourceforge.net/Vtools/Update>

Summaries for different genotype depth (GD) and genotype quality (GQ) filters

The `--genotypes CONDITION` option restricts calculation to genotypes satisfying a given condition. Later we will remove individual genotypes by `DP geno` filters. The command below will calculate summary statistics genotypes of all samples per variant site. It can assist us in determining filtering criteria for genotype call quality.

```
vtools update variant --from_stat 'totalGD10=#(GT)' 'numGD10=#(alt)' 'hetGD10=#(het)'
'homGD10=#(hom)' 'otherGD10=#(other)' 'mafGD10=maf()' --genotypes "DP geno > 10"
```

OUTPUT

```
Counting variants: 100% [=====]
=====] 202 71.5/s in 00:00:02
INFO: Adding variant info field numGD10 with type INT
INFO: Adding variant info field homGD10 with type INT
INFO: Adding variant info field hetGD10 with type INT
INFO: Adding variant info field otherGD10 with type INT
INFO: Adding variant info field totalGD10 with type INT
INFO: Adding variant info field mafGD10 with type FLOAT
Updating variant: 100% [=====]
=====] 6,976 25.2K/s in 00:00:00
INFO: 6976 records are updated
```

```
vtools show fields
vtools show table variant
```

You will notice the change in genotype counts when applying the filter on genotype depth of coverage and only retaining those genotypes with a read depth greater than 10X. There are now 6976 variant sites after filtering on DP geno>10. Note that some variant sites will become monomorphic after removing genotypes due to low read depth.

Minor allele frequencies (MAFs)

In previous steps, we calculated MAFs for each variant site before and after filtering on genotype read depth. Below is a summary of the results:

```
vtools output variant chr pos maf mafGD10 --header --limit 20
```

| OUTPUT | | | |
|--------|---------|------------------|------------------|
| chr | pos | Maf | mafGD10 |
| 1 | 1105366 | 0.0350877192982 | 0.0512820512821 |
| 1 | 1105411 | 0.00943396226415 | 0.0128205128205 |
| 1 | 1108138 | 0.192307692308 | 0.18023255814 |
| 1 | 1110240 | 0.00561797752809 | 0.0 |
| 1 | 1110294 | 0.228125 | 0.242307692308 |
| 1 | 3537996 | 0.12012987013 | 0.152173913043 |
| 1 | 3538692 | 0.0410256410256 | 0.0432098765432 |
| 1 | 3541597 | 0.00561797752809 | 0.00617283950617 |
| 1 | 3541652 | 0.0444444444444 | 0.0533333333333 |
| 1 | 3545211 | 0.00561797752809 | 0.00581395348837 |
| ... | | | |

Adding "> filename.txt" at the end of the above command will write the output to a file.

Next, we examine population specific MAFs. Our data is imported from two files, a CEU dataset (90 samples) and an YRI dataset (112 samples). To calculate allele frequency for each population, let us first assign an additional RACE phenotype (0 for YRI samples and 1 for CEU samples):

```
vtools phenotype --set "RACE=0" --samples "filename like 'YRI%'"
vtools phenotype --set "RACE=1" --samples "filename like 'CEU%'"
vtools show samples --limit 10
```

| OUTPUT | | | | | |
|-------------|---------------------------|--------------------|-----|--------|-------------------------|
| sample_name | filename | panel | SEX | BMI | RACE |
| NA06984 | CEU.exon...notypes.vcf.gz | ILLUMINA | 1 | 36.353 | 1 |
| NA06985 | CEU.exon...notypes.vcf.gz | . | 2 | 21.415 | 1 |
| NA06986 | CEU.exon...notypes.vcf.gz | ABI_SOLID+ILLUMINA | 1 | 26.898 | 1 |
| NA06989 | CEU.exon...notypes.vcf.gz | ILLUMINA | 2 | 25.015 | 1 |
| NA06994 | CEU.exon...notypes.vcf.gz | ABI_SOLID+ILLUMINA | 1 | 23.858 | 1 |
| NA07000 | CEU.exon...notypes.vcf.gz | ABI_SOLID+ILLUMINA | 2 | 36.226 | 1 |
| NA07037 | CEU.exon...notypes.vcf.gz | ILLUMINA | 1 | 32.513 | 1 |
| NA07051 | CEU.exon...notypes.vcf.gz | ILLUMINA | 1 | 37.142 | 1 |
| NA07346 | CEU.exon...notypes.vcf.gz | . | 2 | 30.978 | 1 (192 records omitted) |

Population specific MAF calculations will be performed using those genotypes that passed the read depth filter (DP geno>10)

```

vtools update variant --from_stat 'CEU_mafGD10=maf()' --genotypes 'DP_geno>10' --samples "RACE=1"
vtools update variant --from_stat 'YRI_mafGD10=maf()' --genotypes 'DP_geno>10' --samples "RACE=0"
vtools output variant chr pos mafGD10 CEU_mafGD10 YRI_mafGD10 --header --limit 10

```

| OUTPUT | | | | |
|--------|---------|------------------|------------------|------------------|
| chr | Pos | mafGD10 | CEU_mafGD10 | YRI_mafGD10 |
| 1 | 1105366 | 0.0512820512821 | 0.0512820512821 | 0.0 |
| 1 | 1105411 | 0.0128205128205 | 0.0128205128205 | 0.0 |
| 1 | 1108138 | 0.18023255814 | 0.0212765957447 | 0.371794871795 |
| 1 | 1110240 | 0.0 | 0.0 | 0.0 |
| 1 | 1110294 | 0.242307692308 | 0.025 | 0.428571428571 |
| 1 | 3537996 | 0.152173913043 | 0.170454545455 | 0.135416666667 |
| 1 | 3538692 | 0.0432098765432 | 0.0833333333333 | 0.00595238095238 |
| 1 | 3541597 | 0.00617283950617 | 0.00617283950617 | 0.0 |
| 1 | 3541652 | 0.0533333333333 | 0.0533333333333 | 0.0 |
| 1 | 3545211 | 0.00581395348837 | 0.00581395348837 | 0.0 |

You will observe zero values because some variant sites are monomorphic or they are population specific.

Sample level genotype summaries

Similar operations could be performed on a sample level instead of on a variant level. More details about obtaining genotype level summary information using `vtools phenotype --from stat` can be found at <http://varianttools.sourceforge.net/Vtools/Phenotype>

```

vtools phenotype --from_stat 'CEU_totalGD10=#(GT)' 'CEU_numGD10=#(alt)' --genotypes 'DP_geno>10' --samples "RACE=1"
vtools phenotype --from_stat 'YRI_totalGD10=#(GT)' 'YRI_numGD10=#(alt)' --genotypes 'DP_geno>10' --samples "RACE=0"

```

180 values of 2 phenotypes (2 new, 0 existing) of 90 samples are updated.
224 values of 2 phenotypes (2 new, 0 existing) of 112 samples are updated.

```

vtools phenotype --output sample nameCEU totalGD10CEU numGD10YRI totalGD10YRI numGD10 --header

```

| OUTPUT | | | | |
|-------------|---------------|-------------|---------------|-------------|
| sample_name | CEU_totalGD10 | CEU_numGD10 | YRI_totalGD10 | YRI_numGD10 |
| NA06984 | 2774 | 849 | NA | NA |
| NA06985 | 1944 | 570 | NA | NA |
| NA06986 | 3386 | 1029 | NA | NA |
| NA06989 | 2659 | 819 | NA | NA |
| NA06994 | 1730 | 486 | NA | NA |
| ... | | | | |
| NA19257 | NA | NA | 4969 | 1229 |
| NA19259 | NA | NA | 4182 | 1005 |
| NA19260 | NA | NA | 4404 | 1076 |
| NA19262 | NA | NA | 4308 | 1044 |
| NA19266 | NA | NA | 4878 | 1211 |

1.4 Variant Annotation

For rare variant aggregated association tests, we want to focus on analyzing aggregating variants having potential functional contribution to a phenotype. Thus, each variant site needs to be annotated for its functionality. Annotation is performed using variant annotation tools [7] which implements an ANNOVAR pipeline for variant function annotation [9]. More details about the ANNOVAR pipeline can be found at <http://varianttools.sourceforge.net/Pipeline/Annovar>

```
vtools execute ANNOVAR geneanno
```

OUTPUT

```
INFO: Running vtools update variant --from_file cache/annovar_input.variant_function --format ANNOVAR_variant_function --var_info region_type, region_name
...
Running vtools update variant --from_file cache/annovar_input.exonic_variant_function --format ANNOVAR_exonic_variant_function --var_info mut_type, function
...
INFO: Fields mut_type, function of 6,929 variants are updated
```

The following command will output the annotated variant sites to the screen.

```
vtools output variant chr pos ref alt mut_type --limit 20 --header
```

OUTPUT

```
chr  pos      ref  alt  mut_type
1    1105366    T    C    nonsynonymous SNV
1    1105411    G    A    nonsynonymous SNV
1    1108138    C    T    synonymous SNV
1    1110240    T    A    nonsynonymous SNV
1    1110294    G    A    nonsynonymous SNV
1    3537996    T    C    synonymous SNV
...
```

Many more annotation sources are available which are not covered in this tutorial. Please read <http://varianttools.sourceforge.net/Annotation> for annotation databases, and <http://varianttools.sourceforge.net/Pipeline> for annotation pipelines.

1.5 Data Quality Control (QC) and Variant Selection

Ti/Tv ratio evaluations

Before performing any data QC we examine the transition/transversion (Ti/Tv) ratio for all variant sites. Note that here we are obtaining Ti/Tv ratios for the entire sample, Ti/Tv ratios can also be obtained for each sample.

```
vtools_report trans_ratio variant -n num
```

| num of transition | num of transversion | ratio |
|-------------------|---------------------|---------|
| 161,637 | 44,641 | 3.62082 |

The command above counts the number of transition and transversion variants and calculates its ratio. More details about vtools report trans ratio command can be found at <http://varianttools.sourceforge.net/VtoolsReport/TransRatio>

If only genotype calls having depth of coverage greater than 10 are considered:

```
vtools_report trans_ratio variant -n numGD10
```

| num of transition | num of transversion | ratio |
|-------------------|---------------------|---------|
| 140,392 | 38,710 | 3.62676 |

We can see that Ti/Tv ratio has increase slightly if low depth of coverage calls are removed. There is only a small change in the Ti/Tv ratio since only a few variant sites become monomorphic and are no longer included in the calculation. In practice Ti/Tv ratios can be used to evaluate which threshold should be used in data QC.

Removal of low quality variant sites

We should not need to remove any variant site based on read depth because all variants passed the quality filter. To demonstrate removal of variant sites, let us

```
remove those with a total read depth ${\(\le\)} 15.
vtools select variant "DP<15" -t to_remove
vtools show tables
vtools remove variants to_remove -v0
vtools show tables
```

We can see that one variant site has been removed from master variant table. The `vtools remove` command can remove various items from the current project. More details about `vtools remove` command can be found at <http://varianttools.sourceforge.net/Vtools/Remove>. Using a combination of `select/remove` subcommands low quality variant sites can be easily filtered out. The `vtools show fields`, `vtools show tables`, and `vtools show table variant` commands will allow you to see the new/updated fields and tables you have added/changed to the project.

Filter genotype calls by quality

We have calculated various summary statistics using the command `--genotypes 'CONDITION'` but we have not yet removed genotypes having genotype read depth of coverage lower than 10X. The command below removes these genotypes.

```
vtools remove genotypes "DP geno<10" -v0
```

Select variants by annotated functionality

To select potentially functional variants for association mapping:

```
vtools select variant "mut_type like 'non%' or mut_type like 'stop%' or region_type='splicing' "
-t v_func
vtools show tables
```

The command above selects variant sites that are either nonsynonymous (by condition `"mut type like 'non%'"`) or stop-gain/stop-loss (by condition `mut -type like 'stop%'"`) or alternative splicing (by condition `region-type='splicing'"`)

3367 functional variant sites are selected

2 Association Tests for Quantitative Traits - Part II

2.1 View phenotype data

```
vtools show samples --limit 5
```

| sample_name | filename | panel | SEX | BMI | ... |
|-------------|---------------------------|--------------------|-----|--------|-----|
| NA06984 | CEU.exon...notypes.vcf.gz | ILLUMINA | 1 | 36.353 | ... |
| NA06985 | CEU.exon...notypes.vcf.gz | . | 2 | 21.415 | ... |
| NA06986 | CEU.exon...notypes.vcf.gz | ABI_SOLID+ILLUMINA | 1 | 26.898 | ... |
| NA06989 | CEU.exon...notypes.vcf.gz | ILLUMINA | 2 | 25.015 | ... |
| NA06994 | CEU.exon...notypes.vcf.gz | ABI_SOLID+ILLUMINA | 1 | 23.858 | ... |

2.2 Create sub-projects for association analysis with CEU samples

We want to carry out the association analysis for CEU and YRI separately. It is recommended that we create two projects containing variants and samples for each population. This will greatly improve the computational efficiency. Note that we need to create empty folders to hold each of the projects:

```
vtools select variant --samples "RACE=1" -t CEU
mkdir -p ceu
cd ceu
vtools init ceu --parent ../ --variants CEU --samples "RACE=1" -- build
hg19
vtools show project
```

The above `vtools init --parent` command can create a project from a parent project. More details can be found at <http://varianttools.sourceforge.net/Vtools/Init>

From now on we will only demonstrate analysis of CEU samples (and all the following commands in this chapter will be executed for this project), although the same commands will be applicable for YRI samples. After completing the analysis of CEU samples please use the same commands to analyze the YRI data set. You should not analyze the data from different populations together, once you have the p-values from each analysis, you may perform a meta-analysis.

2.3 Subset data by MAFs

To carry out association tests we need to treat common and rare variants separately. The dataset for our tutorial has very small sample size, but with large sample size it is reasonable to define rare variants as having observed $MAF < 0.01$, and common variants as variants having observed $MAF \geq 0.05$. First, we create variant tables based on calculated alternative allele frequencies for both populations

```
vtools select variant "CEU_mafGD10>=0.05" -t common_ceu
vtools select v_funct "CEU_mafGD10<0.01" -t rare_ceu
```

Notice that for selection of rare variants we only keep those that are annotated as functional (chosen from `v_funct` table). There are 1450 and 604 variant sites selected for $MAF \geq 0.05$ and $MAF < 0.01$, respectively.

2.4 Annotate variants to genes

For gene based rare variant analysis we need annotations that tell us the boundaries of genes. We use the `refGene` annotation database for this purpose.

```
vtools use refGene
```

```
----- [OUTPUT] -----
INFO: Downloading annotation database annoDB/refGene-hg19_20130904.ann
INFO: Downloading annotation database from annoDB/refGene-hg19_20130904.DB.gz refGene-hg19_20130904.DB.gz: 100%
[===== ==] 8,056,345.0 411.6K/s in
00:00:19
INFO: Using annotation DB refGene as refGene in project ceu.
INFO: Known human protein-coding and non-protein-coding genes taken from the NCBI RNA reference sequences
collection (RefSeq).
```

```
vtools show annotation refGene
```

```

Annotation database refGene (version hg19_20130904)
Description:      Known human protein-coding and non-protein-coding genes taken from the NCBI RNA reference sequences collection (RefSeq).
Database type:   range
Reference genome hg19: chr, txStart, txEnd
name (char)     Gene name
chr (char)
strand (char)   which DNA strand contains the observed alleles
txStart (int)   Transcription start position (1-based)
txEnd (int)     Transcription end position
cdsStart (int)  Coding region start (1-based)
cdsEnd (int)    Coding region end
exonCount (int) Number of exons
exonStarts (char) Starting point of exons (adjusted to 1-based positions)
exonEnds (char) Ending point of exons
score (int)     Score
name2 (char)    Alternative name
cdsStartStat (char) cds start stat, can be 'non', 'unk', 'incompl', and 'cpl'
cdsEndStat (char)  cds end stat, can be 'non', 'unk', 'incompl', and 'cpl'

```

The names of genes are contained in the `refGene.name2` field. The `vtools use` command, attaches an annotation database to the project, effectively incorporating one or more attributes available to variants in the project. More details about `vtools use` command can be found at <http://varianttools.sourceforge.net/Vtools/Use>

2.5 Association testing of common/rare variants

The association test program `VAT` is currently under development and is temporarily implemented as the `vtools associate` subcommand. To list available association test options

```

vtools associate -h
vtools show tests
vtools show test LinRegBurden

```

Note that we use the quantitative trait BMI as the phenotype, and we will account for “SEX” as a covariate in the regression framework. More details about `vtools associate` command can be found at <http://varianttools.sourceforge.net/Vtools/Associate>

Analysis of common variants

By default, the program will perform single variant tests using a simple linear model, and the Wald test statistic will be evaluated for p-values:

```

vtools associate common_ceu BMI --covariate SEX -m "LinRegBurden --alternative 2"
-j1 --to_db EA_CV > EA_CV.asso.res

```

```

INFO: 90 samples are found
INFO: 1450 groups are found
Loading genotypes: 100% [=====] 90 56.7/s in 00:00:01
Testing for association: 100% [=====] 1,450/5 684.5/s in 00:00:02
INFO: Association tests on 1450 groups have completed. 5 failed.
INFO: Using annotation DB EA_CV as EA_CV in project ceu.
INFO: Annotation database used to record results of association tests. Created on Fri, 25 Mar 2016 17:45:52
INFO: 1450 out of 3484 variant.chr, variant.pos are annotated through annotation database EA_CV

```



Note

Option `-j1` specifies that 1 CPU core be used for association testing. You may use larger number of jobs for real world data analysis, e.g., use `-j16` if your computational resources has 16 CPU cores available. Linux command `cat /proc/cpuinfo` shows the number of cores and other information related to the CPU on your computer.

Association tests on 1450 groups have completed. 5 failed.

The following command displays error messages about the failed tests. In each case, the sample size was too small to perform the regression analysis.

```
grep -i error *.log
```

OUTPUT

```
2016-03-25 12:45:57,373: DEBUG: An ERROR has occurred in process 0 while processing '6:30018583': Sample size too small (2) to be analyzed for '6:30018583'.
2016-03-25 12:45:57,378: DEBUG: An ERROR has occurred in process 0 while processing '6:30018721': Sample size too small (2) to be analyzed for '6:30018721'.
2016-03-25 12:45:57,574: DEBUG: An ERROR has occurred in process 0 while processing '7:148552665': Sample size too small (2) to be analyzed for '7:148552665'.
2016-03-25 12:45:57,662: DEBUG: An ERROR has occurred in process 0 while processing '8:145718728': Sample size too small (4) to be analyzed for '8:145718728'.
2016-03-25 12:45:57,669: DEBUG: An ERROR has occurred in process 0 while processing '9:205057': Sample size too small(4) to be analyzed for '9:205057'.
```

A summary from the association test is written to the file `EA CV.asso.res`. The first column indicates the variant chromosome and base pair position so that you may follow up on the top signals using various annotation sources that we will not introduce in this tutorial. The result will be automatically built into annotation database if `--to db` option is specified.

You may view the summary using the `less` command

```
less EA_CV.asso.res
```

To sort the results by p-value and output the first 10 lines of the file use the command:

```
sort -g -k7 EA_CV.asso.res | head
```

If you obtain significant p-values be sure to also observe the accompanying sample size. Significant p-values from too small of a sample size may not be results you can trust.

Also, depending on your phenotype you may have to add additional covariates to your analysis. VAT allows you to test many different models for the various phenotypes and covariates. P-values for covariates are also reported.

Similar to using an annotation database, you can use the results from the association test to annotate the project and follow up variants of interest, for example:

```
vtools show fields
```

```
association analysis result columns
```

```
Field name Description
```

```
EA_CV.variant_chr
```

```
EA_CV.variant_pos
```

```
EA_CV.sample_size_LinRegBurden
```

```
EA_CV.beta_x_LinRegBurden
```

```
EA_CV.pvalue_LinRegBurden
```

```
EA_CV.wald_x_LinRegBurden
```

```
EA_CV.beta_2_LinRegBurden
```

```
EA_CV.beta_2_pvalue_LinRegBurden
```

```
EA_CV.wald_2_LinRegBurden
```

```
variant_chr
```

```
variant_pos
```

```
sample_size
```

```
test statistic. In the context of regression, this is estimate of effect size for x p-value
```

```
Wald statistic for x (beta_x/SE(beta_x))
```

```
estimate of beta for covariate 2
```

```
p-value for covariate 2
```

```
Wald statistic for covariate 2
```

You see additional annotation fields starting with EA CV, the name of the annotation database you just created from association test (if you used the `--to db` option mentioned above). You can use them to easily select/output variants of interest. More details about outputting annotation fields for significant findings can be found at <http://varianttools.sourceforge.net/Vtools/Output>

Burden test for rare variants (BRV)

BRV method uses the count of rare variants in given genetic region for association analysis, regardless of the region length.

We use the `-g` option and use the 'refGene.name2' field to define the boundaries of a gene. By default, the test is a linear regression using aggregated counts of variants in a gene region as the regressor.

```
vtools associate rare_ceu BMI --covariate SEX -m "LinRegBurden --alternative 2" -g refGene.name2 -j1 --to_db EA_RV > EA_RV.asso.res
```

OUTPUT

```
INFO: 90 samples are found
INFO: 254 groups are found
Loading genotypes: 100% [=====] 90 48.6/s in 00:00:01
Testing for association: 100% [=====] 254/20 685.4/s in 00:00:00
INFO: Association tests on 254 groups have completed. 20 failed.
INFO: Using annotation DB EA_RV as EA_RV in project ceu.
INFO: Annotation database used to record results of association tests. Created on Fri, 25 Mar 2016 17:47:26
INFO: 254 out of 25360 refGene.refGene.name2 are annotated through annotation database EA_RV
```

Association tests on 254 groups have completed. 20 failed. To view failed tests:

```
grep -i error *.log | tail -10
```

```
OUTPUT
2016-03-25 12:49:49,553: DEBUG: An ERROR has occurred in process 0 while processing 'ABCC1': No variant found in
geno type data for 'ABCC1'.
2016-03-25 12:49:49,620: DEBUG: An ERROR has occurred in process 0 while processing 'ANO9': No variant found in
genot ype data for 'ANO9'.
2016-03-25 12:49:49,781: DEBUG: An ERROR has occurred in process 0 while processing 'C10orf71': No variant found
in g enotype data for 'C10orf71'.
2016-03-25 12:49:49,875: DEBUG: An ERROR has occurred in process 0 while processing 'CCDC127': No variant found
in ge notype data for 'CCDC127'.
2016-03-25 12:49:50,313: DEBUG: An ERROR has occurred in process 0 while processing 'FBXL13': No variant found
in genotype data for 'FBXL13'.
...
```

The output file is `EA_RV.asso.res`. The first column is the gene name, with corresponding p-values in the sixth column for the entire gene.

```
less EA_RV.asso.res
```

You can also sort these results by p-value using command:

```
sort -g -k6 EA_RV.asso.res | head
```

Variable thresholds test for rare variants (VT)

The variable thresholds (VT) method will carry out multiple testing in the same gene region using groups of variants based on observed variant allele frequencies. This test will maximize over statistics thus obtain a final test statistic, and calculate the empirical p-value so that multiple comparisons are adjusted for correctly.

We will use adaptive permutation to obtain empirical p-values. Therefore, to avoid performing too large number of permutations we use a cutoff to limit the number of permutations when the p-value is greater than 0.0005, e.g. not all 100,000 permutations are performed. Generally, even more permutations are used but we limit it to 100,000 to save time for this exercise.

The command using variable thresholds method on our data is:

```
vtools associate rare_ceu BMI --covariate SEX -m "VariableThresholdsQt --alternative 2 -p 100000 \ --adaptive 0.0005" -g refGene.name2 -j1 --to_db EA_RV > EA_RV_VT.asso.res
```

To view test that failed,

```
grep -i error *.log | tail -10
```

To view results,

```
less EA_RV_VT.asso.res
```



Note

The p values you obtained for VT might be slightly different for each run. This is due to the randomness in permutation tests.

Sort and output the lowest p-values using the command:

```
sort -g -k6 EA_RV_VT.asso.res | head
```

Why do some tests fail?

Notice that `vtools associate` command will fail on some association test units. Instances of failure are printed to terminal in red and are recorded in the project log file. Most failures occur due to an association test unit having too few samples or number of variants (for gene based analysis). You should view these error messages after each association scan is complete, e.g., using the Linux command `grep -i error *.log` and make sure you are informed of why failures occur.

In the variable thresholds analysis above, gene `ABCC1` failed the association test. If we look at this gene more closely we can see which variants are being analyzed by our test:

```
vtools select rare_ceu "refGene.name2='ABCC1'" -o chr pos ref alt CEU_mafGD10 numGD10 mut_type --header
```

| chr | Pos | ref | alt | CEU_mafGD10 | numGD10 | mut type |
|-----|----------|-----|-----|-------------|---------|-------------------|
| 16 | 16178858 | T | C | 0.0 | 243 | nonsynonymous SNV |

After applying our QC filters we are left with one variant within the `ABCC1` gene to analyze. Because the MAF for this variant is 0.0 there are no variants in the gene to analyze so that this gene is ignored. Note that all individuals are homozygous for the alternative allele for this variant site.

2.6 Association analysis of YRI samples

Procedures for YRI sample association analysis is the same as for CEU samples as previously has been described, thus is left as an extra exercise for you to work on your own. Commands to perform analysis for YRI are found below:

```
BASH
cd ..
vtools select variant --samples "RACE=0" -t YRI
mkdir -p yri; cd yri
vtools init yri --parent ../ --variants YRI --samples "RACE=0" --
build hg19
vtools select variant "YRI_mafGD10>=0.05" -t common_yri vtools
select v_funct "YRI_mafGD10<0.01" -t rare_yri
vtools use refGene
vtools associate common_yri BMI --covariate SEX -m "LinRegBurden --alternative 2" -j1 --to_db YA_CV > YA_CV.asso.res
vtools associate rare_yri BMI --covariate SEX -m "LinRegBurden --alternative 2" -g refGene.name2 -j1 --to_db YA_RV
> YA_RV.asso.res
vtools associate rare_yri BMI --covariate SEX -m "VariableThresholdsQt --alternative 2 -p 100000 \
--adaptive 0.0005" -g refGene.name2 -j1 --to_db YA_RV >
YA_RV_VT.asso.res
cd ..
```

2.7 Meta-analysis

Here we demonstrate the application of meta-analysis to combine association results from the two populations via `vtools report meta analysis`. More details about `vtools report meta analysis` command can be found at

<http://varianttools.sourceforge.net/VtoolsReport/MetaAnalysis>

The input to this command are the association results files generated from previous steps, for example:

```
vtools_report meta_analysis ceu/EA_RV_VT.asso.res yri/YA_RV_VT.asso.res --beta 5 --pval 6 --se 7 -n 2 --link 1 > ME\ TA_RV_VT.asso.res
```

To view the results,

```
cut -f1,3 META_RV_VT.asso.res | head
```

| refgene_name2 | pvalue | meta |
|---------------|-----------|------|
| CASP7 | 4.751E-01 | |
| POLR2J2 | 3.110E-01 | |
| GNAO1 | 6.875E-02 | |
| C18orf25 | 9.456E-01 | |
| GBP7 | 3.498E-01 | |
| MSH5 | 5.905E-01 | |
| OR51B5 | 5.521E-01 | |
| MAPK14 | 3.063E-01 | |
| BAZ2B | 7.941E-01 | |

Note that for genes that only appears in one study but not the other, or only have a valid p-value in one study but not the other, will be ignored from meta-analysis.

2.8 Summary

Analyzing variants with VAT is much like any other analysis software with a general workflow of:

- Variant level cleaning
- Sample genotype cleaning
- Variant annotation and phenotype information processing
- Sample/variant selection
- Association analysis
- Interpreting the findings

The data cleaning and filtering conditions within this exercise should be considered as general guidelines. Your data may allow you to be laxer with certain criteria or force you to be more stringent with others.

