

Tutorial 2

Economics and Genetics (FEB13089)

Erasmus School of Economics – Bachelor (Block 2)

Learning goals

- Becoming familiar with basic functionality of PLINK.
- Becoming familiar with the structure of genetic data.

Tutorial 2 does not provide direct input for the Individual assignment, but it makes you familiar with the structure of genetic data so that you will be able to successfully complete Tutorial 3 and Tutorial 4. The latter two tutorials provide direct input for Section 3 and Section 4 of your Individual assignment.

Data

On Canvas, you can find the files Example.bed, Example.bim, Example.fam, and Example_height.pheno you need for Tutorial 2. Please download these files from Canvas and store them in an easily accessible folder on your PC. In this Tutorial, I'll assume you have stored these files in the folder C:\Users\Niels\Documents\EUR\E&G\Tutorials. Whenever this path to the folder with data is mentioned in this Tutorial, please use your own path to it.

Introducing PLINK

PLINK is a free, open-source whole genome association analysis toolset, designed to perform a range of basic, large-scale analyses in a computationally efficient manner. PLINK can be downloaded from <https://www.cog-genomics.org/plink/1.9/>. Download the latest “Stable” version of PLINK suitable for your operating system (Windows / MacOS / Linux, this tutorial uses version “beta 6.18, 16 Jun”), and store the files in the downloaded folder in your data folder C:\Users\Niels\Documents\EUR\E&G\Tutorials.

PLINK is a command line program that does not need to be installed. Instead, you run it from the command line (CMD). In Windows, you can run PLINK by launching the command line (click “Start”, type “cmd” and press enter), navigating to the folder C:\Users\Niels\Documents\EUR\E&G\Tutorials in which plink.exe is placed, typing plink and pressing enter.¹ If you work on a MacBook, you can use “Terminal” instead of CMD (<http://www.wikihow.com/Get-to-the-Command-Line-on-a-Mac>). Again, before you can run PLINK, you need to navigate to the folder in which the plink executable is placed. When you are in this folder, type “plink” and press enter to check whether PLINK is running. ***Important note: On some MacBooks, you need to type “./plink” instead of “plink”; You need to do this every time you run PLINK in this Tutorial.***²

¹ You are advised to run this tutorial on your own computer, as it is somewhat complicated to run PLINK on a university PC. However, it is possible, see the document “How to run PLINK and GCTA on a university PC.pdf” on Canvas.

² Some MacOS users get a security warning when trying to run PLINK. I'm using PLINK for more than ten years now and the IT security team of Erasmus University has also checked it, and it's safe to override security settings for this program. Have a look for example at <http://safari.helpmax.