References

- [1] Wang, G.T., Peng, B., and Leal, S.M. (2014). Variant Association Tools for Quality Control and Analysis of Large-Scale Sequence and Genotyping Array Data. *Am. J. Hum. Genet.* 94, 770783
- [2] Li B, Leal SM. Methods for detecting associations with rare variants for common diseases: application to analysis of sequence data. *Am J Hum Genet* 2008 83:311-21
- [3] Auer PL, Wang G, Leal SM. Testing for rare variant associations in the presence of missing data. *Genet Epidemiol* 2013 37:529-38
- [4] Liu DJ, Leal SM. A novel adaptive method for the analysis of next-generation sequencing data to detect complex trait associations with rare variants due to gene main effects and interactions. *PLoS Genet* 2010 6:e1001156
- [5] Madsen BE, Browning SR. A groupwise association test for rare mutations using a weighted sum statistic. *PLoS Genet* 2009 5:e1000384
- [6] Price AL, Kryukov GV, de Bakker PI, Purcell SM, Staples J, Wei LJ, Sunyaev SR. Pooled association tests for rare variants in exon-resequencing studies. *Am J Hum Genet* 20010 86:832-8
- [7] Wu MC, Lee S, Cai T, Li Y, Boehnke M, Lin X. Rare-variant association testing for sequencing data with the sequence kernel association test. *Am J Hum Genet* 2011 89:82-93
- [8] Lucas FAS, Wang G, Scheet P, Peng B. Integrated annotation and analysis of genetic variants from next-generation sequencing studies with variant tools. *Bioinformatics* 2012 28:421-2
- [9] Wang K, Li M, Hakonarson H. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Res* 2010 38:e164
- [10] Manichaikul A, Mychaleckyj JC, Rich SS, Daly K, Sale M, Chen WM. Robust relationship inference in genome-wide association studies. *Bioinformatics* 2010 26(22):2867-2873
- [11] Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MAR, Bender D, Maller J, Sklar P, de Bakker PIW, Daly MJ & Sham PC. PLINK: a toolset for whole-genome association and population-based linkage analysis. *Am J Hum Genet.* 2007 81:559-75

Questions

Question 1 List the four lowest p-values and associated variants or gene regions for the EA CV.asso.res, EA RV.asso.res, and EA RV VT.asso.res test outputs, which are results from single variant Wald test, rare variant BRV and VT tests, respectively, using the European American (CEU) population. Also, list the results using Yoruba African (YRI) population from YA CV.asso.res, YA RV.asso.res and YA RV VT.asso.res

EA CV.asso.res - single variant tests using CEU

- 1) _____ ; 2) _____
- 3) _____ ; 4) _____

EA RV.asso.res - BRV tests using CEU

- 1) _____ ; 2) _____
- 3) _____ ; 4) _____

EA RV VT.asso.res - VT tests using CEU

- 1) _____ ; 2) _____
- 3) _____ ; 4) _____

YA CV.asso.res - single variant tests using YRI

- 1) _____ ; 2) _____
- 3) _____ ; 4) _____

YA RV.asso.res - BRV tests using YRI

- 1) _____ ; 2) _____
- 3) _____ ; 4) _____

YA RV VT.asso.res - VT tests using YRI

- 1) _____ ; 2) _____
- 3) _____ ; 4) _____

Question 2 List any gene regions that show up in the lowest eight p-values for both the BRV and the VT tests. Why might the p-values for the VT tests be higher than the p-values for the BRV tests? Are any of the top p-value hits significant? Why or why not?

Answers

Question 1

EA CV.asso.res

- 107888886 0.000105185
- 1) 15869257 0.00038548
- 2) 56293401 0.000386273
- 3) 15869388 0.00279873

EA RV.asso.res

- 1) CIDEA 0.00504822
- 2) UGT1A10 0.00549521
- 3) UGT1A5 0.00549521
- 4) UGT1A6 0.00549521

EA_RV_VT.asso.res

- 1) UGT1A9 0.007996
- 2) CPED1 0.00999001
- 3) UGT1A10 0.00999001
- 4) UGT1A6 0.011988

YA CV.asso.res

- 1) 107888886 0.00000974
- 2) 6003506 0.000211457
- 3) 25901623 0.001329
- 4) 3392651 0.00194995

YA RV.asso.res

- 1) EMILIN2 0.00262487
- 2) ASIC2 0.0551664
- 3) MDN1 0.0593085
- 4) BAZ2B 0.0607625

YA_RV_VT.asso.res

- 1) EMILIN2 0.00533156
- 2) MDN1 0.013986
- 3) VLDLR 0.01998
- 4) LRRC9 0.025974

Question 2: The p-values do not achieve significance based on the corrected p values above (Bonferroni correction for multiple tests). Since the BMI values were randomly generated for each individual it is unlikely that any of the p-values for the single variant and aggregation tests would have achieved significance. Also, because of the multiple testing, the p-values for the VT tests might be higher than the p-values for the BRV tests.

Association Analysis of Sequence Data using PLINK/SEQ (PSEQ)

Copyright (c) 2022 Stanley Hooker, Biao Li, Di Zhang and Suzanne M. Leal

Purpose

PLINK/SEQ (PSEQ) is an open-source C/C++ library for working with human genetic variation data. The specific focus is to provide a platform for analytic tool development for variation data from large-scale resequencing and genotyping projects, particularly whole-exome and whole-genome studies. PSEQ is independent of, but designed to be complementary to, the existing PLINK (Purcell *et al.*, 2007) package. Here we give an overview of analysis of exome sequence data using PSEQ.

Software Resource

This tutorial was completed with PSEQ 0.10, (released on 14-Jul-2014) available from <https://atgu.mgh.harvard.edu/plinkseq/download.shtml>. Links to PSEQ documentation can also be found on the webpage. Below is an outline of what PSEQ documentation offers:

- Basic Syntax and Conventions
- Project Management
- Data Input
- Attaching Auxiliary Data
- Viewing Data
- Data Output
- Summary Statistics
- Association Analysis
- Locus Database Operations
- Reference Database Operations
- Miscellaneous commands

Exercise Genotype Data

Autosomal exome genotype data was downloaded from the 1000 Genomes pilot data July 2010 release for both the CEU (Utah residents with Northern and Western European ancestry) and YRI (Yoruba in Ibadan, Nigeria) populations. The data sets (CEU.exon.201003.genotypes.vcf.gz and YRI.exon.201003.genotypes.vcf.gz) are available from:

ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/pilot_data/release/2010_07/exon/snps

The genomic co-ordinate for this data set is hg18 based. To use the PSEQ annotation data source which is hg19 based, you will lift over this data set to use hg19 co-ordinate. Since PSEQ does not provide a liftover feature therefore the data has already been lifted over for you using Variant Association Tools. The resulting data files, **CEU.exon.201003.genotypes.hg19.vcf.gz** and **YRI.exon.201003.genotypes.hg19.vcf.gz**, will be used for this exercise. One data set contains exome data for European-Americans (CEU) from 1000 Genomes while the other for Yoruba (YRI). The liftover feature may also have to be used with your data set as new hg coordinates become available. For additional information see <http://varianttools.sourceforge.net/Vtools/Liftover>

Phenotype Data

To demonstrate performing an association analysis, we simulated a quantitative trait phenotype (BMI). Please note that these phenotypes are **NOT** from the 1000 genomes project. The phenotype data for the

exercise can be found in the text file **phenotype.phe**. This phenotype file contains data for 202 individuals from both the CEU and YRI populations.

Computation Resources

The following tutorial uses a small data set so that the association analysis can be completed in a short period-of-time. Large next-generation sequenced data sets require a reasonably powerful machine with a high-speed internet connection.

Data Cleaning and Variant/Sample Selection

Getting Started

To get a list of PSEQ subcommands use:

```
pseq help
```

Or,

```
pseq help all
```

Create a new project

```
pseq myproj new-project --resources hg19  
Creating new project specification file [ myproj.pseq ]
```

The “--resources” flag tells **pseq** where your supporting databases are located. For this exercise the necessary databases have already been created and are within your exercise directory. Instructions on how to create these databases is located at:

<http://atgu.mgh.harvard.edu/plinkseq/resources.shtml>.

Load variant data

Import all vcf files under the current directory:

```
pseq myproj load-vcf --vcf CEU.exon.2010_03.genotypes.hg19.vcf.gz YRI.exon.2010_03.genotypes.hg19.vcf.gz  
loading : /home/gmc01/data/pseq/CEU.exon.2010_03.genotypes.hg19.vcf.gz ( 90 individuals )  
parsed 3000 rows  
loading : /home/gmc01/data/pseq/YRI.exon.2010_03.genotypes.hg19.vcf.gz ( 112 individuals )  
parsed 5000 rows  
/home/gmc01/data/pseq/CEU.exon.2010_03.genotypes.hg19.vcf.gz : inserted 3489 variants  
/home/gmc01/data/pseq/YRI.exon.2010_03.genotypes.hg19.vcf.gz : inserted 5175 variants
```

Note CEU are European-Americans and YRI are Yoruba from Nigeria.

Load phenotype data

```
pseq myproj load-pheno --file phenotype.phe  
Processed 202 rows
```

The “phenotype.phe” file contains phenotypes for SEX, BMI and RACE (BMI is body mass index, males are denoted by a 1 and females by 2). Instruction on formatting .phe file can be found at <https://atgu.mgh.harvard.edu/plinkseq/input.shtml#phe>.

View variants and samples

To view variant sites info:

```
pseq myproj v-view | head
```

```
chr1:1115461 . C/T . 1 PASS
chr1:1115503 . T/C . 1 SBFilter
chr1:1115510 . C/T . 1 PASS
chr1:1115548 . G/A . 1 PASS
chr1:1115604 . C/A . 1 PASS
chr1:1118275 rs61733845 C/T . 2 PASS
chr1:1119399 . C/T . 1 PASS
chr1:1119434 . C/A . 1 PASS
chr1:1120370 . C/G . 1 PASS
chr1:1120377 . T/A . 1 PASS
```

v-view command outputs a per-variant level view of a project, with the above fields: chromosome (base-position); variant-ID (or '.' If novel); ref/alt alleles; a sample/file identifier (or '.' If consensus variant); # of samples the variant observed in; filter values for samples (here 'PASS' means that the variant site passes all filter and 'SBFilter' means that the variant site fails to pass the strand bias (SB) filter). More details about v-view command can be found at <https://atgu.mgh.harvard.edu/plinkseq/view.shtml#var>

To view samples and phenotypes:

i-view command writes to standard output to view individuals' phenotype information

```
pseq myproj i-view | head
```

```
#BMI (Float) "BMI"
#RACE (String) "RACE"
#SEX (Integer) "SEX"
#PHE .
#STRATA .
#ID FID IID MISS SEX PAT MAT META
NA06984 . . 0 0 . . BMI=36.353;RACE=CEU;SEX=1
NA06985 . . 0 0 . . BMI=21.415;RACE=CEU;SEX=2
NA06986 . . 0 0 . . BMI=26.898;RACE=CEU;SEX=1
NA06989 . . 0 0 . . BMI=25.015;RACE=CEU;SEX=2
```

There are 3 fields, BMI, RACE and SEX contained in the input phenotype file, phenotype.phe. The headers are #ID – main unique individual ID; FID – optional family ID; IID: optional individual ID; MISS – a flag to indicate missing data; SEX – sex; PAT – paternal ID; MAT – maternal ID; META – meta information of fields from input phenotype file. More details about i-view command outputs can be found at <https://atgu.mgh.harvard.edu/plinkseq/view.shtml#ind>.

Summary

To view a summary of the complete project

```
pseq myproj summary
```

Command above will generate a long list of output. To view summaries of portions of the project, i.e., variant data, phenotype data, locus data, reference data, sequence data, input files and meta data:

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
6987 unique variants
File tag : 1 (3489 variants, 90 individuals)
File tag : 2 (5175 variants, 112 individuals)
```

pseq myproj ind-summary

-----Individual DB summary-----

202 unique individuals
Phenotype : BMI (Float) "BMI"
Phenotype : RACE (String) "RACE"
Phenotype : SEX (Integer) "SEX"

pseq myproj loc-summary

pseq myproj ref-summary

pseq myproj seq-summary

pseq myproj file-summary

pseq myproj meta-summary

More details about viewing summary information for project databases can be found at <https://atgu.mgh.harvard.edu/plinkseq/proj.shtml#summ>

Based on the “pseq myproj var-summary” command there are 6987 unique variant sites for CEU and YRI, with the CEU sample having 3489 variant sites and the YRI sample 5175 variant sites. .

For an overview of variant summary statistics:

pseq myproj v-stats

NVAR 6987
RATE 0.568384
MAC 19.8557
MAF 0.0691347
SING 2064
MONO 30
TITV 3.57264
TITV_S 3.77778
DP 8426.74
QUAL NA
PASS 0.999857
FILTER|PASS 0.999857
FILTER|SBFilter 0.000143123
PASS_S 1

v-stats command obtains summary statistics across variants. Output statistics are NVAR – total number of variants; RATE – average call rate; MAC – mean minor allele count; MAF – mean minor allele frequency; SING – number of singletons; MONO – number of monomorphic sites; TITV – transition/transversion (Ti/Tv) ratio; TITV_S – Ti/Tv ratio for singletons; DP – mean variant read depth; QUAL – mean QUAL score from VCF; PASS – proportion of variants that PASS all FILTERS; FILTER|PASS – proportion of variants that pass all filters; FILTER|SBFilter – proportion of variants that fail to pass SB filter. More details about v-stats command outputs can be found at <https://atgu.mgh.harvard.edu/plinkseq/stats.shtml#var>

For individual level summary statistics:

pseq myproj i-stats | head

| ID | NALT | NMIN | NHET | NVAR | RATE | SING | TITV | PASS | PASS_S | QUAL | DP |
|---------|------|------|------|------|----------|------|---------|------|--------|------|---------|
| NA06984 | 719 | 568 | 480 | 3162 | 0.452555 | 8 | 3.61789 | 568 | 8 | NA | 13489 |
| NA06985 | 655 | 531 | 420 | 3144 | 0.449979 | 10 | 3.5 | 531 | 10 | NA | 13530.3 |
| NA06986 | 773 | 643 | 503 | 3437 | 0.491914 | 22 | 3.69343 | 643 | 22 | NA | 12535.8 |
| NA06989 | 699 | 532 | 469 | 3130 | 0.447975 | 8 | 3.22222 | 532 | 8 | NA | 13549.7 |
| NA06994 | 591 | 464 | 377 | 3002 | 0.429655 | 3 | 3.59406 | 464 | 3 | NA | 13923.8 |
| NA07000 | 802 | 613 | 517 | 3388 | 0.484901 | 10 | 3.67939 | 613 | 10 | NA | 12292.6 |
| NA07037 | 800 | 631 | 512 | 3374 | 0.482897 | 4 | 3.60584 | 631 | 4 | NA | 12357.4 |
| NA07048 | 817 | 675 | 607 | 3373 | 0.482754 | 15 | 3.29936 | 675 | 15 | NA | 12909.5 |
| NA07051 | 825 | 637 | 507 | 3451 | 0.493917 | 13 | 3.05732 | 637 | 13 | NA | 11929 |

i-stats command obtains a matrix of summary statistics for every individual in a project. Output statistics are ID – individual ID; NALT – number of non-reference genotypes; NMIN – number of genotypes with a minor allele; NHET – number of heterozygous genotypes for individual; NVAR – total number of called

variants for individual; RATE – genotyping rate for individual; SING – number of singletons individuals has; TITV – mean Ti/Tv for variants for which individual has a nonreference genotype; PASS – number of variants passing for which individual has a nonreference genotype; PASS_S - number of singletons passing for which individual has a (singleton) nonreference genotype; QUAL - mean QUAL for variants for which individual has a nonreference genotype; DP - mean variant DP for variants for which individual has a nonreference genotype. More details about i-stats command output can be found at <https://atgu.mgh.harvard.edu/plinkseq/stats.shtml#ind>

The file tags (listed at the top of the “pseq myproj var-summary” results as “1” for the CEU imported VCF file and “2” for YRI imported VCF file) can be changed to more identifiable names using the commands:

```
pseq myproj tag-file --id 1 --name CEU
```

```
pseq myproj tag-file --id 2 --name YRI
```

To view changes use the command:

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
6987 unique variants
File tag : CEU (3489 variants, 90 individuals)
File tag : YRI (5175 variants, 112 individuals)
```

This will help us later for viewing population specific data as well as filtering and analyzing data based on population.

Variant statistics

Variant statistics such as Hardy-Weinberg equilibrium, minor allele count, and minor allele frequency can be output using the “v-freq” command:

```
pseq myproj v-freq | head
```

| VAR | CHR | POS | REF | ALT | FILTER | QUAL | TI | GENO | MAC | MAF | REFMIN | HWE | HET | NSNP |
|--------------|-----|---------|-----|-----|----------|------|----|----------|-----|------------|--------|----------|-----------|------|
| chr1:1115461 | 1 | 1115461 | C | T | PASS | . | 1 | 0.311881 | 4 | 0.031746 | 0 | 1 | 0.0634921 | 3 |
| chr1:1115503 | 1 | 1115503 | T | C | SBFilter | . | 1 | 0.282178 | 4 | 0.0350877 | 0 | 1 | 0.0701754 | 2 |
| chr1:1115510 | 1 | 1115510 | C | T | PASS | . | 1 | 0.331683 | 2 | 0.0149254 | 0 | 1 | 0.0298507 | 2 |
| chr1:1115548 | 1 | 1115548 | G | A | PASS | . | 1 | 0.262376 | 1 | 0.00943396 | 0 | 1 | 0.0188679 | 1 |
| chr1:1115604 | 1 | 1115604 | C | A | PASS | . | 0 | 0.287129 | 3 | 0.0258621 | 0 | 1 | 0.0517241 | 0 |
| chr1:1118275 | 1 | 1118275 | C | T | PASS | . | 1 | 0.579208 | 45 | 0.192308 | 0 | 0.367544 | 0.282851 | 0 |
| chr1:1119399 | 1 | 1119399 | C | T | PASS | . | 1 | 0.49505 | 3 | 0.015 | 0 | 1 | 0.03 | 1 |
| chr1:1119434 | 1 | 1119434 | C | A | PASS | . | 0 | 0.49505 | 1 | 0.005 | 0 | 1 | 0.01 | 0 |
| chr1:1120370 | 1 | 1120370 | C | G | PASS | . | 0 | 0.49505 | 16 | 0.08 | 0 | 0.478564 | 0.14 | 2 |

Please note that it is not valid to filter for deviation from HWE using the entire project since there are two populations, instead the HWE much be examined for each individual project.

For population specific variant statistics use the “--mask” flag with the “file” option:

```
pseq myproj v-freq --mask file=CEU | head
```

| VAR | CHR | POS | REF | ALT | FILTER | QUAL | TI | GENO | MAC | MAF | REFMIN | HWE | HET | NSNP |
|--------------|-----|---------|-----|-----|----------|------|----|----------|-----|------------|--------|-----|-----------|------|
| chr1:1115503 | 1 | 1115503 | T | C | SBFilter | 0 | 1 | 0.633333 | 4 | 0.0350877 | 0 | 1 | 0.0701754 | 1 |
| chr1:1115548 | 1 | 1115548 | G | A | PASS | 0 | 1 | 0.588889 | 1 | 0.00943396 | 0 | 1 | 0.0188679 | 0 |
| chr1:1118275 | 1 | 1118275 | C | T | PASS | 0 | 1 | 0.677778 | 3 | 0.0245902 | 0 | 1 | 0.0491803 | 0 |
| chr1:1120377 | 1 | 1120377 | T | A | PASS | 0 | 0 | 0.988889 | 1 | 0.00561798 | 0 | 1 | 0.011236 | 1 |
| chr1:1120431 | 1 | 1120431 | G | A | PASS | 0 | 1 | 0.855556 | 6 | 0.038961 | 0 | 1 | 0.0779221 | 0 |
| chr1:3548136 | 1 | 3548136 | T | C | PASS | 0 | 1 | 0.811111 | 18 | 0.123288 | 1 | 1 | 0.219178 | 0 |
| chr1:3548832 | 1 | 3548832 | G | C | PASS | 0 | 0 | 0.988889 | 13 | 0.0730337 | 0 | 1 | 0.146067 | 0 |
| chr1:3551737 | 1 | 3551737 | C | T | PASS | 0 | 1 | 0.988889 | 1 | 0.00561798 | 0 | 1 | 0.011236 | 1 |
| chr1:3551792 | 1 | 3551792 | G | A | PASS | 0 | 1 | 1 | 8 | 0.0444444 | 0 | 1 | 0.0888889 | 0 |

```
pseq myproj v-freq --mask file=YRI | head
```

| VAR | CHR | POS | REF | ALT | FILTER | QUAL | TI | GENO | MAC | MAF | REFMIN | HWE | HET | NSNP |
|--------------|-----|---------|-----|-----|--------|------|----|----------|-----|-----------|--------|----------|-----------|------|
| chr1:1115461 | 1 | 1115461 | C | T | PASS | 0 | 1 | 0.5625 | 4 | 0.031746 | 0 | 1 | 0.0634921 | 1 |
| chr1:1115510 | 1 | 1115510 | C | T | PASS | 0 | 1 | 0.598214 | 2 | 0.0149254 | 0 | 1 | 0.0298507 | 1 |
| chr1:1115604 | 1 | 1115604 | C | A | PASS | 0 | 0 | 0.517857 | 3 | 0.0258621 | 0 | 1 | 0.0517241 | 0 |
| chr1:1118275 | 1 | 1118275 | C | T | PASS | 0 | 1 | 0.5 | 42 | 0.375 | 0 | 0.395585 | 0.535714 | 0 |
| chr1:1119399 | 1 | 1119399 | C | T | PASS | 0 | 1 | 0.892857 | 3 | 0.015 | 0 | 1 | 0.03 | 1 |
| chr1:1119434 | 1 | 1119434 | C | A | PASS | 0 | 0 | 0.892857 | 1 | 0.005 | 0 | 1 | 0.01 | 0 |
| chr1:1120370 | 1 | 1120370 | C | G | PASS | 0 | 0 | 0.892857 | 16 | 0.08 | 0 | 0.478564 | 0.14 | 1 |
| chr1:1120431 | 1 | 1120431 | G | A | PASS | 0 | 1 | 0.741071 | 67 | 0.403614 | 0 | 0.360868 | 0.542169 | 4 |
| chr1:1120488 | 1 | 1120488 | A | C | PASS | 0 | 0 | 0.857143 | 10 | 0.0520833 | 0 | 1 | 0.104167 | 3 |

As you see, the “--mask” flag is used to set conditions for the viewing or filtering variants or individuals.

More details about “v-freq” command can be found at

<https://atgu.mgh.harvard.edu/plinkseq/tutorial.shtml>

Data Cleaning

Removal of low quality variants

To view the number of variants that passed all quality filters:

```
pseq myproj v-view --mask any.filter.ex | head
```

```
chr1:1115461      .          C/T      .          1          PASS
chr1:1115510      .          C/T      .          1          PASS
chr1:1115548      .          G/A      .          1          PASS
chr1:1115604      .          C/A      .          1          PASS
chr1:1118275      rs61733845 C/T      .          2          PASS
chr1:1119399      .          C/T      .          1          PASS
chr1:1119434      .          C/A      .          1          PASS
chr1:1120370      .          C/G      .          1          PASS
chr1:1120377      .          T/A      .          1          PASS
chr1:1120431      rs1320571  G/A      .          2          PASS
```

```
pseq myproj v-view --mask any.filter.ex | wc -l
```

There are 6986 unique variant sites that have passed the quality filters. The “--mask” flag gives the condition(s) that must be met for the variant to be listed. Here “any.filter.ex” tells **pseq** to remove any variants that failed 1 or more quality filters. Only variants that have a ‘PASS’ value in the FILTER field of the vcf file will be selected. More details about filtering variants on FILTER field can be found at <https://atgu.mgh.harvard.edu/plinkseq/masks.shtml#filter>

To view the number of variants that failed any quality filter:

```
pseq myproj v-view --mask any.filter | wc -l
```

One variant failed the filter. To select only variants that passed all quality filters:

```
pseq myproj var-set --group pass --mask any.filter.ex
```

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
6987 unique variants
File tag : CEU (3489 variants, 90 individuals)
File tag : YRI (5175 variants, 112 individuals)
```

```
Set pass containing 8663 variants
```

The “var-set” option tells **pseq** that we will be creating a new set of variants, the input following the “--group” flag gives the name of the new variant set, and the input following the “--mask” flag gives the condition(s) that must be met for the variant to be included in the new variant set.

If we consider variant sites with a read depth < 15 as low quality variant sites and we want to remove variants that did not meet this threshold. Note that ‘DP’, which denotes total read depth of a variant site, is contained in the INFO field of vcf file.

```
pseq myproj var-set --group pass_DP15 --mask include="DP>14" var=pass
```

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
6987 unique variants
File tag : CEU (3489 variants, 90 individuals)
File tag : YRI (5175 variants, 112 individuals)
```

```
Set pass containing 8663 variants
Set pass_DP15 containing 8662 variants
```

Only one variant site is removed. The “var=allpass” option allows us to use a previously defined variant set as a reference for additional filtering of a previously filtered variant set. By using various “--mask” commands you can filter out variants that are not useful for your particular study.

Filter data by genotype read depth 10

```
pseq myproj var-set --group pass_DP15_DPgeno10 --mask geno=DP:ge:11 var=pass_DP15
```

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
6987 unique variants
File tag : CEU (3489 variants, 90 individuals)
File tag : YRI (5175 variants, 112 individuals)
```

```
Set pass containing 8663 variants
Set pass_DP15 containing 8662 variants
Set pass_DP15_DPgeno10 containing 8662 variants
```

This command sets all genotypes with a sequencing depth (DP) < 11 to null using the option “geno=DP:ge:11”. In the vcf file, genotype level DP information is contained in the genotype columns, present under each individual ID and is specific to every individual’s genotype. Available genotype level information is denoted by FORMAT column in the vcf file.

Association Tests for a Quantitative Trait

NOTE: From this step forward the association tests will be performed for the CEU population only. The “file=YRI” tag can be used to perform the same tests on the YRI data.

Select CEU variant sites

```
pseq myproj var-set --group pass_DP15_DPgeno10_CEU --mask file=CEU var=pass_DP15_DPgeno10
```

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
...
Set pass_DP15_DPgeno10_CEU containing 3488 variants
```

There are 3488 variant sites that can be found in CEU population dataset after QC.

Exclude variant sites with HWE p-value < 5.7e-7

```
pseq myproj var-set --group pass_DP15_DPgeno10_CEU_HWE --mask hwe=5.7e-7:1 var=pass_DP15_DPgeno10_CEU
```

```
pseq myproj var-summary
```

```
-----Variant DB summary-----
```

```
...
Set pass_DP15_DPgeno10_CEU containing 3479 variants
```

There are 3479 variant sites that are in HWE (Hardy-Weinberg equilibrium) in CEU population. Details about tests for deviation from HWE can be found at http://en.wikipedia.org/wiki/Hardy-Weinberg_principle. Here we use a p-value cutoff of 5.7e-7 to exclude variant sites, for more details see reference <http://www.nature.com/nature/journal/v447/n7145/full/nature05911.html>

Filter variants by minor allele frequency (MAF)

We wish to analyze variant sites with different allele frequencies. In order to obtain the different data sets the following commands are used.

Advanced Gene Mapping Course: Mendelian Randomization Exercise

Andrew DeWan, PhD, MPH

This exercise is designed to give you practical experience conducting a two-sample Mendelian randomization study using the online version of MR-base (<https://www.mrbase.org/>).

Part I:

You will be conducting an analysis to investigate the causal relationship between low density lipoprotein (LDL) and coronary heart disease (CHD) based on summary statistics from previously published GWAS data.

Exposure: Fasting LDL measurements from in 173,082 subjects and 2,437,752 genetic variants. Subjects are of European, East and South Asian and African ancestry.

Willer CJ, Schmidt EM, Sengupta S, Peloso GM, Gustafsson S, Kanoni S, Ganna A, et al. Discovery and refinement of loci associated with lipid levels. *Nat Genet.* 2013 Nov;45(11):1274-1283. doi: 10.1038/ng.2797. Epub 2013 Oct 6. PMID: [24097068](#); PMID: PMC3838666.

Outcome: CHD (e.g. myocardial infarction (MI), acute coronary syndrome, chronic stable angina, or coronary stenosis >50%) in 184,305 subjects (60,801 cases and 123,504 controls) and 9,455,779 genetic variants. Subjects are of European, East and South Asian, Hispanic and African ancestry.

Nikpay M, Goel A, Won HH, Hall LM, Willenborg C, Kanoni S, Saleheen D et al. A comprehensive 1,000 Genomes-based genome-wide association meta-analysis of coronary artery disease. *Nat Genet.* 2015 Oct;47(10):1121-1130. doi: 10.1038/ng.3396. Epub 2015 Sep 7. PMID: [26343387](#); PMID: PMC4589895.

- 1) Conduct an MR analysis of LDL and CAD. Studies can be search by PubmedID in MR-base (make sure PubmedID is checked), however, please note the following:
 - A. For the exposure for this publication, use the larger set of subjects for this first analysis (N=173,082)
 - B. For the exposure, use a p-value threshold of $5e10^{-8}$, LD $R_{sq} = 0.001$ and clumping distance of 1000kb. Also make sure "Perform Clumping" is checked.
 - C. For the outcome for this publication, use the trait denoted "Coronary heart disease"
 - D. When running the MR analysis you will want to allow LD proxies to be selected for the outcome using a minimum R_{sq} of 0.8 and also allow for

6. How would you interpret the results of the three analyses together (i.e. IVW, MR Egger and Weighted Median)?

2) Re-run the analysis but for myocardial infarction (MI) using outcome data from the same publication.

Questions:

1. Is there evidence of an association between LDL and MI?

2. Can the association between LDL and CHD be explained by MI?

3) Feel free to explore associations with additional exposures such as HDL, BMI (you can use the Yengo et al. SNPs) or other exposures/outcomes of interest to you.

Part II:

Let's now see if we can validate the finding of an association between LDL and CHD by using different exposure data source and potentially dissect this signal to see if we pinpoint the features of LDL that might be driving this signal. We will use metabolomics data that was generated in a sample of 24,925 individuals.

Kettunen, J., Demirkan, A., Würtz, P. *et al.* Genome-wide study for circulating metabolites identifies 62 loci and reveals novel systemic effects of *LPA*. *Nat Commun* **7**, 11122 (2016). <https://doi.org/10.1038/ncomms11122>.

Exposures: LDL.C, LDL.D, S.LDL.C, S.LDL.L, S.LDL.P, M.LDL.C, M.LDL.CE, M.LDL.L, M.LDL.P, M.LDL.PL, L.LDL.C, L.LDL.CE, L.LDL.FC, L.LDL.L, L.LDL.P, L.LDL.PL (16 metabolites)

Where S. = small, M. = medium and L. = large; .C = total cholesterol, .D = diameter, .L = total lipids, .P = concentration, .CE = cholesterol esters, .PL = phospholipids

These can all be selected when you are on the "Choose Exposure" screen and selecting the "Metabolite level QTLs". You can then type in "LDL" in the analyte window and select each of these in the window that pops up. The most efficient way is to select all of the metabolites you're interested in and then run the MR analyses together. Before clicking off of this screen you will want to click on the "Select All" under Row Selection. This will allow you to run the analysis on all of the SNPs for each metabolite.

Use the same CHD outcome as you did for Part I (Nikpay PMID: [26343387](#)) using the full set of cases and controls (N=184,305).

- 1) Conduct an MR analysis as you did previously.

Questions:

1. For LDL.C, does the association between LDL and CHD validate the previous findings?
2. Considering all of the associations, do these results differentiate between the different characteristics of LDL (Please note: you will want to take into account the 16 association tests)?

3. What might be one explanation for the similarity between results for the different LDL characteristics?

4. Are there any concerns about heterogeneity or pleiotropy?

Let's now look at the associations with VLDL metabolites using the same exposure and outcome data sources.

Exposures: 33 VLDL metabolites (Please note additional abbreviations: XS. = very small, XL. = very large, XXL. = extremely large; .TG = triglycerides)

2) Conduct an MR analysis as you did previously.

Question:

5. Considering all of the associations, are there any obvious trends in the results (Please note: you will want to take into account the 33 association tests)?

Advanced Gene Mapping Course: Mendelian Randomization Exercise - Answers

Andrew DeWan, PhD, MPH

This exercise is designed to give you practical experience conducting a two-sample Mendelian randomization study using the online version of MR-base (<https://www.mrbase.org/>).

Part I:

You will be conducting an analysis to investigate the causal relationship between low density lipoprotein (LDL) and coronary heart disease (CHD) based on summary statistics from previously published GWAS data.

Exposure: Fasting LDL measurements from in 173,082 subjects and 2,437,752 genetic variants. Subjects are of European, East and South Asian and African ancestry.

Willer CJ, Schmidt EM, Sengupta S, Peloso GM, Gustafsson S, Kanoni S, Ganna A, et al. Discovery and refinement of loci associated with lipid levels. *Nat Genet.* 2013 Nov;45(11):1274-1283. doi: 10.1038/ng.2797. Epub 2013 Oct 6. PMID: [24097068](#); PMCID: PMC3838666.

Outcome: CHD (e.g. myocardial infarction (MI), acute coronary syndrome, chronic stable angina, or coronary stenosis >50%) in 184,305 subjects (60,801 cases and 123,504 controls) and 9,455,779 genetic variants. Subjects are of European, East and South Asian, Hispanic and African ancestry.

Nikpay M, Goel A, Won HH, Hall LM, Willenborg C, Kanoni S, Saleheen D et al. A comprehensive 1,000 Genomes-based genome-wide association meta-analysis of coronary artery disease. *Nat Genet.* 2015 Oct;47(10):1121-1130. doi: 10.1038/ng.3396. Epub 2015 Sep 7. PMID: [26343387](#); PMCID: PMC4589895.

- 1) Conduct an MR analysis of LDL and CAD. Studies can be search by PubmedID in MR-base (make sure PubmedID is checked), however, please note the following:
 - A. For the exposure for this publication, use the larger set of subjects for this first analysis (N=173,082)
 - B. For the exposure, use a p-value threshold of $5e10^{-8}$, LD $R_{sq} = 0.001$ and clumping distance of 1000kb. Also make sure "Perform Clumping" is checked.
 - C. For the outcome for this publication, use the trait denoted "Coronary heart disease"
 - D. When running the MR analysis you will want to allow LD proxies to be selected for the outcome using a minimum R_{sq} of 0.8 and also allow for

palindromic SNPs with a MAF threshold of 0.3. Make sure you set “Allele harmonization” to “Attempt to align strands for palindromic SNPs”

- E. Select the following methods:
- Inverse variance weighted (NOTE: this is a random effects model)
 - MR Egger
 - Weighted Median

Questions:

1. How many variants are included in your genetic instrument for the exposure and how many are included in the outcome analysis? Of these, how many are proxies?

There are 79 variants that surpass the $p < 5e-8$ threshold for LDL in the exposure GWAS. Of these 77 are identified in the CHD outcome GWAS, 1 of which is a proxy.

2. Based on the descriptions above, is the study used to define the IV appropriate for the outcome population?

They are fairly well matched in terms of the population ancestries in the two studies, however, the outcome GWAS has subjects of Hispanic ancestry which could be a minor issue. This would be something to mention in the Discussion section of a manuscript. There is a subset of only European subjects for LDL but not for CHD, however, if you had access to the original data you could subset the subjects by ancestry to better match the exposure and outcome groups.

3. Is there evidence of an association between LDL and CHD?

Yes, the IVW yields a beta = 0.4114 ($p = 1.626e-15$) which corresponds to an OR of 1.51 (95% CI: 1.36 – 1.67) per SD increase in LDL.

4. Is there evidence of heterogeneity in the genetic effects?

Yes, there is significant heterogeneity across effects of each SNP on CHD ($p = 2.822e-40$) indicating that the random effects model is appropriate.

5. Is there evidence of pleiotropy?

From the MR-Egger regression there is no significant evidence of pleiotropy as the regression intercept is not significantly different from zero ($p = 0.118$).

6. How would you interpret the results of the three analyses together (i.e. IVW, MR Egger and Weighted Median)?

The IVW method (OR = 1.51, 95% CI: 1.36 – 1.67, $p=1.626e-16$), MR-Egger (OR = 1.66, 95% CI: 1.42 – 1.93, $p=1.086e-8$) and Weighted median (OR = 1.49, 95% CI: 1.36 – 1.63, $p=1.962e-19$) are relatively consistent meaning the causal effect estimate is likely to be between 1.49 and 1.66. There is no evidence that this estimate is influenced by horizontal pleiotropy as the MR-Egger intercept is not significant.

- 2) Re-run the analysis but for myocardial infarction (MI) using outcome data from the same publication.

Questions:

1. Is there evidence of an association between LDL and MI?

Yes, IVW method provides significant evidence of an association between LDL and MI (OR = 1.48, 95% CI: 1.33 – 1.66, $p=1.42e-12$). The other MR measures of association are consistent with this estimate and there is again no evidence of horizontal pleiotropy.

2. Can the association between LDL and CHD be explained by MI?

We would need to test the other traits included in the CHD definition to see if they were associated with LDL or not and test for heterogeneity of the effects. However it is reassuring that the effect estimates are consistent between the larger CHD group and the smaller subgroup of subjects with MI.

- 3) Feel free to explore associations with additional exposures such as HDL, BMI (you can use the Yengo et al. SNPs) or other exposures/outcomes of interest to you.

I'm more than happy to discuss additional results one-on-one or during the time we discuss the answers to this exercise.

Part II:

Let's now see if we can validate the finding of an association between LDL and CHD by using different exposure data source and potentially dissect this signal to see if we pinpoint the features of LDL that might be driving this signal. We will use metabolomics data that was generated in a sample of 24,925 individuals.

Kettunen, J., Demirkan, A., Würtz, P. *et al.* Genome-wide study for circulating metabolites identifies 62 loci and reveals novel systemic effects of *LPA*. *Nat Commun* 7, 11122 (2016). <https://doi.org/10.1038/ncomms11122>.

Exposures: LDL.C, LDL.D, S.LDL.C, S.LDL.L, S.LDL.P, M.LDL.C, M.LDL.CE, M.LDL.L, M.LDL.P, M.LDL.PL, L.LDL.C, L.LDL.CE, L.LDL.FC, L.LDL.L, L.LDL.P, L.LDL.PL (16 metabolites)

Where S. = small, M. = medium and L. = large; .C = total cholesterol, .D = diameter, .L = total lipids, .P = concentration, .CE = cholesterol esters, .PL = phospholipids

These can all be selected when you are on the "Choose Exposure" screen and selecting the "Metabolite level QTLs". You can then type in "LDL" in the analyte window and select each of these in the window that pops up. The most efficient way is to select all of the metabolites you're interested in and then run the MR analyses together. Before clicking off of this screen you will want to click on the "Select All" under Row Selection. This will allow you to run the analysis on all of the SNPs for each metabolite.

Use the same CHD outcome as you did for Part I (Nikpay PMID: [26343387](#)) using the full set of cases and controls (N=184,305).

- 1) Conduct an MR analysis as you did previously.

Questions:

1. For LDL.C, does the association between LDL and CHD validate the previous findings?

Yes, although the magnitude of the effect is slightly attenuated. The IVW yields a beta = 0.3665 (p=1.19e-07) which corresponds to an OR of 1.44. The previous OR estimate was 1.51.

2. Considering all of the associations, do these results differentiate between the different characteristics of LDL (Please note: you will want to take into account the 16 association tests)?

All of the associations yield statistically significant results, except for LDL diameter which doesn't meet the corrected significance threshold.

The remaining metabolites have statistically significant betas ranging from 0.3665 (LDL diameter) to 0.4709 (concentration of small LDL)

3. What might be one explanation for the similarity between results for the different LDL characteristics?

There is a high degree of overlap between the variants contained in the instrument variable for each of the metabolites.

4. Are there any concerns about heterogeneity or pleiotropy?

There is significant heterogeneity across all of the metabolites, except for the diameter of LDL. There is some evidence of pleiotropy among the large LDL metabolites, but not among any of the others.

Let's now look at the associations with VLDL metabolites using the same exposure and outcome data sources.

Exposures: 33 VLDL metabolites (Please note additional abbreviations: XS. = very small, XL. = very large, XXL. = extremely large; .TG = triglycerides)

2) Conduct an MR analysis as you did previously.

Question:

5. Considering all of the associations, are there any obvious trends in the results (Please note: you will want to take into account the 33 association tests)?

The results tend to be significant ($p < 0.00015$) for the small and very small VLDL metabolites (except for the small VLDL triglycerides, $p = 0.00019$), whereas the large, very large and extremely large LDL metabolites are non-significant (except for the cholesterol esters in large VLDL).

Advanced Gene Mapping Course: Pleiotropy Exercise

Andrew DeWan, PhD, MPH

This exercise was designed to give you practical experience identifying cross phenotype associations using both univariate and multivariate methods and then dissecting these cross-phenotype associations to determine if they show evidence of biological and/or mediated pleiotropy.

A population-based dataset with 3000 subjects and two quantitative traits (Trait 1 and Trait 2) along with 2000 SNPs on one chromosome were simulated. Let's assume that Trait 1 was measured 20 years prior to Trait 2 (i.e. Trait 1 will act as the mediator in our mediation analysis). The two quantitative traits are correlated and there are markers associated with one or both phenotypes as well as unassociated.

The dataset has been QC'd. The files for the initial analyses are:

pleiotropy_exercise.bed, .bim, .fam and pleiotropy_exercise_phenotypes.txt

I have included a summary table that you will want to fill out as you are working through this exercise. This will help keep track of the SNPs you select for the mediation analysis as well as the interpretation of the results at the end of the exercise

Univariate analyses

- a. Conduct a univariate analysis (using `--linear`) in PLINK for both datasets and both traits
Note: You will need to use the `--pheno/--pheno-name` commands to specify the phenotype file and phenotype name.

```
plink\  
--bfile pleiotropy_exercise\  
--pheno pleiotropy_exercise_phenotypes.txt\  
--pheno-name Trait1\  
--sex\  
--linear\  
--out Trait1
```

For use in several downstream steps, let's create files with only the header and SNP results for each of the univariate analyses:

```
grep 'TEST' Trait1.assoc.linear > Trait1_snp.assoc.linear  
grep 'ADD' Trait1.assoc.linear >> Trait1_snp.assoc.linear  
grep 'TEST' Trait2.assoc.linear > Trait2_snp.assoc.linear  
grep 'ADD' Trait2.assoc.linear >> Trait2_snp.assoc.linear
```

- b. Try visualizing the data by creating a Hudson plot in R. This will give you some sense of the overlapping signals between the two association analyses.

```
library(hudson)
dat1<-read.table("Trait1_snp.assoc.linear",header=T)
dat2<-read.table("Trait2_snp.assoc.linear",header=T)
names(dat1_snp)<-c("CHR", "SNP", "POS", "A1", "TEST", "NMISS", "BETA",
+"STAT", "pvalue")
names(dat2_snp)<-(names(dat1_snp))
gmirror(top=dat1_snp, bottom=dat2_snp, tline=5e-08, bline=5e-08,

+ toptitle="Trait1", bottomtitle = "Trait2",
+ highlight_p = c(0.00000005,0.00000005), highlighter="green",
+ file = 'pleiotropy_hudson', res = 300, type = 'pdf')
```

- c. Now Identify genome-wide significant SNPs ($p < 5 \times 10^{-8}$) that overlap for both traits. This can be done using some simple R code:

```
Trait1 <- read.table("Trait1_snp.assoc.linear", header = T)
Trait2 <- read.table("Trait2_snp.assoc.linear", header = T)
SigTrait1 <- subset(Trait1, P<0.00000005)
SigTrait2 <- subset(Trait2, P<0.00000005)
intersect(SigTrait1$SNP, SigTrait2$SNP)
```

- d. As you can see, there are some genome-wide significant SNPs that are adjacent or close to each other. To explore whether or not these are independent associations, let's perform some simple LD clumping. You will want to carry through the index SNP identified for each clumped region. You will also want to carry through any SNPs from 1c above that were not part of a clumped region. plink\

```
--bfile pleiotropy_exercise\
--clump Trait1_snp.assoc.linear,Trait2_snp.assoc.linear\
--clump-kb 250\
--clump-p1 5e-8\
--clump-p2 5e-8\
--clump-r2 0.2\
--clump-replicate\
--clump-verbose\
--out Trait1_Trait2_clump
```

Multivariate analysis

- a. Before moving on to dissecting the cross-phenotype associations, let's see if we can include a few additional SNPs/regions to explore by using multivariate analysis. But let's only consider additional regions that are genome-wide suggestive for both phenotypes.

First run a multivariate analysis on Traits 1 and 2.

```
plink.multivariate\  
--noweb\  
--bfile pleiotropy_exercise\  
--mult-pheno pleiotropy_exercise_phenotypes.txt\  
--sex\  
--mqfam\  
--out Trait1_Trait2
```

Please note: You should use the --noweb flag due to this program being built on an old version of PLINK.

- b. Now let's identify the intersection of SNPs that are genome-wide significant in the multivariate analysis and at least suggestive for each trait in the univariate analysis, i.e. we want to make sure that both traits are contributing to the multivariate signal.

```
Trait1<-read.table("Trait1_snp.assoc.linear", header=T)  
Trait2<-read.table("Trait2_snp.assoc.linear", header=T)  
multi<-read.table("Trait1_Trait2.mqfam.total", header=T)  
sigMulti<-subset(multi, P<0.00000005)  
suggTrait1<-subset(Trait1, P<0.000005)  
suggTrait2<-subset(Trait2, P<0.000005)  
Reduce(intersect, list(suggTrait1$SNP, suggTrait2$SNP, sigMulti$SNP))
```

Select the additional SNPs that are identified from the intersection of the multivariate analysis and genome-wide suggestive lists for both traits that were not in your original list.

- c. You may want to re-run the LD clumping with a suggestive threshold to see if these additional SNPs clump with your existing clumps or are new potential regions to explore.

```
plink\  
--bfile pleiotropy_exercise\  
--clump Trait1_snp.assoc.linear,Trait2_snp.assoc.linear\  

```

```
--clump-p1 0.000005\  
--clump-p2 0.000005\  
--clump-r2 0.2\  
--clump-replicate\  
--clump-verbose\  
--out Trait1_Trait2_clump_suggestive
```

Mediation analyses

- a. For each SNPs that you have identified as a cross phenotype association (evidence of overlapping association signals as well as incorporating results from LD clumping and multivariate association) you will need to extract this data from the original plink files and create a genotype file that is coded as 0|1|2 for the genotypes. This can be done in PLINK using the `--recodeA` command and the `--extract` command by providing a file with the list of snps. This will give you a `.raw` genotype file with only the snps that you will be using in the mediation analysis.
- b. Conduct a mediation analysis in R using the *mediation* R library. Sample code for this is below (Note: replace `<SNP>` with the variable name for the SNP you are investigating. You will need to repeat this for each SNP that you have selected):

```
library(mediation)  
genotypes <- read.table("snps_for_mediation.raw", header=T)  
phenotypes<-read.table("pleiotropy_exercise_phenotypes.txt", header=T)  
combined<-merge(genotypes,phenotypes)  
med.fit<-lm(Trait1~rs125_0, data=combined)  
out.fit<-lm(Trait2~Trait1+rs125_0, data=combined)  
med.out<-mediate(med.fit,out.fit,treat="rs125_0", mediator="Trait1", boot=TRUE,  
+boot.ci.type="bca", sims=1000)  
summary(med.out)
```

This will print out a summary of the mediation analysis.

Please note: The more simulations (sims) you specify in the `med.out` step the more the CI and p-value estimates will be, however, this can also be time-consuming. If this step is taking a substantial amount of time (>20 minutes) you may want to reduce the number of simulations for the purposes of completing the exercise.

Pleiotropy Exercise - Answers

Andrew DeWan, PhD, MPH

This exercise was designed to give you practical experience identifying cross phenotype associations using both univariate and multivariate methods and then dissecting these cross phenotype associations to determine if they show evidence of biological and/or mediated pleiotropy.

A population-based dataset with 3000 subjects and two quantitative traits (Trait 1 and Trait 2) along with 2000 SNPs on one chromosome were simulated. Let's assume that Trait 1 was measured 20 years prior to Trait 2 (i.e. Trait 1 will act as the mediator in our mediation analysis). The two quantitative traits are correlated and there are markers associated with one or both phenotypes as well as unassociated.

The dataset has been QC'd. The files for the initial analyses are:

pleiotropy_exercise.bed, .bim, .fam and pleiotropy_exercise_phenotypes.txt

I have included a summary table that you will want to fill out as you are working through this exercise. This will help keep track of the SNPs you select for the mediation analysis as well as the interpretation of the results at the end of the exercise.

Univariate analyses

- a. Conduct a univariate analysis (using `--linear`) in PLINK for both datasets and both traits

Note: You will need to use the `--pheno/--pheno-name` commands to specify the phenotype file and phenotype name.

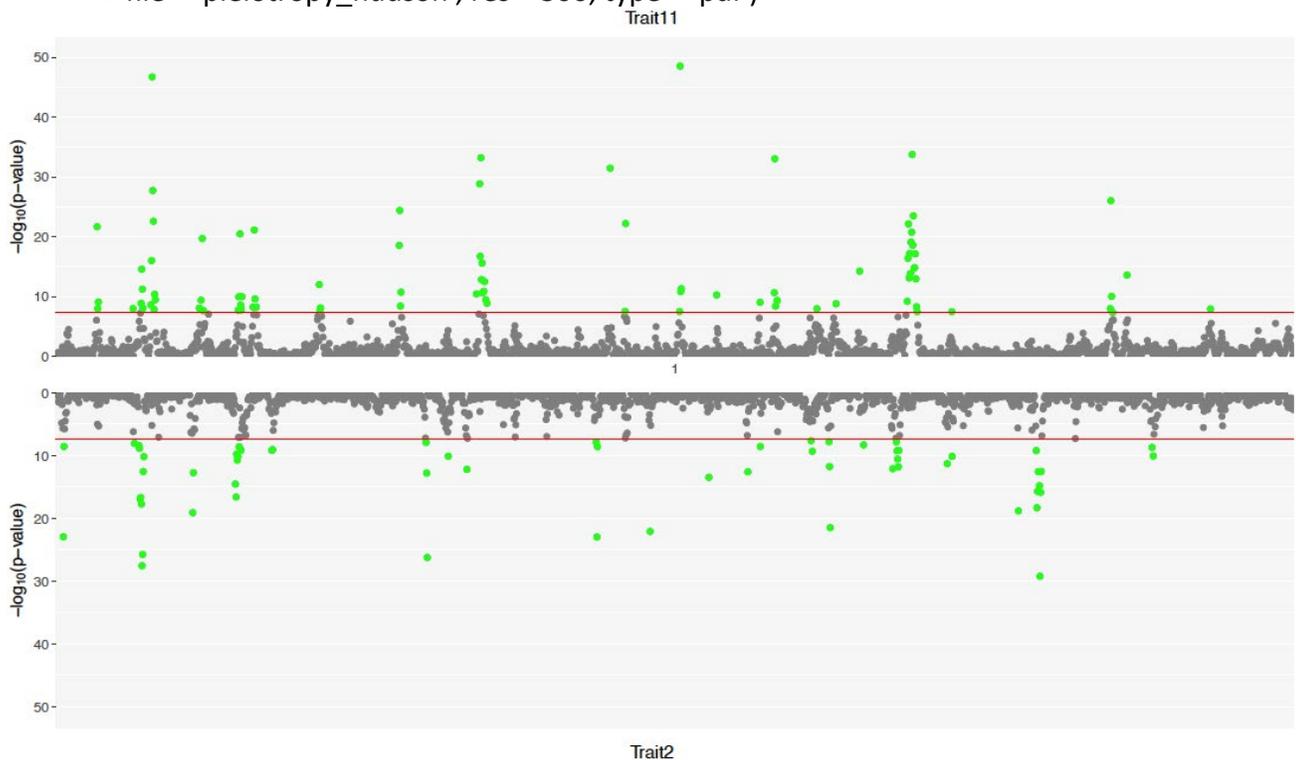
```
plink\  
--bfile pleiotropy_exercise\  
--pheno pleiotropy_exercise_phenotypes.txt\  
--pheno-name Trait1\  
--sex\  
--linear\  
--out Trait1
```

For use in several downstream steps, let's create files with only the header and SNP results for each of the univariate analyses:

```
grep 'TEST' Trait1.assoc.linear > Trait1_snp.assoc.linear  
grep 'ADD' Trait1.assoc.linear >> Trait1_snp.assoc.linear  
grep 'TEST' Trait2.assoc.linear > Trait2_snp.assoc.linear  
grep 'ADD' Trait2.assoc.linear >> Trait2_snp.assoc.linear
```

- b. Try visualizing the data by creating a Hudson plot in R. This will give you some sense of the overlapping signals between the two association analyses.

```
devtools::install_github('anastasia-lucas/hudson')
library(hudson)
dat1<-read.table("Trait1_snp.assoc.linear",header=T)
dat2<-read.table("Trait2_snp.assoc.linear",header=T)
names(dat1_snp)<-c("CHR", "SNP", "POS", "A1", "TEST", "NMISS", "BETA",
+"STAT", "pvalue")
names(dat2_snp)<-(names(dat1_snp))
gmirror(top=dat1_snp, bottom=dat2_snp, tline=5e-08, bline=5e-08,
+ toptitle="Trait11", bottomtitle = "Trait2",
+ highlight_p = c(0.00000005,0.00000005), highlighter="green",
+ file = 'pleiotropy_hudson', res = 300, type = 'pdf')
```



- c. Now Identify genome-wide significant SNPs ($p < 5 \times 10^{-8}$) that overlap for both traits. This can be done using some simple R code:

```
Trait1 <- read.table("Trait1_snp.assoc.linear", header = T)
Trait2 <- read.table("Trait2_snp.assoc.linear", header = T)
SigTrait1 <- subset(Trait1, P<0.00000005)
SigTrait2 <- subset(Trait2, P<0.00000005)
intersect(SigTrait1$SNP, SigTrait2$SNP)
```

The overlapping genome-wide significant variants are: rs138, rs139, rs140, rs141, rs296, rs299, rs1138, rs1448

- d. As you can see, there are some genome-wide significant SNPs that are adjacent or close to each other. To explore whether or not these are independent associations, let's perform some simple LD clumping. You will want to carry through the index SNP identified for each clumped region. You will also want to carry through any SNPs from 1c above that were not part of a clumped region.

```
plink\  
--bfile pleiotropy_exercise\  
--clump Trait1_snp.assoc.linear,Trait2_snp.assoc.linear\  
--clump-kb 250\  
--clump-p1 5e-8\  
--clump-p2 5e-8\  
--clump-r2 0.2\  
--clump-replicate\  
--clump-verbose\  
--out Trait1_Trait2_clump
```

OUTPUT:

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 | |
|------|-------|-----|-------|--------|----------|------|-----|-----|---|
| S001 | S0001 | | | | | | | | |
| 0 | 1 | 2 | rs139 | 139000 | 2.86e-28 | 9 | 0 | 0 | 0 |
| | 0 | 9 | | | | | | | |

| (INDEX) | rs139 | KB | RSQ | ALLELES | F | P |
|---------|-------|----|-------|---------|---|----------|
| | rs139 | 0 | 1.000 | 0 | 2 | 2.86e-28 |
| | rs137 | -2 | 0.247 | 00/00 | 2 | 2.17e-17 |
| | rs138 | -1 | 0.399 | 00/00 | 1 | 1.34e-09 |
| | rs138 | -1 | 0.399 | 00/00 | 2 | 1.91e-18 |
| | rs139 | 0 | 1 | 00/00 | 1 | 2.77e-15 |
| | rs140 | 1 | 0.229 | 00/00 | 1 | 6.05e-12 |
| | rs140 | 1 | 0.229 | 00/00 | 2 | 1.85e-26 |
| | rs141 | 2 | 0.235 | 00/00 | 1 | 9.9e-09 |
| | rs141 | 2 | 0.235 | 00/00 | 2 | 2.98e-13 |

RANGE: chr1:137000..141000
SPAN: 4kb

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 | |
|------|-------|-----|-------|--------|----------|------|-----|-----|---|
| S001 | S0001 | | | | | | | | |
| 0 | 1 | 1 | rs296 | 296000 | 1.15e-10 | 5 | 0 | 0 | 0 |
| | 0 | 5 | | | | | | | |

| (INDEX) | rs296 | KB | RSQ | ALLELES | F | P |
|---------|-------|----|-------|---------|---|----------|
| | rs296 | 0 | 1.000 | 0 | 1 | 1.15e-10 |
| | rs295 | -1 | 0.429 | 00/00 | 1 | 2.01e-08 |
| | rs296 | 0 | 1 | 00/00 | 2 | 2.6e-09 |
| | rs299 | 3 | 0.267 | 00/00 | 1 | 2.77e-09 |
| | rs299 | 3 | 0.267 | 00/00 | 2 | 7.29e-10 |

RANGE: chr1:295000..299000
SPAN: 4kb

From clump 1, let's choose rs139 (and not rs138, rs140 and rs141) and from clump 2 let's choose rs296 (and not rs295 and rs299) to carry forward to our mediation analysis. We will also carry forward rs1138 and rs1448 since these two SNPs are not part of any other clumps but are genome-wide significant for both traits.

Multivariate analysis

- a. Before moving on to dissecting the cross phenotype associations, let's see if we can include a few additional SNPs/regions to explore by using multivariate analysis. But let's only consider additional regions that are genome-wide suggestive for both phenotypes.

First run a multivariate analysis on Traits 1 and 2.

```
plink.multivariate\  
--noweb\  
--bfile pleiotropy_exercise\  
--mult-phenotype pleiotropy_exercise_phenotypes.txt\  
--sex\  
--mqfam\  
--out Trait1_Trait2
```

Please note: You should use the --noweb flag due to this program being built on an old version of PLINK.

- b. Now let's identify the intersection of SNPs that are genome-wide significant in the multivariate analysis and at least suggestive for each trait in the univariate analysis, i.e. we want to make sure that both traits are contributing to the multivariate signal.

```
Trait1<-read.table("Trait1_snp.assoc.linear", header=T)  
Trait2<-read.table("Trait2_snp.assoc.linear", header=T)  
multi<-read.table("Trait1_Trait2.mqfam.total", header=T)  
sigMulti<-subset(multi, P<0.00000005)  
suggTrait1<-subset(Trait1, P<0.000005)  
suggTrait2<-subset(Trait2, P<0.000005)  
Reduce(intersect, list(suggTrait1$SNP, suggTrait2$SNP, sigMulti$SNP))
```

Select the additional SNPs that are identified from the intersection of the multivariate analysis and genome-wide suggestive lists for both traits that were not in your original list.

We identify the following overlapping SNPs: rs125, rs135, rs137, rs138, rs139, rs140, rs141, rs295, rs296, rs298, rs299, rs300, rs920, rs921, rs923, rs1138, rs1166, rs1361, rs1448. Of course, this list includes the original set of 8 variants that were genome-wide significant for both Traits 1 and 2.

- c. You may want to re-run the LD clumping with a suggestive threshold to see if these additional SNPs clump with your existing clumps or are new potential regions to explore.

```
plink\  
--bfile pleiotropy_exercise\  
--clump Trait1_snp.assoc.linear,Trait2_snp.assoc.linear\  
--clump-p1 0.000005\  
--clump-p2 0.000005\  
--clump-r2 0.2\  
--clump-replicate\  
--clump-verbose\  
--out Trait1_Trait2_clump_suggestive
```

OUTPUT:

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 |
|------|-------|-------|--------|----------|-------|------|-----|-----|
| S001 | S0001 | | | | | | | |
| 1 | 2 | rs139 | 139000 | 2.86e-28 | 9 | 0 | 0 | 0 |
| 0 | 9 | | | | | | | |

| (INDEX) | SNP | KB | RSQ | ALLELES | F | P |
|---------|-------|----|-------|---------|---|----------|
| | rs139 | 0 | 1.000 | 0 | 2 | 2.86e-28 |
| | rs137 | -2 | 0.247 | 00/00 | 1 | 6.05e-08 |
| | rs137 | -2 | 0.247 | 00/00 | 2 | 2.17e-17 |
| | rs138 | -1 | 0.399 | 00/00 | 1 | 1.34e-09 |
| | rs138 | -1 | 0.399 | 00/00 | 2 | 1.91e-18 |
| | rs139 | 0 | 1 | 00/00 | 1 | 2.77e-15 |
| | rs140 | 1 | 0.229 | 00/00 | 1 | 6.05e-12 |
| | rs140 | 1 | 0.229 | 00/00 | 2 | 1.85e-26 |
| | rs141 | 2 | 0.235 | 00/00 | 1 | 9.9e-09 |
| | rs141 | 2 | 0.235 | 00/00 | 2 | 2.98e-13 |

RANGE: chr1:137000..141000

SPAN: 4kb

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 |
|------|-------|-------|--------|----------|-------|------|-----|-----|
| S001 | S0001 | | | | | | | |
| 1 | 1 | rs921 | 921000 | 6.29e-23 | 5 | 0 | 0 | 0 |
| 1 | 4 | | | | | | | |

| (INDEX) | SNP | KB | RSQ | ALLELES | F | P |
|---------|-------|----|-------|---------|---|----------|
| | rs921 | 0 | 1.000 | 0 | 1 | 6.29e-23 |
| | rs920 | -1 | 0.224 | 00/00 | 1 | 3.11e-08 |
| | rs920 | -1 | 0.224 | 00/00 | 2 | 6.25e-08 |
| | rs921 | 0 | 1 | 00/00 | 2 | 1.94e-07 |
| | rs922 | 1 | 0.202 | 00/00 | 1 | 4.52e-07 |

RANGE: chr1:920000..922000

SPAN: 2kb

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 |
|------|-------|-------|--------|---------|-------|------|-----|-----|
| S001 | S0001 | | | | | | | |
| 1 | 2 | rs136 | 136000 | 1.3e-17 | 5 | 0 | 1 | 0 |
| 0 | 4 | | | | | | | |

| (INDEX) | SNP | KB | RSQ | ALLELES | F | P |
|---------|-------|----|-------|---------|---|----------|
| | rs136 | 0 | 1.000 | 0 | 2 | 1.3e-17 |
| | rs134 | -2 | 0.229 | 00/00 | 2 | 3.97e-09 |
| | rs135 | -1 | 0.379 | 00/00 | 1 | 1.41e-06 |

rs135 -1 0.379 00/00 2 1.47e-09

RANGE: chr1:134000..136000
SPAN: 2kb

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 |
|---------|-------|--------|---------|----------|-------|------|----------|-----|
| S001 | S0001 | | | | | | | |
| 1 | 2 | rs1361 | 1361000 | 1.68e-12 | 9 | 0 | 1 | 2 |
| 1 | 5 | | | | | | | |
| (INDEX) | | rs1361 | 0 | 1.000 | 0 | 2 | 1.68e-12 | |
| | | rs1359 | -2 | 0.238 | 00/00 | 2 | 5.98e-10 | |
| | | rs1360 | -1 | 0.281 | 00/00 | 2 | 2.8e-11 | |
| | | rs1361 | 0 | 1 | 00/00 | 1 | 2.65e-07 | |
| | | rs1362 | 1 | 0.271 | 00/00 | 2 | 6.54e-10 | |
| | | rs1363 | 2 | 0.204 | 00/00 | 2 | 1.64e-07 | |

RANGE: chr1:1359000..1363000
SPAN: 4kb

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 |
|---------|-------|-------|--------|----------|-------|------|----------|-----|
| S001 | S0001 | | | | | | | |
| 1 | 1 | rs296 | 296000 | 1.15e-10 | 5 | 0 | 0 | 0 |
| 0 | 5 | | | | | | | |
| (INDEX) | | rs296 | 0 | 1.000 | 0 | 1 | 1.15e-10 | |
| | | rs295 | -1 | 0.429 | 00/00 | 1 | 2.01e-08 | |
| | | rs295 | -1 | 0.429 | 00/00 | 2 | 8.62e-08 | |
| | | rs296 | 0 | 1 | 00/00 | 2 | 2.6e-09 | |
| | | rs299 | 3 | 0.267 | 00/00 | 1 | 2.77e-09 | |
| | | rs299 | 3 | 0.267 | 00/00 | 2 | 7.29e-10 | |

RANGE: chr1:295000..299000
SPAN: 4kb

| CHR | F | SNP | BP | P | TOTAL | NSIG | S05 | S01 |
|---------|-------|--------|---------|----------|-------|------|----------|-----|
| S001 | S0001 | | | | | | | |
| 1 | 1 | rs1138 | 1138000 | 9.58e-10 | 3 | 0 | 1 | 0 |
| 0 | 2 | | | | | | | |
| (INDEX) | | rs1138 | 0 | 1.000 | 0 | 1 | 9.58e-10 | |

| | | | | | |
|--------|----|-------|-------|---|----------|
| rs1137 | -1 | 0.315 | 00/00 | 1 | 4.09e-07 |
| rs1138 | 0 | 1 | 00/00 | 2 | 2.9e-09 |

RANGE: chr1:1137000..1138000
SPAN: 1kb

Based on the multivariate analysis and additional clumping, you should add the following SNPs to your list of SNPs for mediation: rs125, rs135, rs300, rs921, rs923, rs1166, rs1361.

The final list of SNPs that were selected to carry through to the mediation analysis are:

rs125, rs135, rs139, rs296, rs300, rs921, rs923, rs1138, rs1166, rs1361, rs1448

Mediation analyses

- a. For each SNPs that you have identified as a cross phenotype association (evidence of overlapping association signals as well as incorporating results from LD clumping and multivariate association) you will need to extract this data from the original plink files and create a genotype file that is coded as 0|1|2 for the genotypes. This can be done in PLINK using the `--recodeA` command and the `--extract` command by providing a file with the list of snps. This will give you a `.raw` genotype file with only the snps that you will be using in the mediation analysis.

- b. Conduct a mediation analysis in R using the *mediation* R library. Sample code for this is below (Note: replace `<SNP>` with the variable name for the SNP you are investigating. You will need to repeat this for each SNP that you have selected):

```
library(mediation)
genotypes <- read.table("snps_for_mediation.raw", header=T)
phenotypes<-read.table("pleiotropy_exercise_phenotypes.txt", header=T)
combined<-merge(genotypes,phenotypes)
med.fit<-lm(Trait1~rs125_0, data=combined)
out.fit<-lm(Trait2~Trait1+rs125_0, data=combined)
med.out<-mediate(med.fit,out.fit,treat="rs125_0", mediator="Trait1", boot=TRUE,
+boot.ci.type="bca", sims=1000)
summary(med.out)
```

This will print out a summary of the mediation analysis.

Please note: The more simulations (sims) you specific in the med.out step the more the CI and p-value estimates will be, however, this can also be time-consuming. If this step is taking a substantial amount of time (>20 minutes) you may want to reduce the number of simulations for the purposes of completing the exercise.

Questions:

- 1) Which of the SNPs have genome-wide significant ($p < 5 \times 10^{-8}$) associations for both traits?

rs138, rs139, rs140, rs141, rs296, rs299, rs1138, rs1448

- 2) Did the multivariate analyses result in additional SNPs that had genome-wide significant cross phenotype associations but that also had genome-wide suggestive ($p < 5 \times 10^{-6}$) univariate association for each trait? Which SNP(s)?

rs125, rs135, rs137, rs295, rs298, rs300, rs920, rs921, rs923, rs1166, rs1361

Instead of running mediation analysis on all 19 SNPs, I suggested that you perform LD clumping to reduce this number of SNPs and only focus on the index SNP for each clump (or if the index SNP was not associated with both traits to choose another SNP from among the clumped SNPs). This reduced the set of SNPs to 11.

- 3) For each SNP analyzed in the mediation analysis, determine if there is a significant direct effect which is indicative of some level of biological pleiotropy. Do any of the SNPs exhibit complete mediation?

All SNPs show a significant direct effect on Trait 2 indicating some level of biological pleiotropy. rs923 has an ADE p-value of 0.002 but this is still less than the Bonferroni corrected p-value of 0.0045, adjusting for the 11 SNPs. No SNP shows an association with Trait 2 that is completely mediated through its association with Trait 1, i.e. an ACME estimate that is equal to (or close to) the total effect. The strongest mediated effect is for rs921 in which the mediated effect accounts for ~40% of the total effect of the SNP on Trait 2.

- 4) Why do some of the SNPs have negative values for the proportion mediated?

The estimate of the proportion mediated is not the best way to interpret the mediation results, despite its seemingly obvious interpretability. In reality this proportion does not range from 0-1 but can rather be less than 0 and greater than 1. The negative proportion mediated values that we see for many of the SNPs we have analyzed is due to the fact that these SNPs have an effect estimate for the total effect and mediated effect that are in opposite directions, i.e. the effect of the SNPs on Trait 1 and Trait 2 is in opposite directions. Depending on your study question, you may want to limit your selection of SNPs to only those with effects on the two traits in the same direction. We did not see this among our SNPs, but a proportion mediated > 1 can happen when the strength of the association with the mediator (Trait 1) is much higher than the strength of the association with the outcome (Trait 2). This is often why it is recommended that the direct, indirect and total effects be used in the interpretation rather than the proportion mediated.

Summary table of pleiotropy results

| SNP | Beta (Trait 1) | P (Trait 1) | Beta (Trait 2) | P (Trait 2) | MV ¹ P | MV ¹ Loading (Trait 1) | MV ¹ Loading (Trait 2) | ADE | ADE (P) | ACME | ACME (P) | Total Effect | Total Effect (P) | Prop Mediated | Prop Mediated (P) |
|--------|----------------|-------------|----------------|-------------|-------------------|-----------------------------------|-----------------------------------|--------|---------|--------|----------|--------------|------------------|---------------|-------------------|
| rs125 | 0.072 | 1.08E-08 | 0.062 | 6.45E-07 | 1.80E-10 | 0.8397 | 0.7096 | 0.045 | <2e-16 | 0.015 | <2e-16 | 0.0596 | <2e-16 | 0.2516 | <2e-16 |
| rs135 | -0.040 | 1.41E-06 | 0.050 | 1.47E-09 | 2.82E-17 | -0.5457 | 0.7024 | 0.059 | <2e-16 | -0.008 | <2e-16 | 0.0509 | <2e-16 | -0.3317 | <2e-16 |
| rs139 | -0.065 | 2.77E-15 | 0.090 | 2.86E-28 | 2.26E-50 | -0.5249 | 0.7196 | 0.103 | <2e-16 | -0.014 | <2e-16 | 0.0891 | <2e-16 | -0.1580 | <2e-16 |
| rs296 | -0.056 | 1.15E-10 | -0.051 | 2.60E-09 | 1.78E-14 | 0.8100 | 0.7456 | -0.039 | <2e-16 | -0.012 | <2e-16 | -0.0512 | <2e-16 | 0.2306 | <2e-16 |
| rs300 | -0.046 | 1.85E-08 | -0.039 | 2.09E-06 | 1.34E-10 | -0.8386 | -0.7110 | -0.029 | <2e-16 | -0.010 | <2e-16 | -0.0392 | <2e-16 | 0.2507 | <2e-16 |
| rs921 | 0.109 | 6.29E-23 | 0.057 | 1.95E-07 | 1.16E-24 | -0.9475 | -0.5144 | 0.036 | <2e-16 | 0.023 | <2e-16 | 0.0595 | <2e-16 | 0.3908 | <2e-16 |
| rs923 | 0.041 | 1.48E-06 | 0.042 | 4.04E-07 | 2.35E-09 | 0.7615 | 0.7957 | 0.034 | 0.002 | 0.009 | <2e-16 | 0.0421 | <2e-16 | 0.2035 | <2e-16 |
| rs1138 | -0.050 | 9.58E-10 | 0.048 | 2.90E-09 | 2.44E-20 | 0.6511 | -0.6027 | 0.058 | <2e-16 | -0.011 | <2e-16 | 0.0468 | <2e-16 | -0.2319 | <2e-16 |
| rs1166 | -0.051 | 4.77E-10 | 0.041 | 6.30E-07 | 1.02E-17 | -0.7175 | 0.5275 | 0.049 | <2e-16 | -0.011 | <2e-16 | 0.0382 | <2e-16 | -0.2918 | <2e-16 |
| rs1361 | -0.056 | 2.65E-07 | 0.076 | 1.68E-12 | 3.52E-21 | -0.5349 | 0.7114 | 0.087 | <2e-16 | -0.012 | <2e-16 | 0.0751 | <2e-16 | -0.1614 | <2e-16 |
| rs1448 | -0.079 | 3.46E-08 | 0.092 | 1.51E-11 | 2.21E-20 | -0.5847 | 0.6679 | 0.108 | <2e-16 | -0.017 | <2e-16 | 0.0908 | <2e-16 | -0.1879 | <2e-16 |

¹Multivariate

Computer Practical Exercise on Family-based Association using FaST-LMM, PLINK and R

Overview

Purpose

In this exercise you will be carrying out association analysis of data from a mini genome-wide association study. The data comes from families (related individuals) measured for a quantitative trait of interest. The purpose is detect which (if any) of the loci are associated with the quantitative trait.

Methodology

We will use the linear mixed model approach implemented in FaST-LMM and (for comparison) standard linear regression in PLINK.

Program documentation

PLINK documentation:

PLINK has an extensive set of documentation including a pdf manual, a web-based tutorial and web-based documentation:

Original PLINK (1.07) (which has arguably clearer documentation):

<http://zzz.bwh.harvard.edu/plink/>

New PLINK (1.90) (which includes documentation on new additional features):

<https://www.cog-genomics.org/plink2>

R documentation:

The R website is at <http://www.r-project.org/>

From within R, one can obtain help on any command `xxxx` by typing ``help(xxxx)'`

FaST-LMM documentation:

Documentation can be downloaded together with the FaST-LMM program from

<http://research.microsoft.com/en-us/downloads/aa90ccfb-b2a8-4872-ba00-32419913ca14/>

Data overview

We will be using family data consisting of 498 individuals typed at 134,946 SNPs. All individuals have measurements of a quantitative trait of interest. You can assume that appropriate quality control (QC) checks on SNPs and individuals have been carried out prior to the current analysis i.e. the data set is already QC-ed.

Appropriate data

Appropriate data for this exercise is genome-wide genotype data for related and/or apparently unrelated individuals. Genome-wide data is required in order to estimate relationships between people and allow for relatedness in the analysis. The individuals should be phenotyped for either a dichotomous trait or a quantitative trait of interest.

Instructions

Data files

The data is in PLINK binary-file format. Check you have the required files by typing:

```
ls -l
```

You should find 3 PLINK binary-format files in your directory: `quantfamdata.bed`, `quantfamdata.bim` and `quantfamdata.fam`. The file `quantfamdata.bed` is the binary genotype file which will not be human readable. The file `quantfamdata.bim` is a map file. You can take a look at this (e.g. by typing `more quantfamdata.bim`). The file `quantfamdata.fam` gives the pedigree structure in a format that is compatible with the binary genotype file. You can take a look at this (e.g. by typing `more quantfamdata.fam`). Note this file is the same as the first six columns of a standard pedigree file, with the last column giving each individual's quantitative trait value.

Step-by-step instructions

1. Analysis in PLINK

To start with, we will use PLINK to perform a test equivalent to linear regression analysis, without worrying about the relatedness between individuals:

```
plink --bfile quantfamdata --assoc --out plinkresults
```

A copy of the screen output is saved in the file `plinkresults.log`. The association results are output to a file `plinkresults.qassoc`. Take a look at this file. Each line corresponds to the results for a particular SNP. Each line contains the following columns:

| | |
|--------------|--|
| CHR | Chromosome number |
| SNP | SNP identifier |
| BP | Physical position (base-pair) |
| NMISS | Number of non-missing genotypes |
| BETA | Regression coefficient |
| SE | Standard error |
| R2 | Regression r-squared |
| T | Wald test (based on t-distribution) |
| P | Wald test asymptotic p-value |

The most useful columns are T (the test statistic) and its p value (P).

To visualise these results properly we will use R. Open up a new terminal window, move to the directory where you performed this analysis, and start R (by typing `R`).

Now (within R) read in the data by typing:

```
res1<-read.table("plinkresults.qassoc", header=T)
```

This reads the results into a dataframe named "res1". To see the top few lines of this dataframe, type:

```
head(res1)
```

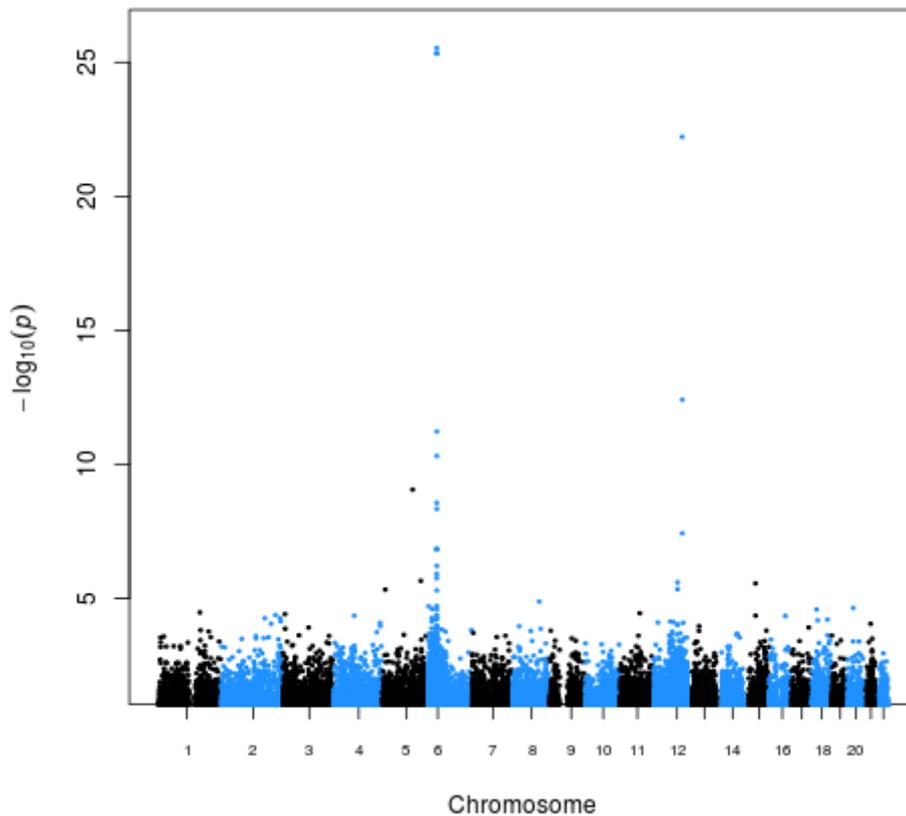
The data frame has 134,946 lines, one for each SNP. It would be very laborious to go through and look at each line by eye. Instead we will plot the results for all chromosomes, colouring each chromosome differently. To do this we need to first read in from an external file some special functions for creating such ``Manhattan'' plots:

```
source("qqmanHJCupdated.R")
```

Then we use the following command to actually make the plot, and save it in the file "mh1.png":

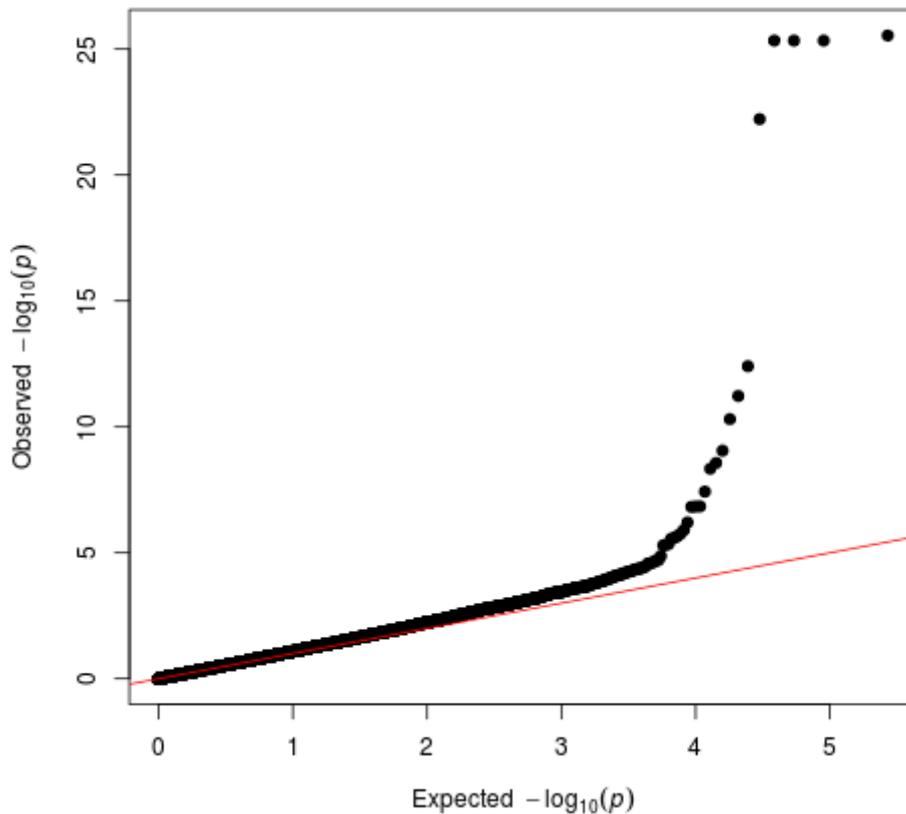
```
png("mh1.png")
manhattan(res1, pch=20, suggestiveline=F, genomewideline=F, ymin=2,
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)
dev.off()
```

Be warned, this may take some time to plot.



Visually it looks like there may be significant results on chromosomes 6 and 12, and possibly on chromosome 5 as well. One way to assess the significance of the results, in light of the large number of tests performed, is to use a Q-Q plot. To plot a Q-Q plot for these P values, and save it in the file "qq1.png", type:

```
png("qq1.png")  
qq(res1$P)  
dev.off()
```



What one would hope to see is most of the values lying along the straight line with gradient 1, indicating that most results are consistent with the null hypothesis of no association. However, one would also hope to see a few high values at the top that depart from the straight line, which are hopefully true associations.

Our results seem fairly consistent with this expectation, but there may be a little bit of inflation (i.e. a slope slightly bigger than 1) due to relatedness between individuals. To calculate the genomic control inflation factor, we first convert the P values to chi-squared test statistics on 1df, and then use the formula from Devlin and Roeder (1999):

```
chi<-(qchisq(1-res1$P,1))
lambda=median(chi)/0.456
lambda
```

You should find a slightly inflated value (lambda=1.10)

2. Analysis in FaST-LMM

Now we will try re-running the analysis using FaST-LMM, which estimates and accounts for the relatedness between individuals. Go back to the window where you ran PLINK and run FaST-LMM as follows:

```
fastlmmc -bfile quantfamdata -pheno quantfamdata.fam -mpheno 4 -bfileSim
quantfamdata -ML -out FLMMresults
```

Here we use the `-bfile quantfamdata` command to tell the program the name (stem) of the files with the input genotype data containing the SNPs to be tested for association, and the `-bfileSim quantfamdata` command to tell the program the

name of the files containing the SNPs to be used for estimating relatedness. Here we just use the same files both times, but FaST-LMM would allow us to use different files for these two operations if we prefer.

The command `-pheno quantfamdata.fam -mpheno 4` tells FaST-LMM to read the phenotype data in from the file `quantfamdata.fam`, using the 4th phenotype column (not including the two first columns which give the family and person IDs). The `-ML` command tells FaST-LMM to use maximum likelihood estimation (in case you prefer this as opposed to the default restricted maximum likelihood (REML)). The command `-out FLMMresults` tells FaST-LMM the name to use for the output file.

Take a look at the results file. FaST-LMM automatically orders the results by significance.

Now go back to your R window and read the results into R:

```
res2<-read.table("FLMMresults", header=T)
```

Check the column names by typing:

```
head(res2)
```

The P value is in a column called ```Pvalue```. Remember FaST-LMM has automatically ordered the results by significance, so these top few rows will show the most significant results.

First let us check the genomic control inflation factor. We convert the P values to chi-squared test statistics on 1df, and then use the formula from Devlin and Roeder (1999):

```
chi<-(qchisq(1-res2$Pvalue,1))
lambda=median(chi)/0.456
lambda
```

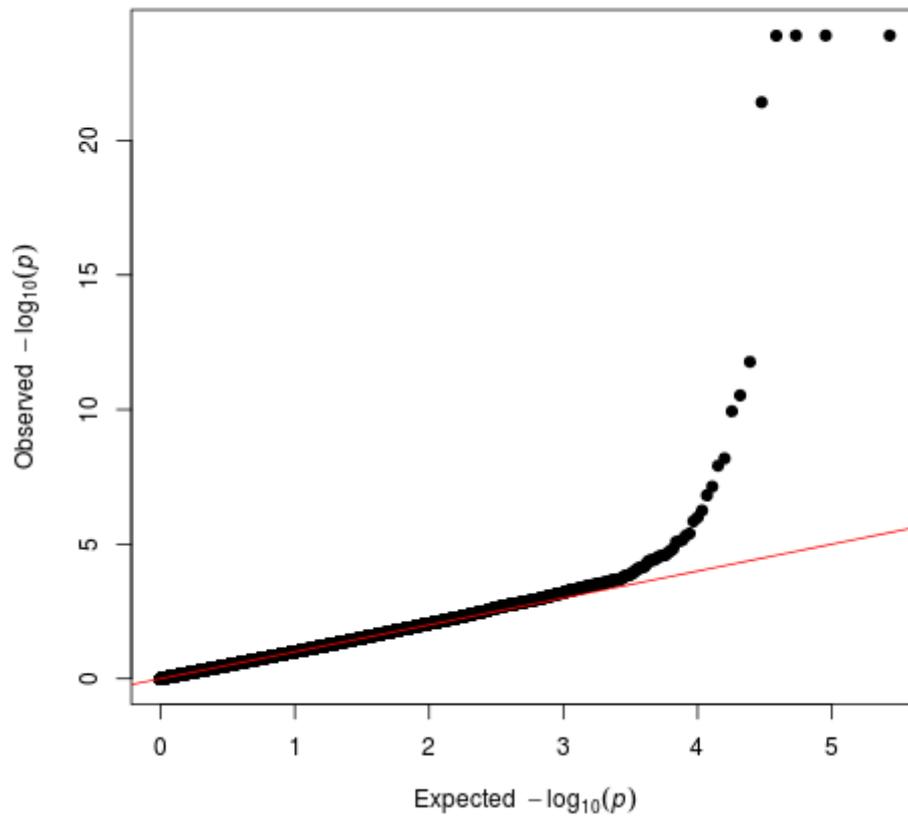
You should find a less inflated value (`lambda=0.99`) than we found previously with PLINK.

To plot Manhattan and Q-Q plots you can use similar commands to before, but the columns need to be named appropriately. The easiest thing is to make a new smaller dataframe containing the required data:

```
new<-data.frame(res2$SNP, res2$Chromosome, res2$Position, res2$Pvalue)
names(new)<-c("SNP", "CHR", "BP", "P")
head(new)
```

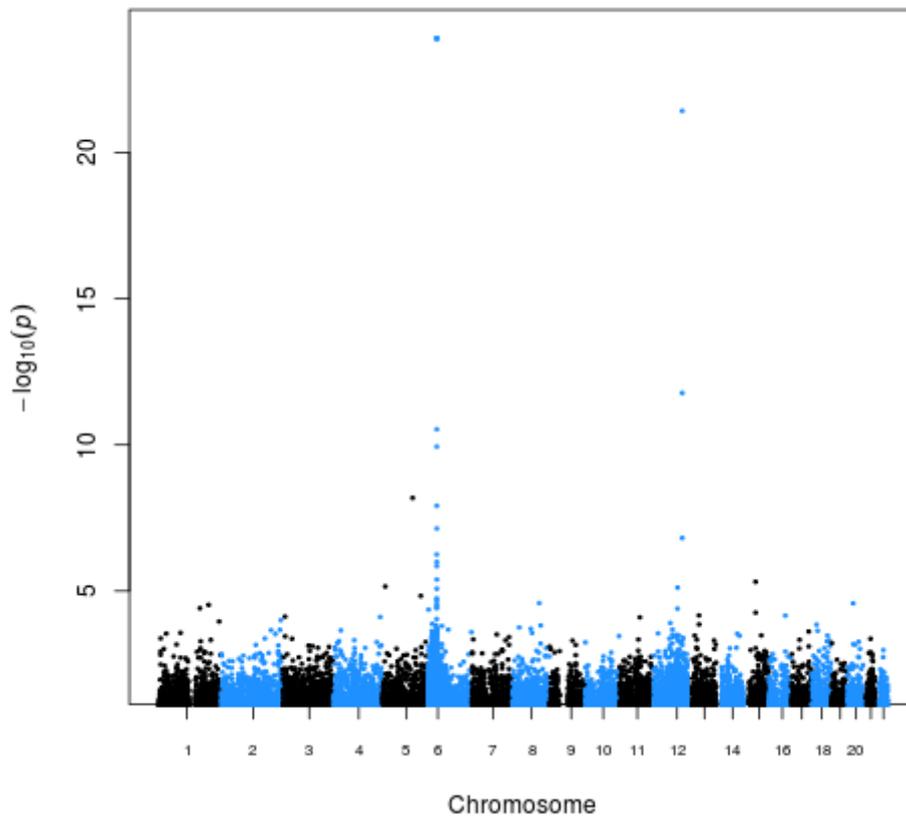
Now you can plot the Q-Q plot:

```
png("qq2.png")
qq(new$P)
dev.off()
```



And the Manhattan plot:

```
png("mh2.png")  
manhattan(new, pch=20, suggestiveline=F, genomewideline=F, ymin=2,  
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)  
dev.off()
```



The significant effects on chromosomes 6 and 12 are still easily visible. In fact, this is simulated data, and these signals do correspond correctly to the positions of the underlying causal variants.

Answers

How to interpret the output

Interpretation of the output is described in the step-by-step instructions. In general, the output will consist of a likelihood-ratio or chi-squared test for whatever you are test you are performing, and regression coefficients or odds ratio estimates for the predictor variables in the current model. Please ask if you need help in understanding the output for any specific test.

Comments

Advantages/disadvantages

PLINK is useful for data management and analysis of genome-wide association data. FaST-LMM is more appropriate for analysis of related individuals, or for correcting for population stratification in apparently unrelated individuals.

Other packages

Other packages that can implement a similar analysis to FaST-LMM include EMMAX, GEMMA, MMM, GenABEL, Mendel.

References

Lippert C, Listgarten J, Liu Y, Kadie CM, Davidson RI, Heckerman D (2011) FaST linear mixed models for genome-wide association studies Nat Methods 8(10):833-835.

Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MAR, Bender D, Maller J, Sklar P, de Bakker PIW, Daly MJ & Sham PC (2007) PLINK: a toolset for whole-genome association and population-based linkage analysis. American Journal of Human Genetics, 81:559-575.

Exercises prepared by: Heather Cordell

Checked by:

Programs used: PLINK, R, FaST-LMM

Last updated: 01/17/2020 12:33:40

Computer Practical Exercise using GCTA (with R)

Overview

Purpose

This exercise repeats the linear mixed model analysis from the previous exercise using the program GCTA instead of FaST-LMM. In addition, we use GCTA to estimate the heritability accounted for by all genotyped SNPs, and by various subsets of SNPs.

Methodology

We will use the linear mixed model approach implemented in GCTA.

Program documentation

GCTA documentation:

Documentation can be obtained together with the GCTA program from:

<http://cnsgenomics.com/software/gcta/>

Data overview

As a reminder, we are using family data consisting of 498 individuals typed at 134,946 SNPs. All individuals have measurements of a quantitative trait of interest.

Appropriate data

Appropriate data for this exercise is genome-wide genotype data for individuals who are phenotyped for either a dichotomous trait or a quantitative trait of interest. GCTA is really designed for the analysis of apparently unrelated individuals, but in this case we will apply it to a set of related individuals, in order to compare the results with those we obtained previously for these individuals.

Instructions

Data files

We will use the same PLINK binary-file format files `quantfamdata.bed`, `quantfamdata.bim` and `quantfamdata.fam` used previously. We will also use R to create an additional phenotype file required by GCTA.

Step-by-step instructions

1. Create phenotype file in R

To start with, we will use R to create the phenotype file required by GCTA. Start R (by typing `R`) and create a new phenotype file from the `.fam` file by typing the following commands:

```
fam<-read.table("quantfamdata.fam", header=F)
pheno=data.frame(fam[,1:2],fam[,6])
write.table(pheno,file="phenos.txt",col.names=F,row.names=F,quote=F)
```

Take a look at the file `phenos.txt` that you just created, to check you understand it.

2. GCTA Analysis

To use GCTA to perform association analysis while allowing for relatedness between individuals, type:

```
gcta64 --mlma --bfile quantfamdata --pheno phenos.txt --out GCTAresults
```

Here we use the `--mlma` option to tell GCTA to perform association analysis, we use the `--bfile` and `--pheno` options to tell GCTA which files to read in the genotype and phenotype data from, and we use `GCTAresults` as the stem name for the output files.

To calculate the genomic control inflation factor, and to produce QQ and Manhattan plots from the above analysis, you can use the following sequence of commands within R. (Make sure that you understand the commands - if not please ask an instructor).

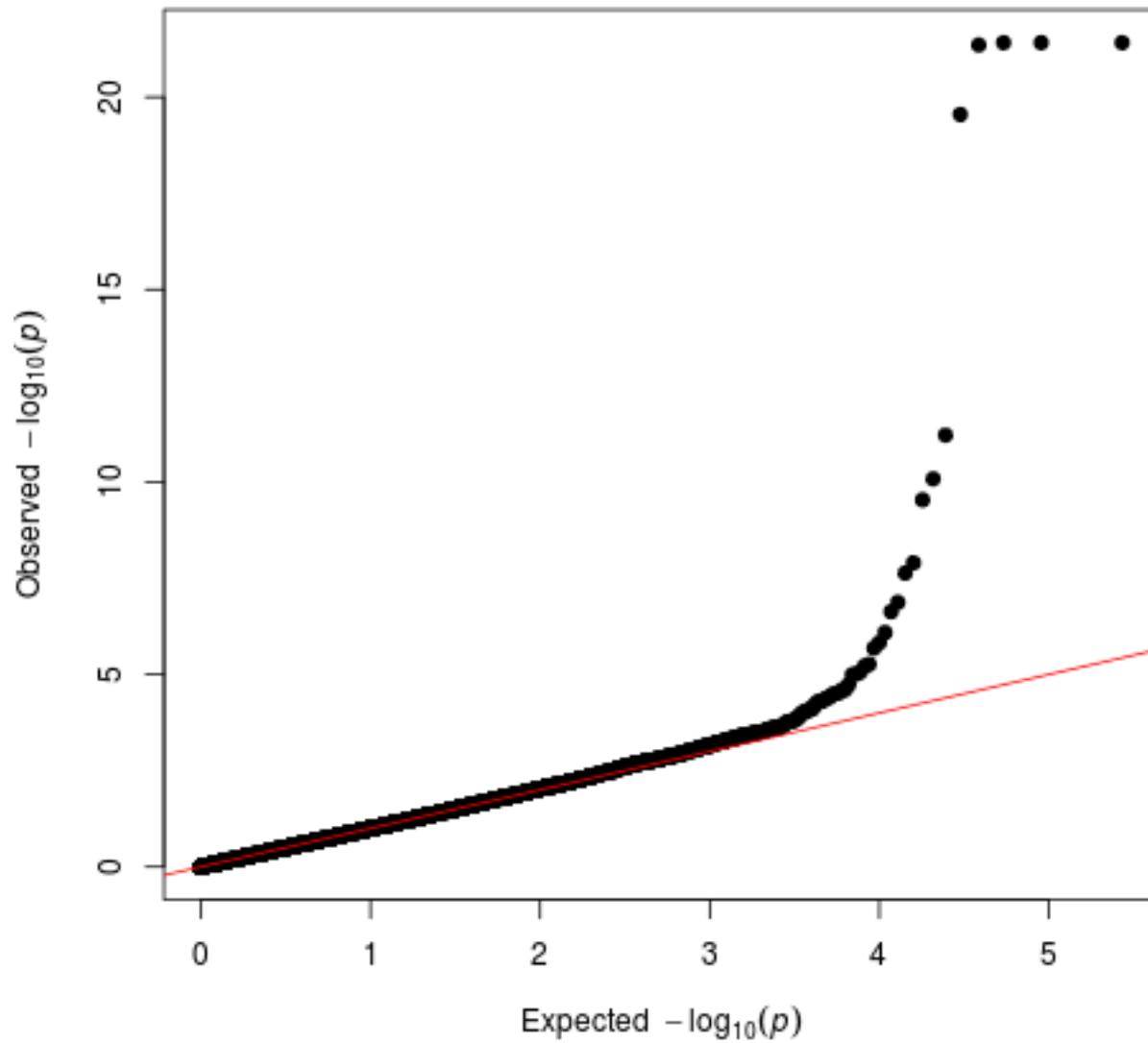
```
source("qqmanHJCupdated.R")

res3<-read.table("GCTAresults.mlma", header=T)
head(res3)

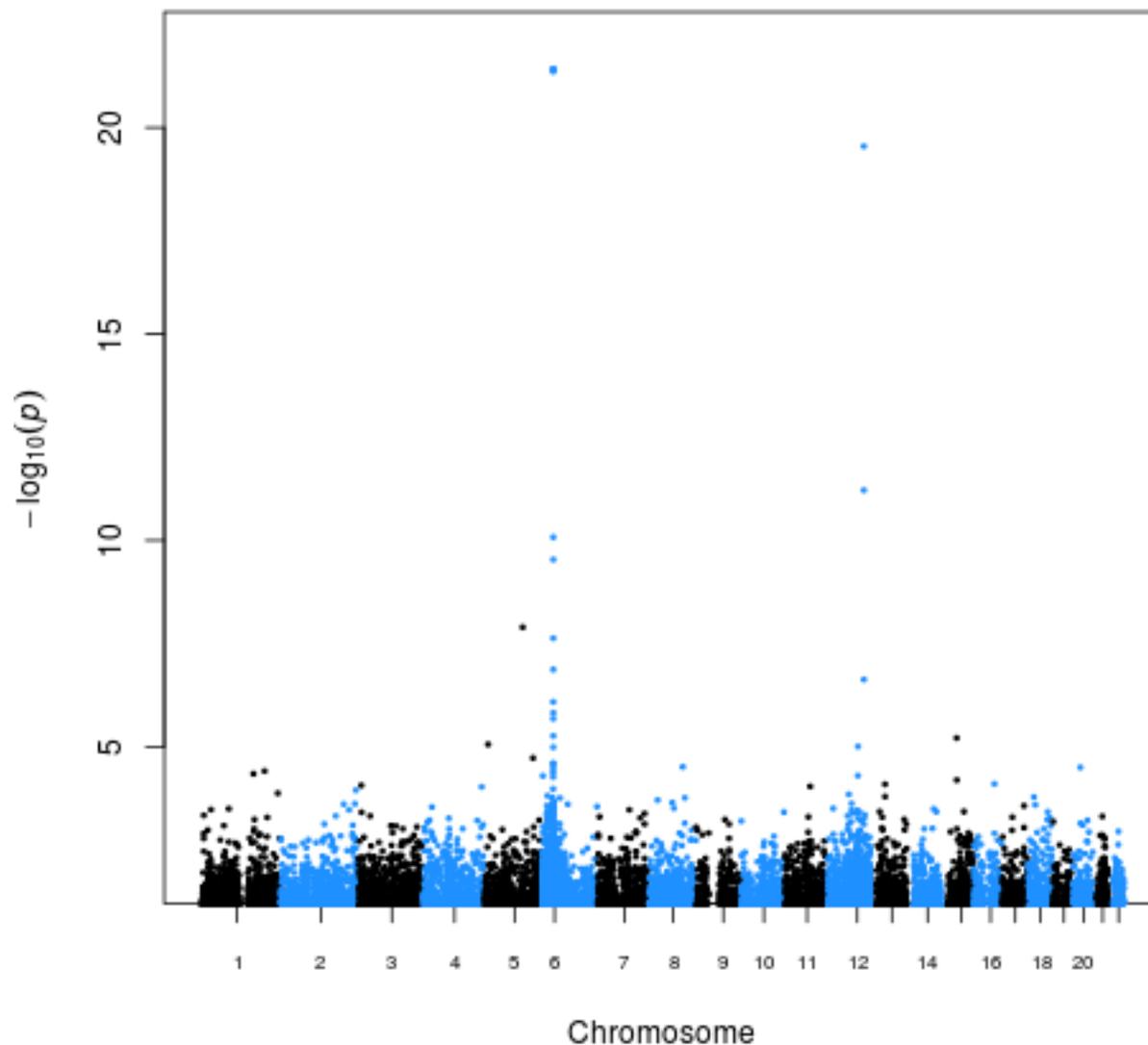
chi<-(qchisq(1-res3$p,1))
lambda=median(chi)/0.456
lambda
```

```
new3<-data.frame(res3$SNP, res3$Chr, res3$bp, res3$p)
names(new3)<-c("SNP", "CHR", "BP", "P")
head(new3)
```

```
png("qq3.png")
qq(new3$P)
dev.off()
```



```
png("mh3.png")
manhattan(new3, pch=20, suggestiveline=F, genomewideline=F, ymin=2,
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)
dev.off()
```



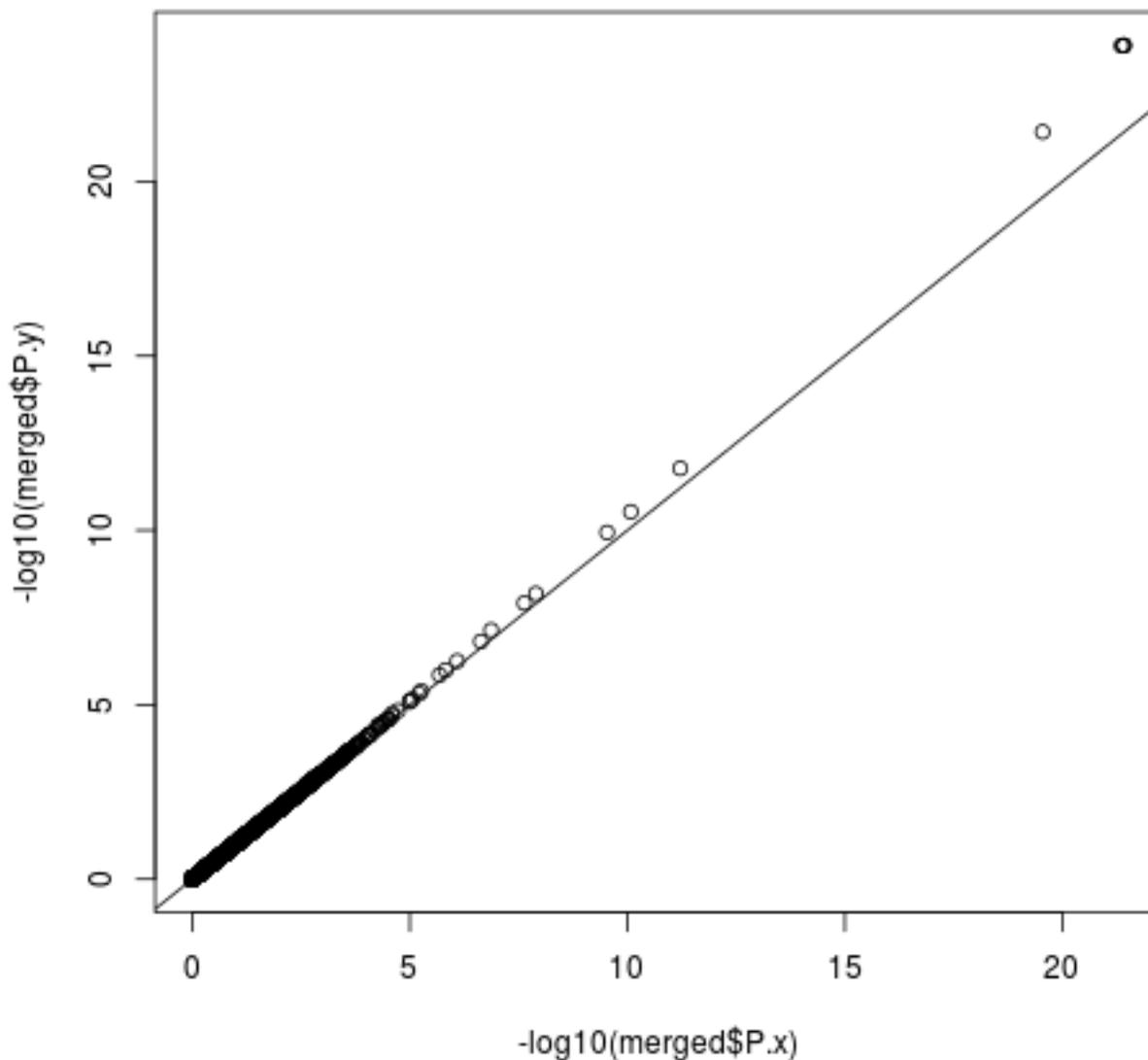
You should find that the genomic control factor is close to 1.0, and the QQ and Manhattan plots are similar to those you obtained from FaST-LMM.

To compare the results (`res3`) with our previous FaST-LMM results (`res2`), use the following sequence of commands within R:

```
res2<-read.table("FLMMresults", header=T)
new2<-data.frame(res2$SNP, res2$Chromosome, res2$Position, res2$Pvalue)
names(new2)<-c("SNP", "CHR", "BP", "P")
merged=merge(new3,new2, by="SNP", sort=F)

head(res2)
head(new2)
head(new3)
head(merged)

png("compareGCTAFLMM.png")
plot(-log10(merged$P.x),-log10(merged$P.y))
abline(0,1)
dev.off()
```



You should find that the GCTA results (on the x axis) are very similar to the FaST-LMM results (on the y axis), although the $-\log_{10}$ P-values from FaST-LMM are consistently just a little bit higher than those from GCTA.

To use GCTA to estimate the heritability accounted for by all autosomal genome-wide SNPs, you need to first estimate the GRM, and then use the GRM to estimate the (SNP) heritability. This can be achieved using the following commands:

```
gcta64 --bfile quantfamdata --autosome --make-grm-bin --out GCTAgrim
gcta64 --reml --grm-bin GCTAgrim --pheno phenos.txt --out GCTAherit
```

The screen output estimates the SNP heritability $V(G)/V_p$ to be 0.480590 or around 48%.

To estimate the heritability accounted for by SNPs on chromosomes 1, 2, 6 and 12 (for example), use the following commands:

```
gcta64 --bfile quantfamdata --chr 1 --make-grm-bin --out GCTAgrimchr1
gcta64 --reml --grm-bin GCTAgrimchr1 --pheno phenos.txt \
--out GCTAheritchr1
```

```
gcta64 --bfile quantfamdata --chr 2 --make-grm-bin --out GCTAgrimchr2
gcta64 --reml --grm-bin GCTAgrimchr2 --pheno phenos.txt \
--out GCTAheritchr2
```

```
gcta64 --bfile quantfamdata --chr826 --make-grm-bin --out GCTAgrimchr6
gcta64 --reml --grm-bin GCTAgrimchr6 --pheno phenos.txt \
```

```
--out GCTAheritchr6
```

```
gcta64 --bfile quantfamdata --chr 12 --make-grm-bin --out GCTAgrmchr12  
gcta64 --reml --grm-bin GCTAgrmchr12 --pheno phenos.txt \  
--out GCTAheritchr12
```

You should find that the SNP heritabilities on chromosomes 1 and 2 do not look particularly significant (given the estimated standard errors), but the SNP heritabilities on chromosomes 6 and 12 are significant (as might be expected from the strong effects seen on these chromosomes).

The sum of the SNP heritabilities on these 4 chromosomes (0.206479+0.111512+0.368184+0.286570) adds up to more than the overall SNP heritability of 0.480589. This is due to the phenomenon that, in the presence of population substructure or close relatedness, chromosome-specific heritability estimates can be confounded by shared non-genetic effects (for examples shared environment) or correlations between SNPs on different chromosomes, leading to an over-estimate of the chromosome-specific heritability.

To correctly partition the overall heritability between the 22 autosomes, we need to first estimate chromosome-specific GRMs and then include them all simultaneously in the model.

We first calculate the GRMs for all additional chromosomes:

```
gcta64 --bfile quantfamdata --chr 3 --make-grm-bin --out GCTAgrmchr3  
gcta64 --bfile quantfamdata --chr 4 --make-grm-bin --out GCTAgrmchr4  
gcta64 --bfile quantfamdata --chr 5 --make-grm-bin --out GCTAgrmchr5  
gcta64 --bfile quantfamdata --chr 7 --make-grm-bin --out GCTAgrmchr7  
gcta64 --bfile quantfamdata --chr 8 --make-grm-bin --out GCTAgrmchr8  
gcta64 --bfile quantfamdata --chr 9 --make-grm-bin --out GCTAgrmchr9  
gcta64 --bfile quantfamdata --chr 10 --make-grm-bin --out GCTAgrmchr10  
gcta64 --bfile quantfamdata --chr 11 --make-grm-bin --out GCTAgrmchr11  
gcta64 --bfile quantfamdata --chr 13 --make-grm-bin --out GCTAgrmchr13  
gcta64 --bfile quantfamdata --chr 14 --make-grm-bin --out GCTAgrmchr14  
gcta64 --bfile quantfamdata --chr 15 --make-grm-bin --out GCTAgrmchr15  
gcta64 --bfile quantfamdata --chr 16 --make-grm-bin --out GCTAgrmchr16  
gcta64 --bfile quantfamdata --chr 17 --make-grm-bin --out GCTAgrmchr17  
gcta64 --bfile quantfamdata --chr 18 --make-grm-bin --out GCTAgrmchr18  
gcta64 --bfile quantfamdata --chr 19 --make-grm-bin --out GCTAgrmchr19  
gcta64 --bfile quantfamdata --chr 20 --make-grm-bin --out GCTAgrmchr20  
gcta64 --bfile quantfamdata --chr 21 --make-grm-bin --out GCTAgrmchr21  
gcta64 --bfile quantfamdata --chr 22 --make-grm-bin --out GCTAgrmchr22
```

We then run the analysis:

```
gcta64 --reml --mgrm-bin multipleGRMs.txt --pheno phenos.txt \  
--out GCTAherit22GRMs
```

Note this command makes use of a file `multipleGRMs.txt` which we created for you in advance, listing the stem names of the individual GRM files. Unfortunately, in this example the analysis fails to converge, probably because this type of analysis ideally requires a larger number of less closely related individuals.

To instead partition the heritability among two sets of SNPs, chromosome 6

and all other autosomes, we first join together the GRMs for all non-chromosome 6 chromosomes:

```
gcta64 --mgrm-bin multipleGRMsnot6.txt --make-grm --out GCTAgrmnot6
```

Note this command makes use of another file `multipleGRMsnot6.txt` which we created for you in advance, listing the stem names of the individual GRM files (excluding the one for chromosome 6).

We will run the analysis making use of another file `multipleGRMs6andnot6.txt` which we created for you in advance. Take a look at this file and check you understand it.

To run the analysis type:

```
gcta64 --reml --mgrm-bin multipleGRMs6andnot6.txt --pheno phenos.txt \
--out GCTAherit6andnot6
```

The results suggest that a total SNP heritability of 0.469171 can be partitioned as 0.294445 accounted for by chromosome 6, and 0.174726 accounted for by the other autosomes.

3. GCTA fastGWA Analysis

GCTA has an alternative method for performing mixed linear model (MLM)-based GWAS analysis that is particularly designed for large biobank-scale datasets such as the UK Biobank. Here we will apply it to the (much smaller scale) dataset that we have already analysed.

First we have to make a sparse genetic relationship matrix (GRM) from the full-dense GRM e.g. using a cutoff value of 0.05 (so entries less than 0.05 are set to 0):

```
gcta64 --grm GCTAgrm --make-bK-sparse 0.05 --out GCTAsparsegrm
```

This creates a file `GCTAsparsegrm.grm.sp` containing the pairs of individuals whose entries in the GRM are greater than 0.05.

For real biobank-scale data, creating the full-dense GRM in the first place can be computationally challenging, and it can be advantageous to partition the GRM into m parts (by row), and compute the parts separately (before joining them back together). To compute i -th part in the current run, we use the syntax `--make-grm-part m i`. For example, use the following commands to recalculate the full-dense GRM by partitioning the calculation into 3 parts (while also using 5 threads):

```
gcta64 --bfile quantfamdata --make-grm-part 3 1 --thread-num 5 \
--out test
gcta64 --bfile quantfamdata --make-grm-part 3 2 --thread-num 5 \
--out test
gcta64 --bfile quantfamdata --make-grm-part 3 3 --thread-num 5 \
--out test
```

Merge all the parts together:

```
cat test.part_3_*.grm.id > test.grm.id
cat test.part_3_*.grm.bin > test.grm.bin
cat test.part_3_*.grm.N.bin > test.grm.N.bin
```

Now we can create a sparse GRM from this re-calculated version of the full-dense GRM:

```
gcta64 --grm test --make-bK-sparse 0.05 --out newsparsegrm
```

Check that the top few lines of newsparsegrm.grm.sp seem to match GCTAsparsegrm.grm.sp:

```
head *.sp
```

To perform the association analysis, we would normally use the sparse GRM to model close relationships as random effects, while additionally including principal components as fixed effects. So let us first use GCTA to calculate the first 5 principal components:

```
gcta64 --grm-bin GCTAgrm --pca 5 --out pcs
```

To use original GCTAsparsegrm.grm.sp in a fastGWA analysis, type:

```
gcta64 --bfile quantfamdata --grm-sparse GCTAsparsegrm \
--fastGWA-mlm --pheno phenos.txt --qcovar pcs.eigenvec \
--out sparse_assoc
```

To use newsparsegrm.grm.sp in a fastGWA analysis, type:

```
gcta64 --bfile quantfamdata --grm-sparse newsparsegrm \
--fastGWA-mlm --pheno phenos.txt --qcovar pcs.eigenvec \
--out newsparse_assoc
```

Once we have included the principal components, it seems that the estimate of V_g is not statistically significant, and so fastGWA has automatically moved to using linear regression rather than a linear mixed model. This is not really what we wanted! Let us not include principal components - we would then expect the estimate of V_g to be significant, and so in this way we force fastGWA to use a linear mixed model:

```
gcta64 --bfile quantfamdata --grm-sparse GCTAsparsegrm \
--fastGWA-mlm --pheno phenos.txt --out LMMsparse_assoc
```

```
gcta64 --bfile quantfamdata --grm-sparse newsparsegrm \
--fastGWA-mlm --pheno phenos.txt --out LMMnewsparse_assoc
```

Let us now use R to check the QQ plots, Manhattan plots and Genomic control factors for these 4 sets of results. Start up R and use the following commands:

```
res4<-read.table("sparse_assoc.fastGWA", header=T)
res5<-read.table("newsparse_assoc.fastGWA", header=T)
res6<-read.table("LMMsparse_assoc.fastGWA", header=T)
res7<-read.table("LMMnewsparse_assoc.fastGWA", header=T)
```

```
head(res4)
head(res5)
head(res6)
```

```
head(res7)
```

```
chi<-(qchisq(1-res4$P,1))  
lambda=median(chi)/0.456  
lambda
```

```
chi<-(qchisq(1-res5$P,1))  
lambda=median(chi)/0.456  
lambda
```

```
chi<-(qchisq(1-res6$P,1))  
lambda=median(chi)/0.456  
lambda
```

```
chi<-(qchisq(1-res7$P,1))  
lambda=median(chi)/0.456  
lambda
```

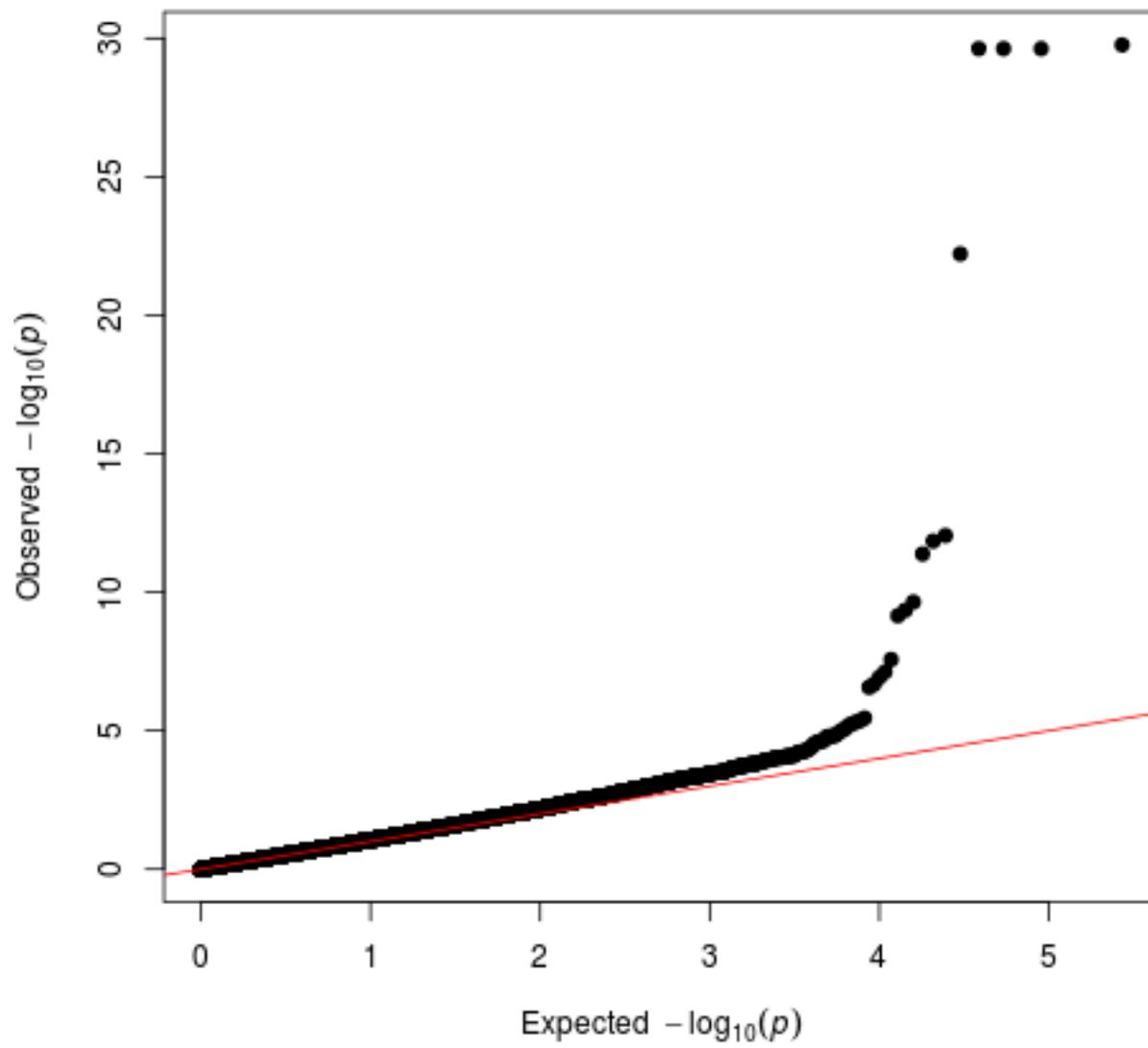
You should see that the linear mixed model (with a sparse GRM) does a better job at controlling for relatedness (giving lambda closer to 1.0) than just including 5 principal components.

Now continue in R to make the plots for these 4 analyses (which will be very similar to those you made previously):

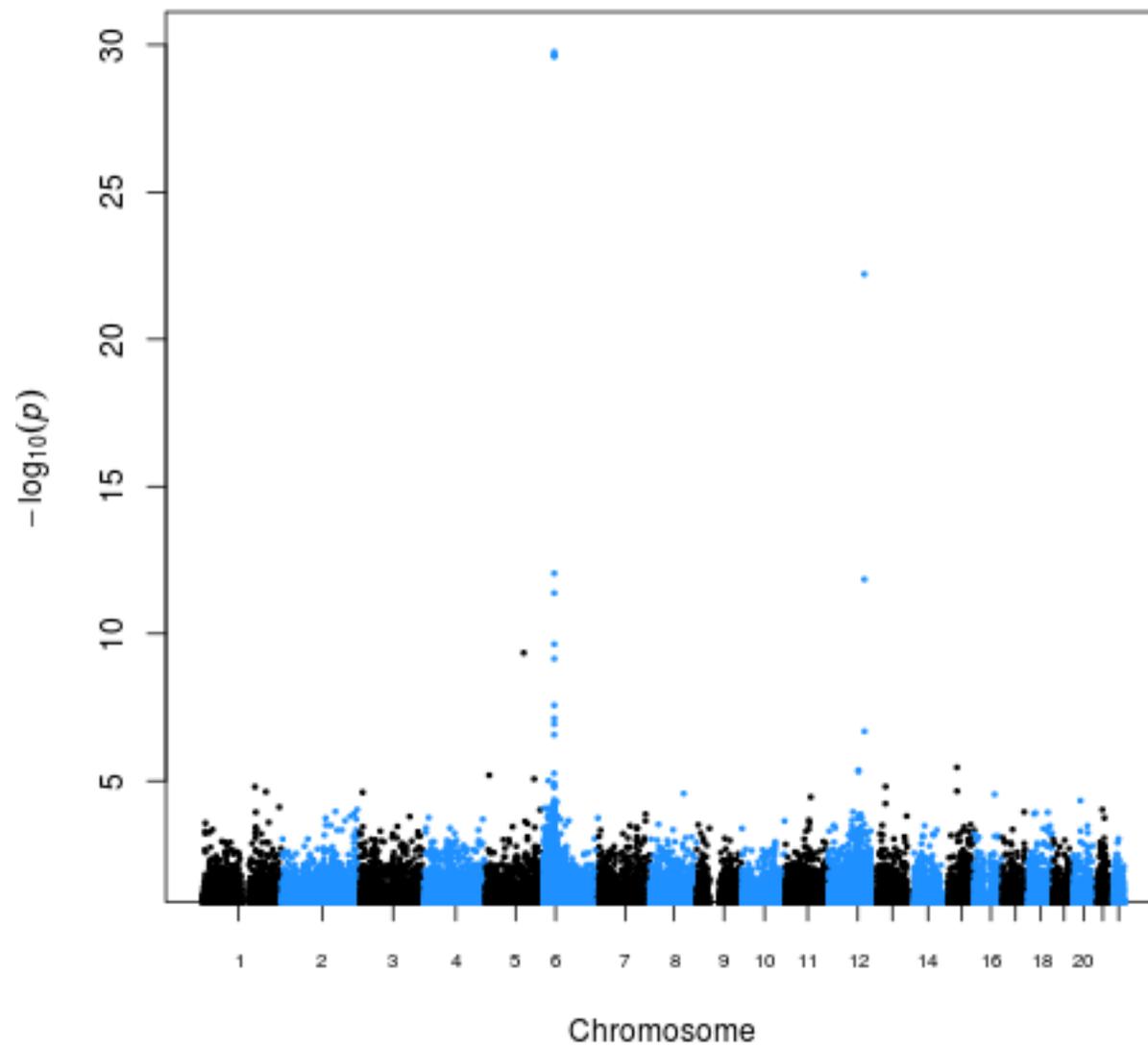
```
source("qqmanHJCupdated.R")
```

```
new4<-data.frame(res4$SNP, res4$CHR, res4$POS, res4$P)  
names(new4)<-c("SNP", "CHR", "BP", "P")
```

```
png("qq4.png")  
qq(new4$P)  
dev.off()
```

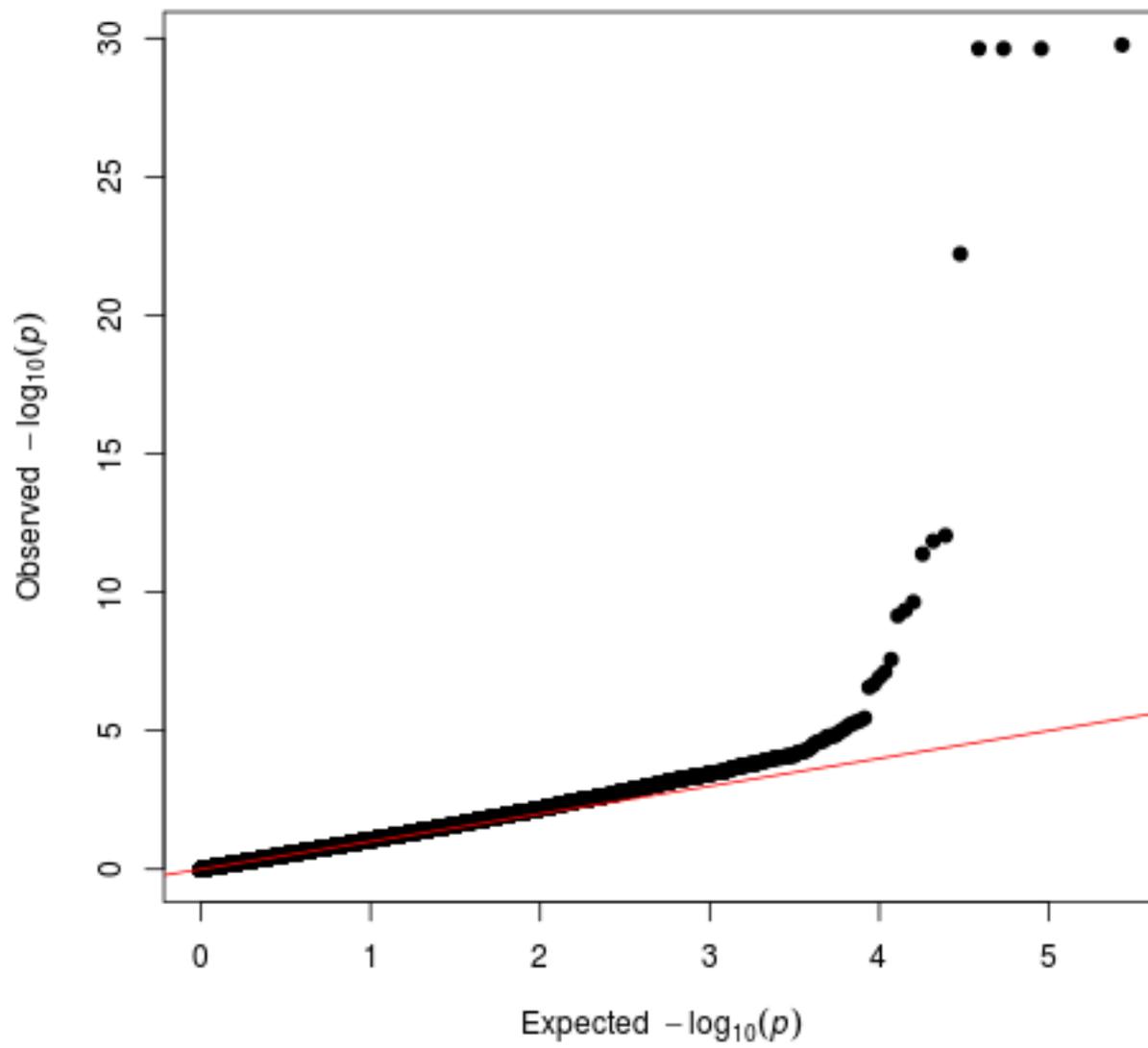


```
png("mh4.png")  
manhattan(new4, pch=20, suggestiveline=F, genomewideline=F, ymin=2,  
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)  
dev.off()
```

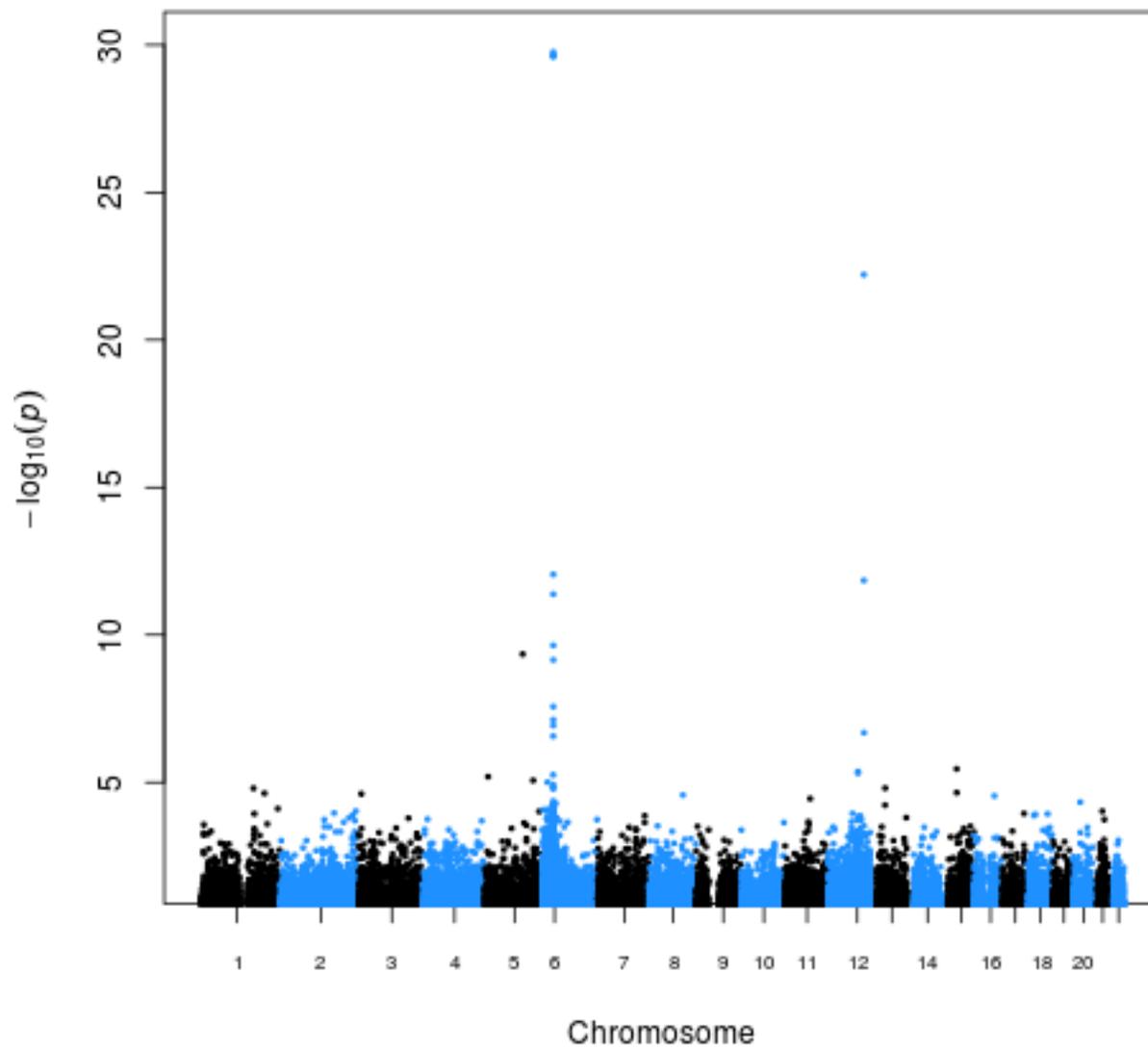


```
new5<-data.frame(res5$SNP, res5$CHR, res5$POS, res5$P)  
names(new5)<-c("SNP", "CHR", "BP", "P")
```

```
png("qq5.png")  
qq(new5$P)  
dev.off()
```

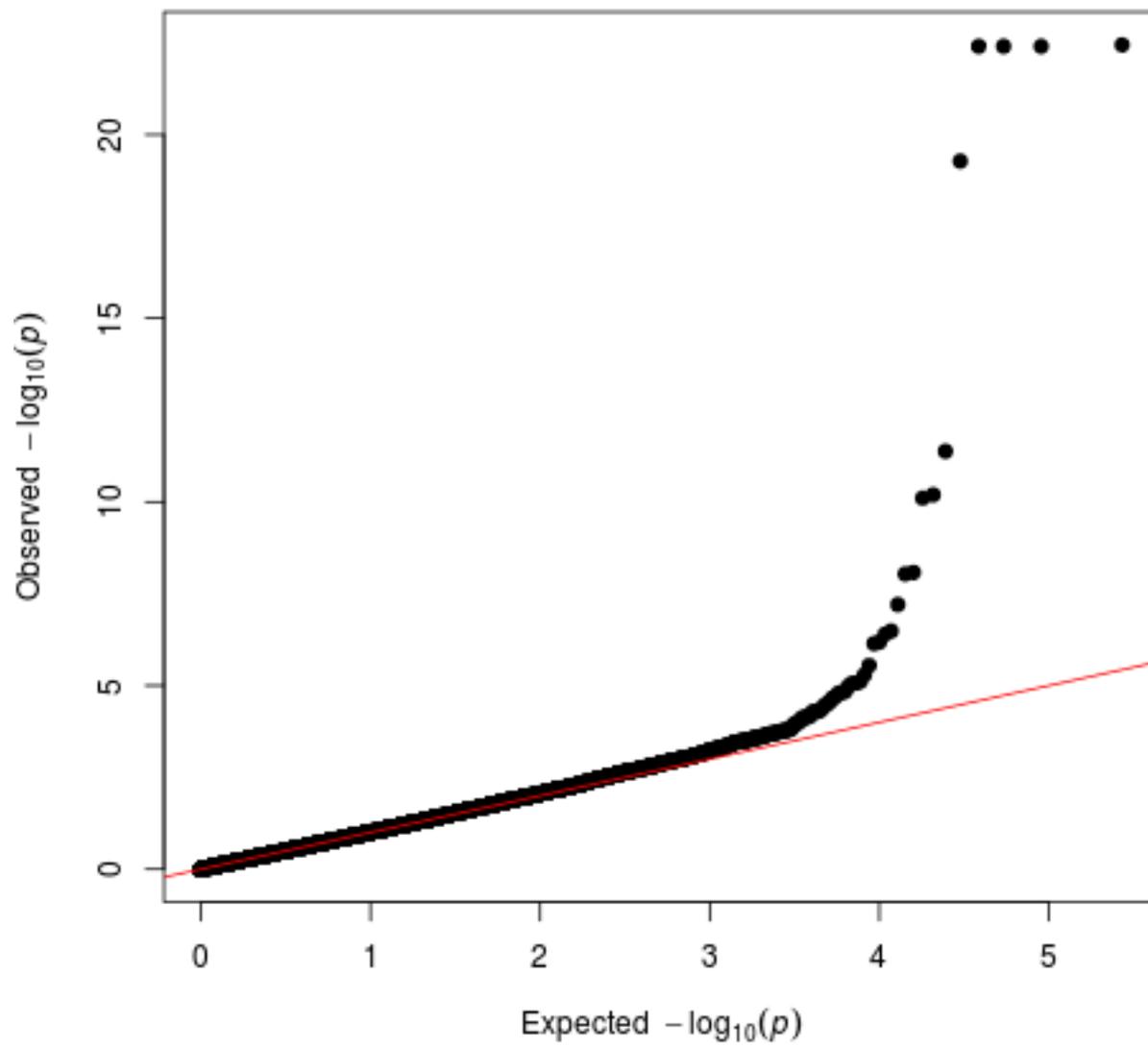


```
png("mh5.png")  
manhattan(new5, pch=20, suggestiveline=F, genomewideline=F, ymin=2,  
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)  
dev.off()
```

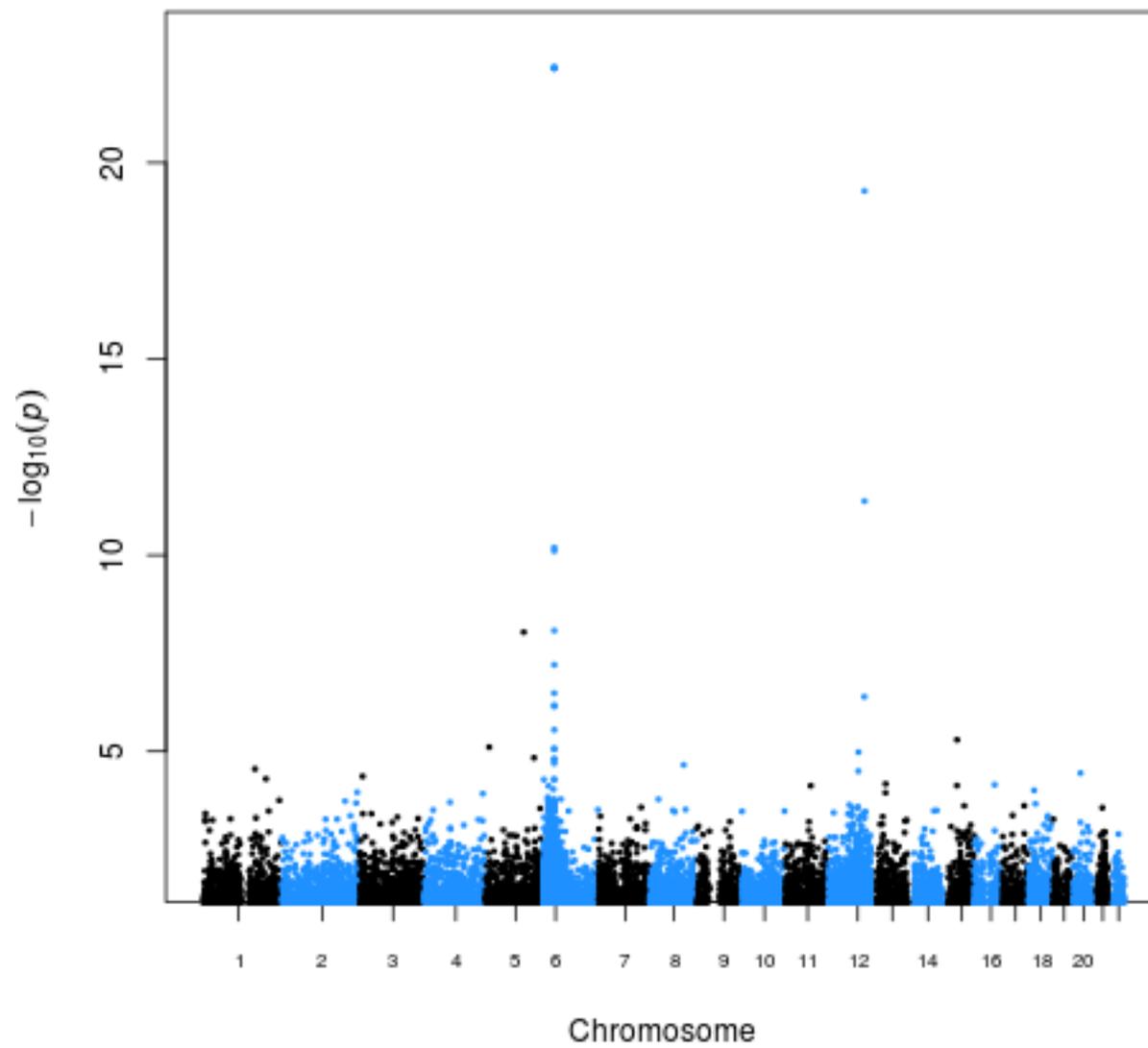


```
new6<-data.frame(res6$SNP, res6$CHR, res6$POS, res6$P)  
names(new6)<-c("SNP", "CHR", "BP", "P")
```

```
png("qq6.png")  
qq(new6$P)  
dev.off()
```

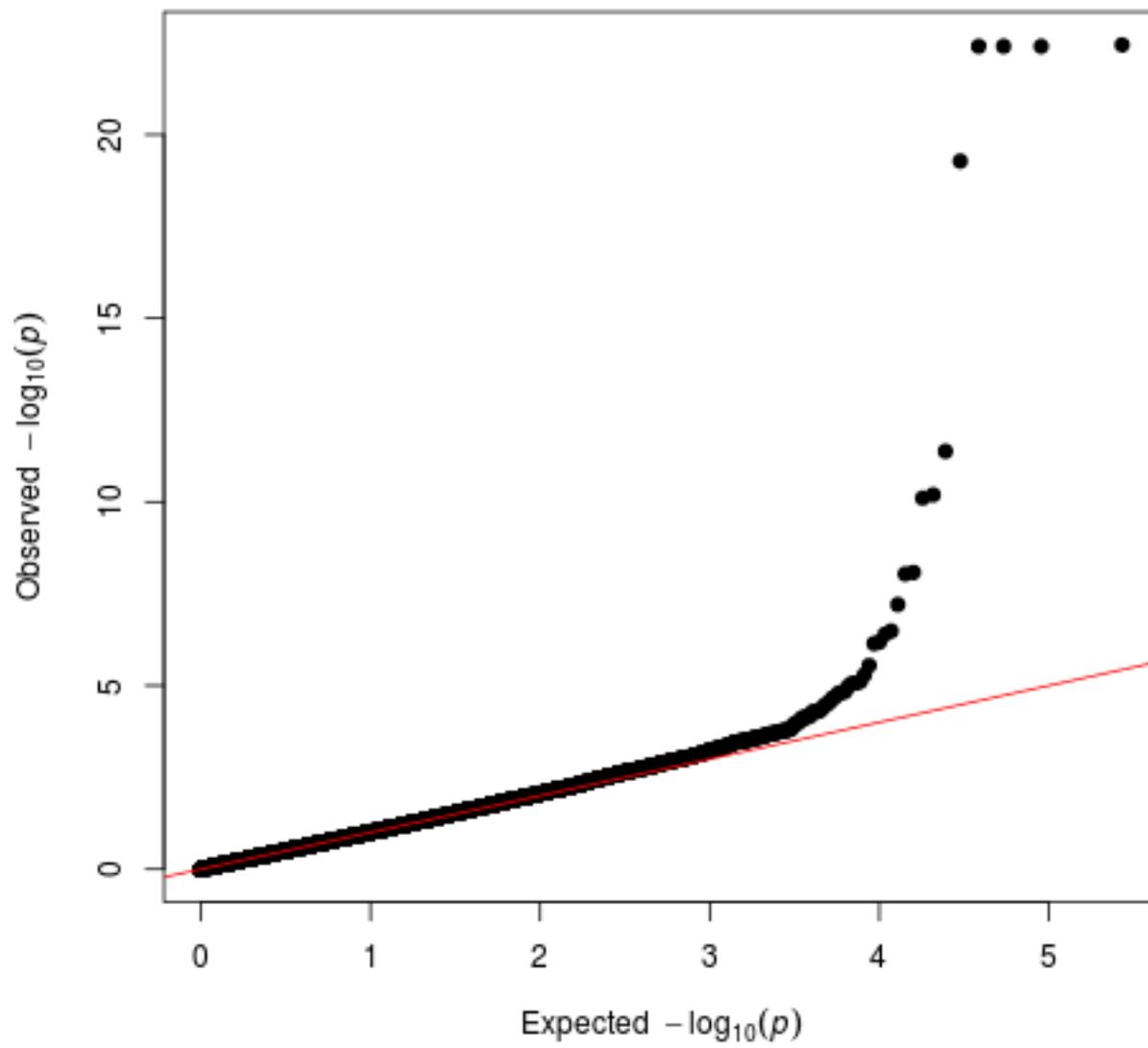


```
png("mh6.png")  
manhattan(new6, pch=20, suggestiveline=F, genomewideline=F, ymin=2,  
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)  
dev.off()
```

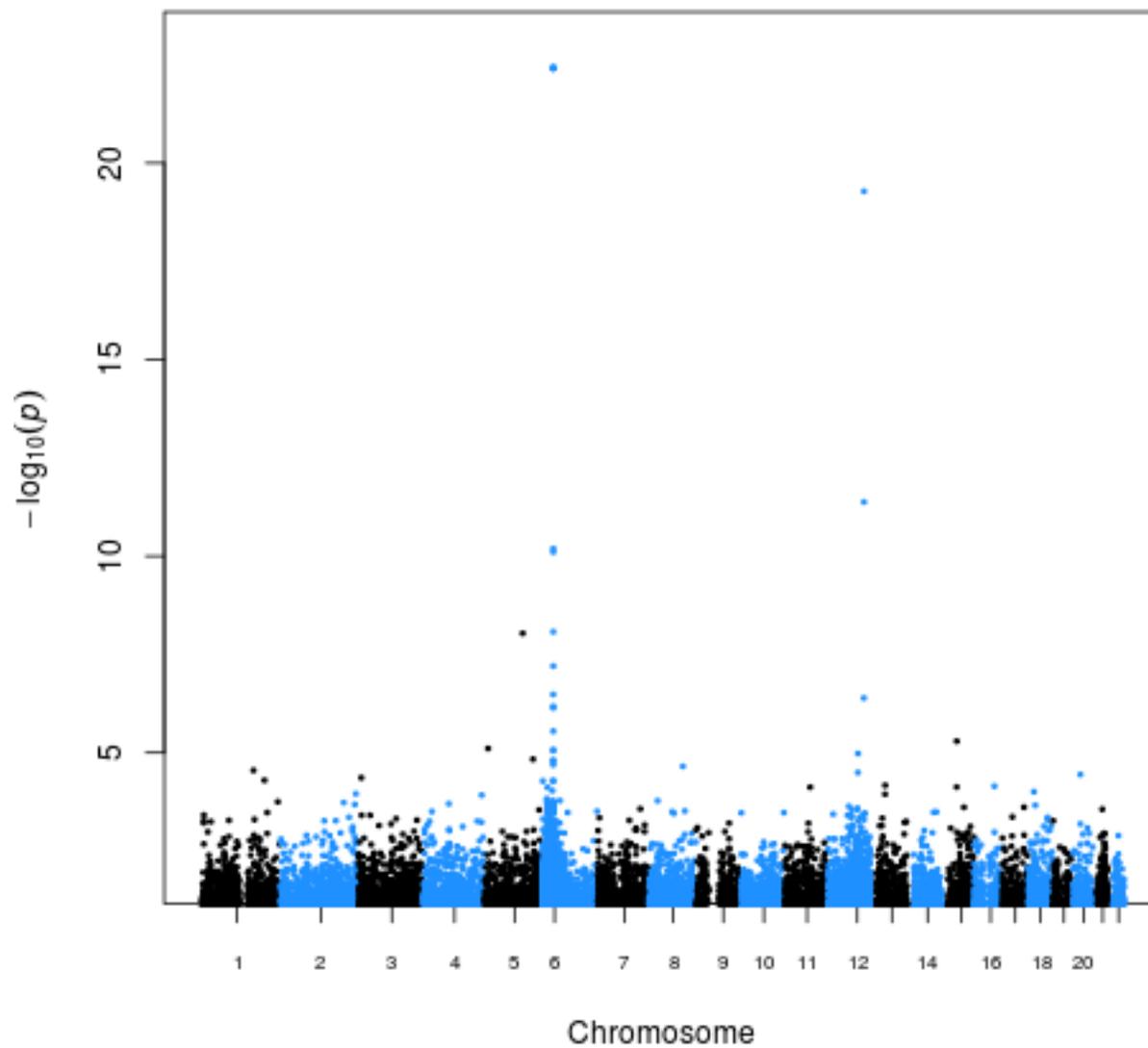


```
new7<-data.frame(res7$SNP, res7$CHR, res7$POS, res7$P)  
names(new7)<-c("SNP", "CHR", "BP", "P")
```

```
png("qq7.png")  
qq(new7$P)  
dev.off()
```



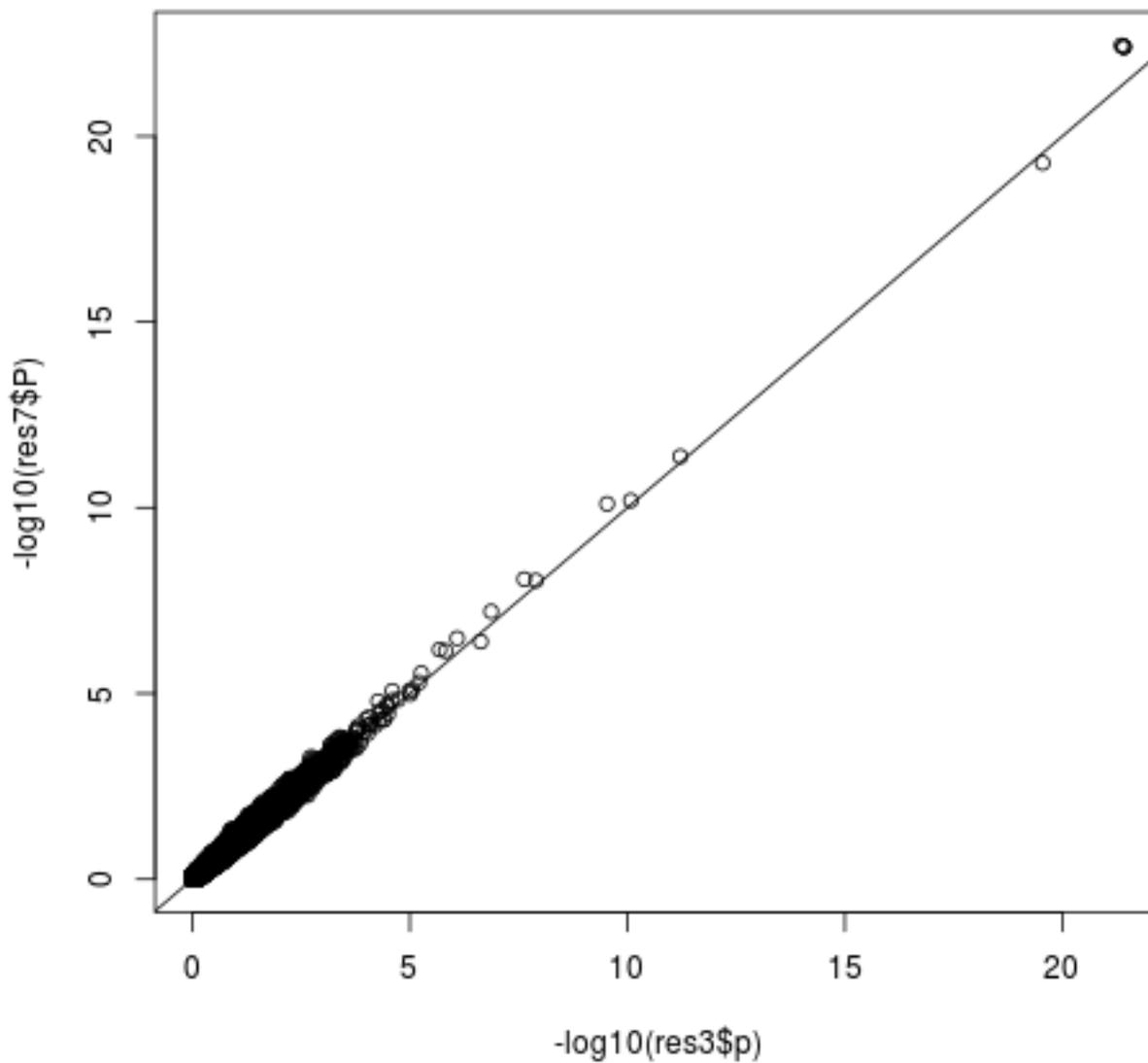
```
png("mh7.png")  
manhattan(new7, pch=20, suggestiveline=F, genomewideline=F, ymin=2,  
cex.x.axis=0.65, colors=c("black","dodgerblue"), cex=0.5)  
dev.off()
```



Finally we will check how the results from the linear mixed model (with a sparse GRM) from fastGWA compares to the original GCTA `--mlma` results:

```
res3<-read.table("GCTAresults.mlma", header=T)
res7<-read.table("LMMnewsparse_assoc.fastGWA", header=T)
head(res3)
head(res7)
```

```
png("compareGCTAfastGWA.png")
plot(-log10(res3$p), -log10(res7$p))
abline(0,1)
dev.off()
```



As you can see, the results are extremely similar.

Answers

How to interpret the output

Interpretation of the output is described in the step-by-step instructions. Please ask if you need help in understanding the output.

Comments

Other packages

Another package that can implement a similar analysis to GCTA is DISSECT

References

Yang et al. (2011) GCTA: A tool for genome-wide complex trait analysis.
American Journal of Human Genetics, 88:76-82.

Exercises prepared by: Heather Cordell

Checked by:

Programs used: R, GCTA

Last updated: 01/15/2021 14:53:55 01/15/2021 14:53:36

Interaction analysis using PLINK and CASSI

Overview

Purpose

In this exercise you will be performing association analysis and testing for interaction effects using case/control data.

Methodology

The methodology used includes logistic regression in PLINK and CASSI, as well as some related alternative approaches.

Program documentation

PLINK documentation:

PLINK has an extensive set of documentation including a pdf manual, a web-based tutorial and web-based documentation:

Original PLINK (1.07) (which has arguably clearer documentation):
<http://zzz.bwh.harvard.edu/plink/>

New PLINK (1.90) (which includes documentation on new additional features):
<https://www.cog-genomics.org/plink2>

CASSI documentation:

CASSI documentation is available from:

<http://www.staff.ncl.ac.uk/richard.howey/cassi/downloads.html>

Exercise

Data overview

The data consists of simulated genotype data at 100 SNP loci, typed in 2000 cases and 2000 controls. The data has been simulated in such a way that the first

five SNPs have some relationship with disease, whereas the remaining 95 SNPs have no effect on disease outcome.

The complication with these data is that SNPs 1 and 2 have been simulated in such a way that they show no marginal association with the disease: their association will only be visible when you look at both SNPs in combination. SNPs 3-5 have been simulated to only have an effect on disease when an individual is homozygous at all three of these loci. Although potentially this could lead to marginal effects at the loci, formally this corresponds to a model of pure interaction, with no main effects, at these 3 SNPs.

Appropriate data

Appropriate data for this exercise is genotype data for a set of linked or unlinked loci typed in a group of unrelated affected individuals (cases) and in a group of unaffected or randomly chosen individuals from the same population (controls).

All the programs will deal with much larger numbers of loci than the 100 SNPs considered here. PLINK, in particular, was specifically designed for the analysis of large numbers of loci e.g. generated as part of a genome-wide association study.

Instructions

Data format

The data for the 100 SNPs [simcasecon.ped](#) is in standard linkage pedigree file format, with columns corresponding to family id, subject id (within family), father's id, mother's id, sex (1=m, 2=f), affection status (1=unaffected, 2=affected) and one column for each allele for each locus genotype. Note that since this is case/control rather than family data, there is only one individual per family and everyone's parents are coded as unknown.

PLINK requires an additional map file [simcasecon.map](#) describing the markers (in order) in the pedigree file. The PLINK-format map file contains exactly 4 columns:

chromosome (1-22, X, Y or 0 if unplaced)
rs number or snp identifier
Genetic distance (morgans) (not often used - so can be set to 0)
Base-pair position (bp units)

Take a look at the data files, and check that you understand how the data is coded. Then (if necessary) save the files as .txt files to the appropriate directory (folder) on your computer.

Step-by-step instructions

1. Analysis in PLINK

Move to the directory where you have saved the data files.

To carry out a basic association analysis in PLINK, type

```
plink --ped simcasecon.ped --map simcasecon.map --assoc
```

Here the `--ped xxxx` command tells PLINK that the name of the pedigree file is `xxxx` and the `--map yyyy` command tells PLINK that the name of the map file is `yyyy`. The `--assoc` command tells PLINK to perform a basic allele-based chisquared association test.

PLINK outputs some useful messages (you should always read these to make sure they match up with what you expect!) and outputs the results to a file `plink.assoc`.

Take a look at the file `plink.assoc` (e.g. by typing `more plink.assoc`). For each SNP the following columns of results are reported:

| | |
|--------------|--|
| 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) |

Does there appear to be evidence of association at any of the five "true" loci? What about the 95 null loci?

Try performing a genotype-based (rather than an allele-based) analysis in PLINK and take a look at the results by typing the following 3 commands:

```
plink --ped simcasecon.ped --map simcasecon.map --model  
head -1 plink.model  
grep GENO plink.model
```

Again, does there appear to be evidence of association at any of the five "true" loci? What about the 95 null loci?

To test for pairwise epistasis in PLINK, the fastest option is to use the `--fast-epistasis` command:

```
plink --ped simcasecon.ped --map simcasecon.map --fast-epistasis
```

Formally, this tests whether the OR for association between two SNPs differs between cases and controls, which can be shown to approximate a logistic regression based test of interaction between the SNPs. Results can be found in the file `plink.epi.cc`. Only pairwise interaction tests with $p \leq 0.0001$ are reported (otherwise, for genome-wide studies, there would be too many results to report, given the large number of pairwise tests performed).

Take a look at the file `plink.epi.cc`. You should find a very significant interaction between SNPs 1 and 2, and a less significant interaction between SNPs 15 and 77. Since this is simulated data, we know that this less significant result is a false positive.

A more powerful test for SNPs that are not in LD with one another (i.e. that are

not too close to one another, in terms of their genomic location) is to additionally use the `--case-only` option:

```
plink --ped simcasecon.ped --map simcasecon.map --fast-epistasis --case-only
```

Results can be found in the file `plink.epi.co`. Again only pairwise interaction tests with $p \leq 0.0001$ are reported. You should again find a very significant interaction between SNPs 1 and 2 (even more significant than previously, owing to the increased power with a case-only test).

A problem with the `--fast-epistasis` test is that it can be affected by LD between the SNPs (although only the case-only test is seriously affected). A more accurate test is to carry out logistic regression by using the slower `--epistasis` command:

```
plink --ped simcasecon.ped --map simcasecon.map --epistasis
```

Results can again be found in the file `plink.epi.cc` (which will now have been overwritten). You can see that again the interaction between SNPs 1 and 2 remains highly significant ($p=1.22E-63$), together with just one other (false positive) interaction between SNPs 15 and 77.

Since the `--epistasis` option is slower, but most accurate, for genome-wide studies it might be sensible to first to screen for interactions using the `--fast-epistasis` command, but then confirm any findings using the `--epistasis` command on the smaller set of detected SNPs.

2. Analysis in CASSI

We will also compare our PLINK results with those obtained using the CASSI program, which implements a variety of tests including linear and logistic regression, and an improved Joint Effects (JE) test of pairwise interaction as described in Ueki and Cordell (2012). First we need to convert our data to PLINK binary format:

```
plink --ped simcasecon.ped --map simcasecon.map --make-bed --out simbinary
```

This should create PLINK binary format files `simbinary.bed`, `simbinary.bim` and `simbinary.fam`. Then we use the CASSI program with the input file `simbinary.bed` to perform pairwise interaction tests at all pairs of loci. (By default, only those pairs of SNPs showing interaction with a p -value < 0.0001 are output, though this can be changed if desired).

We start by using logistic regression. The logistic regression test in CASSI is essentially the same as the `--epistasis` test in PLINK, except that CASSI uses a likelihood ratio test rather than the asymptotically equivalent Wald (?) test used by PLINK. CASSI also has the advantage of allowing covariates into the analysis, if desired.

```
cassi -lr -i simbinary.bed
```

Take a look at the output file `cassi.out`. The most important columns are the first 4 columns (listing the SNP numbers/names) and the last 4 columns listing the log odds ratio, its standard error, the likelihood ratio chi-squared test statistic and its p -value. It can be quite hard to work out which column is which, so we suggest you start up R by typing

R

and then read in and look at the results by typing

```
results<-read.table("cassi.out", header=T)
results
```

You can see that SNPs 1 and 2 show a very strong pairwise interaction ($p=5.94E-72$), which is actually a bit more significant than the result from PLINK ($p=1.22E-63$). We also still detect the false positive interaction between SNPs 15 and 77.

Now try using the Joint Effects (JE) test, telling CASSI to use the output filename `cassiJE.out`

```
cassi -je -o cassiJE.out -i simbinary.bed
```

Take a look at the output file `cassiJE.out`. The most important columns are the first 4 columns (listing the SNP numbers/names) and the last 4 columns listing the case/control and case-only interaction test chi-squareds and p-values. Again it can be quite hard to work out which column is which, so we suggest you read in and look at the results in R:

```
resultsJE<-read.table("cassiJE.out", header=T)
resultsJE
```

You can see that SNPs 1 and 2 show a very strong pairwise interaction (Case-Con test p-value $JE_CC_P=1.67e-129$; Case-Only test p-value $JE_CO_P=1.71e-274$). Interestingly we also detect, albeit at lower significance levels, the (true) pairwise interactions between SNPs 3 and 4 and between SNPs 4 and 5. We also detect two false positive interactions, between SNPs 15 and 77, and between SNPs 31 and 100.

Answers

Interpretation of output

Answers and interpretation of the output are described in the step-by-step instructions. Please ask if you need help in understanding the output for any specific test.

Comments

Advantages/disadvantages

PLINK and CASSI are designed for genome-wide studies, allowing the inclusion of many thousands of markers. Analysis in a standard statistical package does not generally allow so many markers, but may have some advantage of allowing a lot of extra flexibility with regards to the models and analyses performed e.g. it easy

to include additional predictor variables such as environmental factors, gene-environment interactions etc. However, you are required to know or learn how to use the package in order to gain that extra flexibility, and to produce reliable results.

Study design issues

With case/control data it is relatively easy to obtain large enough sample sizes to detect small genetic effects. However, detection of interactions generally requires much larger sample sizes.

Other packages

Logistic regression analysis for detection of interactions can be performed in most statistical packages such as R, Stata, SAS, SPSS. Alternative Bayesian Epistasis mapping approaches are available in the BEAM (Zhang et al. 2007; Zhang 2011) or BIA software packages.

Several packages are available for implementing different data-mining and machine-learning approaches for detecting interactions or detecting association allowing for interaction. See Cordell (2009) and other references below for more details.

References

Cordell HJ (2009) Genome-wide association studies: Detecting gene-gene interactions that underlie human diseases. *Nat Rev Genet* 10(6):392-404.

Y Chung and S Y Lee and R C Elston and T Park (2007) Odds ratio based multifactor-dimensionality reduction method for detecting gene-gene interactions. *Bioinformatics* 23:71-76.

L W Hahn and M D Ritchie and J H Moore (2003) Multifactor dimensionality reduction software for detecting gene-gene and gene-environment interactions *Bioinformatics* 19:376-382.

Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MAR, Bender D, Maller J, Sklar P, de Bakker PIW, Daly MJ & Sham PC (2007) PLINK: a toolset for whole-genome association and population-based linkage analysis. *American Journal of Human Genetics*, 81:559-575.

Ritchie MD, Hahn LW, Roodi N, Bailey LR, Dupont WD, Parl FF and Moore JH (2001) Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *Am J Hum Genet* 69:138-147.

Ueki M, Cordell HJ (2012) Improved statistics for genome-wide interaction analysis. *PLoS Genetics* 8(4):e1002625.

Zhang Y, Liu JS (2007) Bayesian inference of epistatic interactions in case-control studies. *Nat Genet* 39:1167-1173.

Zhang Y (2011) A novel Bayesian graphical model for genome-wide multi-SNP association mapping. *Genet Epidemiol* 36: 36-47.

Exercises prepared by: Heather Cordell

Checked by:

Programs used: PLINK, CASSI

Last updated: 01/17/2020 12:35:48

Sample Size Calculations - Cochran-Armitage Test for Trend

Copyrighted © 2022 Suzanne M. Leal

Webpage for the exercises:

http://csg.sph.umich.edu/abecasis/cats/gas_power_calculator/index.html

<http://ihg.helmholtz-muenchen.de/cgi-bin/hw/power2.pl>

<http://zzz.bwh.harvard.edu/gpc/cc2.html>

Question 1

For a complex disease study, you plan to collect 35,000 cases and 70,000 controls and wish to know if this is a sufficient sample size to detect associations with disease susceptibility loci. The disease has a population prevalence of 5%. You wish to estimate the power for a genotypic relative risk of 1.2 and a disease allele frequency of 0.02. What is the power for $\alpha=5 \times 10^{-8}$ under a multiplicative model ($\gamma_2 = \gamma_1^2$) a.) _____ and dominant model ($\gamma_2 = \gamma_1$) b.) _____?

Question 2

For your study, you hypothesize that you will try to replicate associations for 100 variants that are in linkage equilibrium and you want to reject the null hypothesis using a p-value of 0.05. What is the Bonferroni correction you should use a.) _____. Determine what your power would be if you used a Bonferroni correction to control for the Family Wise Error Rate (FWER) for testing 100 variants. Using the parameters provided in question 1 but for a sample size of 20,000 cases and 20,000 controls what is the power under the multiplicative model b.) _____ and under a dominant model c.) _____?

Question 3

You determine that you can ascertain 50,000 cases and 50,000 controls what is the power using the same parameters as described in question 1 for the multiplicative model _____ and dominant model _____?

Question 4

The power of the Cochran-Armitage test for trend is dependent on the underlying genetic model. Using the parameters from question 1 which of the following underlying genetic models: multiplicative ($\gamma_2 = \gamma_1^2$), additive ($\gamma_2 = 2\gamma_1 - 1$), dominant ($\gamma_2 = \gamma_1$) or recessive ($\gamma_1 = 1$) would you predict to be the most powerful a.) _____ and least powerful b.) _____?

Question 5

For study design with equal numbers of cases and controls a genotype relative risk of 1.5 under a recessive model for a disease with a population prevalence of 0.05 and disease allele frequency of 0.1. How many cases a.) _____ and controls b.) _____ should you ascertain for $\alpha=5.0 \times 10^{-8}$ and $1-\beta=0.80$? *Use power2 or Genetic Power Calculator, GAS power cannot calculate for more than 100,000 cases.

Question 6

You are performing a rare variant association study and you assume that that cumulative frequency of the causal variants in your gene region is 0.01 with every variant having an effect size of 1.4. The disease you are studying has a prevalence of 5%. For a study with 0.8 power and an $\alpha=2.5 \times 10^{-6}$ under a dominant model for equal numbers of cases and controls what is the total sample size a.) _____ do you need to ascertain. What is the total sample size b.) _____ you need to ascertain if the cumulative frequency of causal variants is only 0.005?

Question 7

You are performing a study using the UK Biobank and for your phenotype of interest you have 50,000 cases and 100,000 controls. For a disease with 10% prevalence, disease allele frequency of 0.01, where each variant has an effect size of 1.2 under a dominant model what would be the power for an aggregate test where the cumulative allele frequency is 0.01 _____ and a single variant test _____? Clue use the appropriate alpha for each test.

Question 8

Using have a replication sample of 50,000 cases and 50,000 controls and you plan to try to replicate 15 genes and 100 variants. Using the same parameters as in question 7 what would be your power to replicate a.) _____? Note for alpha use a Bonferroni correction.

Question 9

For the above power calculations, you have been using the relative risk which only approximates the odds ratio when a.) _____? You are performing a power calculation for a case control study for a disease/variant frequency of 0.01. You use a dominant model and a gamma of 1.2 for a disease with a prevalence for 0.2. What is the odds ratio for which the power calculations are being performed b.) _____? *Use Genetic Power Calculator – information not provided by GAS or Power2.

ANSWERS

1. a.) 0.702 b.) 0.654
2. a.) 5.0×10^{-4} b.) 0.690 c.) 0.657
3. a.) 0.798 b.) 0.755
4. a.) multiplicative b.) recessive
5. a.) 170,910 b.) 170,910
6. a.) ~43,000 b.) ~84,300
7. a.) 0.73 b.) 0.45 Hint: use $\alpha=5 \times 10^{-8}$ for single variant test and $\alpha=2.5 \times 10^{-6}$ for the aggregate test
8. a.) 0.87 (Hint: use $\alpha=4.3 \times 10^{-4}$)
9. a.) only for disease with low prevalence does the relative risk does not estimate the odds ratio b.) 1.26

Variant Annotation and Functional Prediction

Copyrighted © 2022 Isabelle Schrauwen and Suzanne M. Leal

This exercise touches on several functionalities of the program ANNOVAR to annotate and interpret candidate genetic variants associated with disease, identified through next-generation sequencing methods, imputation or genotyping. When variants are identified to be associated with disease, a common strategy is to perform multiple *in silico* analyses to predict whether they potentially have an impact on gene function.

More information and a detailed guide on installation of ANNOVAR can be found here: <http://annovar.openbioinformatics.org/en/latest/>. ANNOVAR has three main annotation types to help evaluate variants:

[1] **Gene-based annotation:** This annotation annotates variants in respect to their effect on genes (RefSeq genes, UCSC genes, ENSEMBL genes, GENCODE genes, AceView genes) and also outputs the effect of the mutation on the protein in standard HGVS nomenclature (if an effect is predicted).

[2] **Region-based annotation:** With this annotation you can identify variants in specific genomic regions (i.e. conserved regions, predicted transcription factor binding sites, segmental duplication regions, GWAS hits, database of genomic variants, DNase I hypersensitivity sites, ENCODE H3K4Me1/H3K4Me3/H3K27Ac/CTCF sites, ChIP-Seq peaks, RNA-Seq peaks, or many other annotations on genomic intervals).

[3] **Filter-based annotation:** Identify variants that are documented in specific frequency databases (dbSNP, Genome Aggregation Consortium, etc) or functional effect prediction databases (PolyPhen, MutationTaster, FATHMM, etc). For example, find intergenic variants with a CADD c-score >20.

In this exercise, we will evaluate *APOC3* variants by annotating a *.vcf* file (*APOC3.vcf*). Variants in *APOC3* are associated with apoC-III protein levels, triglycerides levels, and coronary heart disease. Previous studies suggested that lifelong deficiency of apoC-III has a cardioprotective effect. When rare variant association tests are performed, variants are often analyzed as a group; and when an association has been found which has been replicated it is not necessarily true that all tested variants are causal. However, for low variant frequencies it is often not possible to test individual variants for an association with a trait. Therefore, bioinformatics tools are often used to predict which variants are likely to be functional and therefore could be involved in trait etiology. For this exercise, six variants in *APOC3* were selected for annotation as an example.

First of all, once you are logged in into dockerhub go to the */shared* directory where the datafiles for this exercise are located by typing:

```
$ /home/shared/functional_annotation
```

The `table_annoar.pl` in ANNOVAR command accepts VCF files. Type in `table_annoar.pl` to learn about the annotation options (Tip: add Annovar to your PATH to be able to use this command in any directory). More info on VCF processing and left-normalization for indels can be found here:

<http://annovar.openbioinformatics.org/en/latest/articles/VCF/>. Note, ANNOVAR can also accept compressed `.vcf.gz` files.

`$ table_annoar.pl`

A. Gene-based annotation: Using Ensembl, RefSeq and UCSC Genome Browser First, we will evaluate the location of these variants in *APOC3*. We will use the Gene-based annotation function, which annotates variants to coding and non-coding genes and indicates the amino acids that are affected. Users can flexibly use RefSeq, UCSC genome browser, ENSEMBL, GENCODE, AceView, or other gene definition databases.

Let us first annotate our variants with the standard refGene database (NCBI):

`$ table_annoar.pl APOC3.vcf humandb/ -buildver hg19 -out APOC3_Gene.vcf -remove -nastring . -protocol refGene -operation g -vcfinput`

Each of the options in the command line is preceded with '-' (again, more information can be found by typing `table_annoar.pl`). The `-operation` option defines the type of annotation, `g`=gene-based; `f`=filter-based and `r`=region-based.

The annotated output file is written to `APOC3_Gene.vcf.hg19_multianno.txt`
Results are also written in VCF format: `APOC3_Gene.vcf.hg19_multianno.vcf`

Now look at the resulting table:

`$ cat APOC3_Gene.vcf.hg19_multianno.txt`

Question 1: Of the six *APOC3* variants that were analyzed how many are exonic?

The output txt file is also easy to view in excel. Open the file in Excel and select "tab-delimited" when opening the file. To filter data, click the "data" tab at the menu bar, then click the "Filter" button.

| | A | B | C | D | E | F | G | H | I | J |
|---|-----|-----------|-----------|-----|-----|-----------|-----------|--------------------|------------|----------------------------------|
| 1 | Chr | Start | End | Ref | Alt | Func.refG | Gene.refG | GeneDetail.refGer | ExonicFur | AAChange.refGene |
| 2 | 11 | 116701353 | 116701353 | C | T | exonic | APOC3 | . | stopgain | APOC3:NM_000040:exon2:c.C55T;p.f |
| 3 | 11 | 116701560 | 116701560 | G | A | exonic | APOC3 | . | nonsynonym | APOC3:NM_000040:exon3:c.G127A;p |
| 4 | 11 | 116701613 | 116701613 | G | T | splicing | APOC3 | NM_000040:exon3:c. | . | . |

Notice all variants are automatically reported following the HGVS nomenclature. Variants are categorized based on these groups:

| | |
|------------|---|
| exonic | variant overlaps a coding region |
| splicing | variant is within 2-bp of a splicing junction (use <code>-splicing_threshold</code> to change this) |
| ncRNA | variant overlaps a transcript without coding annotation in the gene definition |
| UTR5 | variant overlaps a 5' untranslated region |
| UTR3 | variant overlaps a 3' untranslated region |
| intronic | variant overlaps an intron |
| upstream | variant overlaps 1-kb region upstream of transcription start site |
| downstream | variant overlaps 1-kb region downstream of transcription end site (use <code>-neargene</code> to change this) |
| intergenic | variant is in intergenic region |

Next we will annotate using three main databases: Ensembl, RefSeq and UCSC Known Gene, and change boundaries of splice variants (default is 2 bp from splice site, let's set this to 12 bp):

```
$ table_annovar.pl APOC3.vcf humandb/ -buildver hg19 -out APOC3_Gene.vcf -remove -nastring . -protocol refGene,knownGene,ensGene -operation g,g,g -arg '-splicing 12 -exonicsplicing','-splicing 12 -exonicsplicing','-splicing 12 -exonicsplicing' -vcfinput
```

This file has many columns, view select columns with `awk` (depending on which columns you are interested in seeing) using the below command or alternatively, you can open the file in excel:

```
$ awk -F'\t' '{print $1,$2,$6,$7,$8,$9,$10}' APOC3_Gene.vcf.hg19_multianno.txt
```

Question 2: What has changed compared to the initial annotation (hint: the splicing thresholds were changed)

B. Region based annotation

Another functionality of ANNOVAR is to annotate regions associated with variants: For example, DNase I hypersensitivity sites, ENCODE regions, predicted transcription factor binding sites, GWAS hits, and phastCons 46-way alignments to annotate variants that fall within conserved genomic regions as shown here:

```
$ table_annovar.pl APOC3.vcf humandb/ -buildver hg19 -out APOC3_Region.vcf -remove -nastring . -protocol phastConsElements46way -operation r -vcfinput
```

Note `$ cat resultingfile.txt` here to view your results in the terminal or use `awk` to print certain columns of interest. Only conserved regions will display a score (maximum 1000) and a name.

Question 3: Which of the *APOC3* variants are within a conserved genomic region?

We can also identify variants that were previously reported to be associated with diseases or traits in genome-wide association studies:

```
$ table_annoar.pl APOC3.vcf humandb/ -buildver hg19 -out APOC3_Region.vcf -remove -nastring . -protocol gwasCatalog -operation r -vcfinput
```

The gwasCatalog track in ANNOVAR is not fully comprehensive, but will point you towards major associations.

Question 4. Which of these variants are reported in the ANNOVAR GWAS catalog, and what has it been associated with ?

The region-based annotation can be used to evaluate pathogenicity of certain regions, especially non-coding regions. In addition to the examples above, here are some other useful databases in region-annotation:

- wgRna: variants disrupting microRNAs and snoRNAs
- targetScanS: Identify variants disrupting predicted microRNA binding sites
- tfbsConsSites: Transcription factor binding sites
- The Encyclopedia of DNA Elements (ENCODE): A comprehensive parts list of functional elements in the human genome, including elements that act at the protein and RNA levels, and regulatory elements that control cells and circumstances in which a gene is active. Several annotations are possible depending on your interests and can be found here: <http://annoar.openbioinformatics.org/en/latest/user-guide/region/>

C. Filter based annotation

Filter based annotation includes annotation to certain databases, such as gnomAD, dbSNP, and prediction programs to evaluate pathogenicity. There are many options, but we selected these as particularly helpful for complex diseases:

```
$ table_annoar.pl APOC3.vcf humandb/ -buildver hg19 -out APOC3_Filter.vcf -remove -nastring . -protocol gnomad_genome,gnomad_exome,popfreq_max_20150413,gme,avsnp150,dbnsfp33a,dbs csnv11,cadd13gt20,clinvar_20170905,gwava -operation f,f,f,f,f,f,f,f,f,f -vcfinput
```

This command will annotate the following:

- gnomAD genome

- gnomad_exome (includes ExAC)
- popfreq_max_20150413: A database containing the maximum allele frequency from 1000G, ESP6500, ExAC and CG46 (use popfreq_all_20150413 to see all allele frequencies)
- dbSNP150
- gme: Great Middle East allele frequencies from the GME variome project
- dbnsfp33a: whole-exome SIFT, PolyPhen2 HDIV, PolyPhen2 HVAR, LRT, MutationTaster, MutationAssessor, FATHMM, PROVEAN, MetaSVM, MetaLR VEST, M-CAP, CADD, GERP++, DANN, fathmm-MKL, Eigen, GenoCanyon, fitCons, PhyloP and SiPhy scores.
- dbscSNV version 1.1: for splice site prediction by AdaBoost and Random Forest
- Genome-wide CADD version 1.3 score>20
- clinvar_20170905: CLINVAR database with Variants of Clinical Significance
- gWAVA: Prioritization of noncoding variants by integrating various genomic and epigenomic annotations.

Build your own filter annotations here:

<http://annovar.openbioinformatics.org/en/latest/user-guide/download/>

We can split these annotations up into several categories what will help to evaluate pathogenicity:

1. Allele frequency databases

1.a Allele frequency in control populations

Evaluating the frequency of a possible causal/associated variant in several control population is important in any disease/trait. The use of these databases might be different depending on the prevalence of your disease of interest, but these databases can provide valuable information on the rarity of variants and population-specific variants. If a variant is rare in your population, it is encouraged to check whether it might be more frequent in other populations, which might alter your conclusions on pathogenicity:

- i. **gnomAD and ExAC databases:** The [Genome Aggregation Database](#) (gnomAD) and the [Exome Aggregation Consortium](#) (ExAC) are a coalition of investigators seeking to aggregate and harmonize genome and exome sequencing data from a wide variety of large-scale sequencing projects. The ExAC dataset contains spans 60,706 unrelated individuals sequenced as part of various disease-specific and population genetic studies. gnomAD spans 123,136 exome sequences and 15,496 whole-genome sequences from unrelated individuals and includes ExAC data. For both databases individuals known to be affected by severe pediatric disease are removed, as well as their first-degree relatives, so this data set should aid as a useful reference set of allele frequencies for severe disease studies - however, note that some individuals with severe disease may still be included in the data set.
- ii. **BRAVO:** Genome sequencing variants of 62,784 individuals sequenced for NHLBI's TOPMed program, to enhance the understanding of fundamental biological processes that underlie heart, lung, blood and sleep disorders. Currently not implemented in ANNOVAR yet, can be found here: <https://bravo.sph.umich.edu/freeze5/hg38/>.

- iii. **GME database:** The Greater Middle East (GME) Variome Project (<http://igm.ucsd.edu/gme/>) is aimed at generating a coding base reference for the countries found in the Greater Middle East. This dataset is especially useful when dealing with Mendelian families from the Middle East. Although these individuals are not a random sample, they were ascertained as a wide variety of distinct phenotypes such that cohort-specific effects are not expected to bias patterns of variation. For the final filtered set, primarily healthy individuals from families were selected, and wherever possible, removed from datasets the allele that brought the family to medical attention, leaving 1,111 high-quality unrelated individuals.
- iv. **1000G database:** The [1000 Genomes Project](#) ran between 2008 and 2015, creating a public catalogue of human variation and genotype data. Phase 3 includes 26 different populations, and might be useful when interested in population specific variation.
- v. **ESP6500:** The [NHLBI GO Exome Sequencing Project \(ESP\)](#) includes 6,503 samples drawn from multiple cohorts and represents all of the ESP exome variant data. In general, ESP samples were selected to contain deeply phenotyped individuals, the extremes of specific traits (LDL and blood pressure), and specific diseases (early onset myocardial infarction and early onset stroke), and lung diseases. This dataset contains a set of 2,203 African-Americans and 4,300 European-Americans unrelated individuals, totaling 6,503 samples (13,006 chromosomes).
- vi. **CG46:** CG46 database compiled from unrelated individuals sequenced by the Complete Genomics platform.

1.b Allele frequencies in disease populations

- vii. **Clinvar:** ClinVar is a freely accessible, public archive of reports of the relationships among human variations and phenotypes hosted by the National Center for Biotechnology Information (NCBI) and funded by intramural National Institutes of Health (NIH) funding. Although this database is mainly used for Mendelian disease variants, several rarer variants with a decent effect size in more complex disorders can be found in here as well, such as *APOC3*.

Let us examine if one of our variants we just annotated is in the Clinvar database:

```
$ awk -F'\t' '{print $1,$2,$103,$104}' APOC3_Filter.vcf.hg19_multianno.txt
```

Question 5: Is one of the variants reported as ‘pathogenic’ in Clinvar? If yes, which variants and which phenotype has been associated with these variants?

Next, look at the gnomAD overall exome and genome frequencies in 123,136 individuals for our variants, and specific exome populations:

```
$ awk -F'\t' '{print $1,$2,$6,$14}' APOC3_Filter.vcf.hg19_multianno.txt
```

```
$ awk -F'\t' '{print $1,$2,$15,$16,$17,$18,$19,$20,$21,$22}'
```

```
APOC3_Filter.vcf.hg19_multianno.txt
```

Question 6: Are these variants common or rare, and are some more frequent in a specific population?

1.c All variation

vii. dbSNP: The Single Nucleotide Polymorphism database (dbSNP) or Database of Short Genetic Variations is a public-domain archive for a broad collection of simple genetic polymorphisms.

2. Effect on gene function:

2.a Missense variants

Missense mutations are sometimes more difficult to evaluate compared to loss-of-function mutations. If a variant occurs at a nucleotide or amino acid that is conserved through evolution, it is usually assumed that the specific nucleotide or amino acid is important to function. Whereas conservation scores such as PhyloP use evolution information to measure deleteriousness, there are also tools which combine information on evolution, biochemistry, structure and from public available databases etc., e.g. CADD, Eigen, MutationTaster.

- We highlighted a select useful scoring methods that will help evaluate pathogenicity of a missense mutation:
- CADD*: Combined Annotation Dependent Depletion (CADD) is a framework that integrates multiple annotations into one metric by contrasting variants that survived natural selection with simulated mutations. A scaled C-score of ≥ 10 indicates that the variant is predicted to be within 10% of most deleterious substitutions within the human genome, a score of ≥ 20 indicates the variant is predicted to be within 1% of the most deleterious variants, and so on. In the annotation above, we added all CADD scores in the exome + all CADD score in the genome > 20 c-scores. This score includes single nucleotide variants as well as insertion/deletions.
 - Eigen and Eigen-PC*: Integrates different annotations into one measure of functional importance, a single functional score that can be incorporated in fine-mapping studies. Results for Eigen and Eigen-PC are similar for coding variants, but Eigen-PC has a considerable advantage over Eigen for noncoding variants. A positive Eigen-PC score is considered more damaging than a negative score.
 - SIFT: Sorting Tolerant from Intolerant predicts whether an amino acid substitution affects protein function. SIFT prediction is based on the degree of conservation of amino acid residues in sequence alignments derived from closely related sequences.

- It assumes that important positions in a protein sequence have been conserved throughout evolution and therefore substitutions at these positions may affect protein function. The SIFT score ranges from 0.0 (deleterious or “D”) to 1.0 (tolerated or “T”).
- PolyPhen2. Polymorphism Phenotyping v2. A tool which predicts possible impact of an amino acid substitution on the structure and function of a human protein using straightforward physical and comparative considerations. It obtains information from multiple sources such as variant site (e.g. active, binding, transmembrane, etc), multiple sequence alignment, secondary and 3D structure (if a known model exists), accessible surface area, etc.
 - PolyPhen2 HVAR: This metric is useful for diagnostics of Mendelian diseases, which requires distinguishing mutations with drastic effects from all the remaining human variation, including abundant mildly deleterious alleles. The variant is considered probably damaging (D; score 0.909 and 1), possibly damaging (P; 0.447 and 0.908), or benign (B; 0 and 0.446).
 - PolyPhen2 HDIV: PolyPhen HDIV should be used when evaluating rare variants involved in complex phenotypes and analysis of natural selection from sequence data. Variants can be classified as following: Probably damaging (D; 0.957 and 1), possibly damaging (P; 0.453 and 0.956), or benign (B; 0 and 0.452).
 - LRT: The Likelihood Ratio Test. Using a comparative genomics data set of protein-coding sequences from 32 vertebrate species, the LRT was used to compare the null model that each codon is evolving neutrally, with the alternative model that the codon has evolved under negative selection. LRT can accurately identify a subset of deleterious mutations that disrupt highly conserved amino acids within protein-coding sequences, which are likely to be unconditionally deleterious. LRT prediction ‘D’ stands for ‘deleterious’ and ‘N’ stands for ‘neutral’.
 - MutationTaster*: Mutation taster performs a battery of *in silico* tests to estimate the impact of the variant on the gene product / protein. Tests are made on both, protein and DNA level. MutationTaster is not limited to substitutions of single amino acids but can also handle synonymous or intronic variants. It has four types of prediction outcomes: “disease_causing_automatic”, “disease_causing”, “polymorphism”, and “polymorphism_automatic”, which are coded as “A”, “D”, “N”, and “P,” respectively. Among them, “D” and “N” are determined by the prediction algorithm, whereas “A” and “P” are determined by external information. “A” and “D” can be regarded as prediction for deleteriousness.
 - FATHMM and fathmm-MKL*: Functional Analysis Through Hidden Markov Models can be used for the prediction of the functional consequences of both coding variants and non-coding variants, using different algorithms. The more recent MKL algorithm can be used for all variants, utilizes various genomic annotations, and learns to weight the significance of each component annotation source. Variants are classified as either “damaging” (“D”) or “tolerated” (“T”).
 - GERP++*: Genomic Evolutionary Rate Profiling (GERP) is a method for producing position-specific estimates of evolutionary constraint using maximum likelihood evolutionary rate estimation. GERP++ uses a more rigorous set of algorithms. Positive scores represent a substitution deficit (i.e., fewer substitutions than the average neutral site) and thus indicate that a site may be under evolutionary constraint. Negative scores indicate that a site is probably evolving neutrally. It was suggested that a RS score threshold of 2 provides high sensitivity while still

strongly enriching for truly constrained sites; in practice, the threshold depends on the user.

- PhyloP*: (phylogenetic p-values) Evolutionary conservation at individual alignment sites, based on multiple alignments of 100 vertebrate species (100-way) or 20 mammals (20-way) under a null hypothesis of neutral evolution. Positive PhyloP scores indicate conserved sites (slower evolution than expected under neutral drift), the greater the score, the more conserved the site is; negative PhyloP scores indicate fast-evolving site (faster evolution than expected under neutral drift).

*available genome wide – that means they can be used to evaluate synonymous and non-coding variants as well (not all available genome-wide in ANNOVAR for annotation though). These scores are all integrated in dbSNFP, and more information and references can be found here: <https://sites.google.com/site/jpopgen/dbNSFP>

Let us evaluate some of these predictions above for our variants

`$ awk -F'\t' '{print $1,$2,$3,$6,$7}' APOC3_Filter.vcf.hg19_multianno.txt`

Note that these were loaded from a database here only including the exome. Individual datasets for some of these are available for annotation genome-wide as well.

Question 7: Can you fill in the other cells, which of the 3 missense variants have a prediction to be likely damaging?

| Chr | Position | Ref Allele | Alt Allele | Variant Type | Polyphen2_HDIV | PhyloP_100way | CADD_phred |
|-----|-----------|------------|------------|--------------|----------------|---------------|------------|
| 11 | 116701560 | G | A | missense | 1 | 4.302 | 23.6 |
| 11 | 116703532 | A | G | missense | | | |
| 11 | 116703580 | A | G | missense | | | |

2.b Splice variants:

- AdaBoost and Random Forest: Adaptive boosting (ADA) and random forest (RF) scores in dbSNV. dbSNV includes all potential human SNVs within splicing consensus regions (-3 to +8 at the 5' splice site and -12 to +2 at the 3' splice site). A score > 0.6 is considered damaging. Changing your splice boundaries to include splice region in combination with these scores can be useful to identify additional splice modifying variants.
- Regsnpintron: For all intronic SNPs including splice variants. See paragraph below. Please note that the current version is not working but should be updated soon.

Using the following methods examine the scores for the splice variants that we found earlier in the exercise:

`$ awk -F'\t' '{print $1,$2,$9,$10}' APOC3_Filter.vcf.hg19_multianno.txt`

common, and was proposed as a candidate variant involved plasma triglycerides levels and coronary heart disease (PMID: 27624799).

Question 9: Based on the bioinformatics tools predictions, what do you think about the impact of the six variants on the function of the apoC-III protein?

E. Other useful annotations

Mitochondrial annotations

ANNOVAR has a database to annotate the impact of mitochondrial mutations: mitimpact24, use in the filter option

Gene intolerance to mutations scores

These scores can help evaluate whether a gene is tolerable or intolerable to damaging mutations:

- ExAC constraint metrics (pLI and z-scores): can be found on the ExAC website (Gene search).
 - Synonymous and missense: A signed Z score for the deviation of observed counts from the expected number was created. Positive Z scores imply increased constraint (intolerance to variation; i.e. the gene had fewer variants than expected). Negative Z scores indicated that the gene had more variants than expected.
 - LoF: For this metric, three classes of genes with respect to tolerance to LoF variation are assumed: 1) null (where LoF variation is completely tolerated), 2) recessive (where heterozygous LoFs are tolerated), 3) haploinsufficient (i.e. heterozygous LoFs are not tolerated). The observed and expected variants counts were used to determine the probability that a given gene is extremely intolerant of loss-of-function variation (falls into the third category). The closer pLI is to 1, the more LoF intolerant. A pLI ≥ 0.9 is considered as an extremely LoF intolerant set of genes.
- LoFtool score: gene loss-of-function score percentiles. The smaller the percentile, the most intolerant is the gene to functional variation.
- RVIS-ESV score: RVIS score measures genetic intolerance of genes to functional mutations.
- GDI score: the gene damage index (GDI) depicts the accumulated mutational damage for each human gene in the general population. Highly mutated/damaged genes are unlikely to be disease-causing. Yet these genes generate a big proportion of false positive variants harbored in such genes. Removing high GDI genes is a very effective way to remove confidently false positives from WES/WGS data. Damage predictions (low/medium/high) are made for different disease types.

F. Useful online annotation tools

These webtools are also very useful in annotating variants:

WGS Annotator (WGSAnnot) - an annotation pipeline for human genome re-sequencing studies: <https://sites.google.com/site/jpopgen/wgsa/using-wgsa-via-aws>

Web Annovar: <http://wannovar.wglab.org/>

Seattleseq: <http://snp.gs.washington.edu/SeattleSeqAnnotation138/>

Ensembl variant predictor: <http://www.ensembl.org/info/docs/tools/vep/index.html>

Snp-nexus: <http://www.snp-nexus.org/>

Answers

Question 1: Of the six *APOC3* variants that were analyzed how many are exonic variants?

4 variants are exonic (3 nonsynonymous and one stop gain), one splice site variant and one 3'UTR variant.

| Chr | Position | Ref | Alt | avsnp150 | Func | Gene | ExonicFunc | cDNA/AAchange |
|-----|-----------|-----|-----|-------------|----------|-------|---------------|----------------------------|
| 11 | 116701353 | C | T | rs76353203 | Exonic | APOC3 | stopgain | NM_000040:c.C55T:p.R19X |
| 11 | 116701560 | G | A | rs147210663 | Exonic | APOC3 | nonsynonymous | NM_000040:c.G127A:p.A43T |
| 11 | 116701613 | G | T | rs140621530 | Splicing | APOC3 | . | NM_000040:exon3:c.179+1G>T |
| 11 | 116703532 | A | G | rs121918382 | Exonic | APOC3 | nonsynonymous | NM_000040:c.A232G:p.K78E |
| 11 | 116703580 | A | G | rs121918381 | Exonic | APOC3 | nonsynonymous | NM_000040:c.A280G:p.T94A |
| 11 | 116703671 | G | T | rs4225 | UTR3 | APOC3 | . | NM_000040:c.*71G>T |

Question 2: What has changed compared to the initial annotation (hint: splicing thresholds were changed)?

The first variant at position 11:116701353 changed to *exonic:splicing* by changing our threshold to 12bp distance from the splice site. This variant is located at the -1 position of a 5' donor splice site and could affect splicing as well.

Question 3: Which the *APOC3* variants are within conserved genomic region?

The second and third variant.

Question 4. Which of these variants is reported in the ANNOVAR GWAS catalog, and what has it been associated with?

The first variant, rs76353203, is indicated to have been associated with Triglyceride levels and high density lipoprotein cholesterol levels. This variant was the first variant in *APOC3* to have been associated with apoC-III deficiency, lower serum triglycerides, and higher levels of HDL cholesterol, and lower levels of LDL cholesterol (Pollin et al, 2008; PubMed: 19074352) and reached genome-wide significance in several GWAS for lower plasma triglyceride levels studies afterwards (PubMed: 24941081; PMID:24343240)

Question 5: Is one of the variants reported as 'pathogenic' in Clinvar? If yes, which variants and which phenotype has been associated with these variants?

The first 5 variants are in Clinvar and reported as pathogenic, associated with coronary heart disease, Hyperalphalipoproteinemia, and Apolipoprotein_c-iii.

Question 6: Are these variants common or rare, and are some more frequent in a specific population?

The first five variants (exonic and splice) are rare, the last variant in the 3'UTR is common (44% overall prevalence in the genomes). The second variant has a higher frequency in the ASJ population (1.1%; Ashkenazi Jewish) compared to all other populations.

Question 7: Can you fill in the other cells, which of the 3 missense variants have a prediction to be likely damaging?

| Chr | Position | Ref Allele | Alt Allele | Variant Type | Polyphen2_HDIV | PhyloP_100way | CADD_phred |
|-----|-----------|------------|------------|--------------|----------------|---------------|------------|
| 11 | 116701560 | G | A | missense | 1 | 4.302 | 23.6 |

| | | | | | | | |
|----|-----------|---|---|----------|-------|-------|-------|
| 11 | 116703532 | A | G | missense | 0.611 | 0.719 | 15.56 |
| 11 | 116703580 | A | G | missense | 0.123 | 0.194 | 0.175 |

The first missense variant is very likely to be damaging. The second as well, though the last one is not predicted to be damaging by these 3 scoring methods, and more methods should be evaluated.

Question 8: Can you fill in the ADA and RF scores below for the splice variants. Do these variants affect splicing?

| Chr | Start | Func.refGene | Effect | dbscSNV_ADA_SCORE | dbscSNV_RF_SCORE |
|-----|-----------|-----------------|---------------|-------------------|------------------|
| 11 | 116701353 | exonic;splicing | c.C55T;p.R19X | 0.0001 | 0.16 |
| 11 | 116701613 | splicing | c.179+1G>T | 1.000 | 0.936 |

The second variant is likely to affect splicing, as both scores are > 0.6. The first variant is located within exon (-1 position) of a 5' donor site, but is unlikely to affect splicing. This variant creates a stop mutation instead.

It is important to note that splice region variants (not standardly annotated unless you change boundaries) can still impact splicing, and annotation with these scores can help you evaluate their effect on splicing.

Question 9: Based on the bioinformatics tools predictions, what do you think about the impact of the six variants on the function of the apoC-III protein?

The first 3 variants, studied in a GWAS of 3734 participants and validated in 110,970 persons (PMID: 24941081), are predicted to be the most impactful on gene function based on all annotations.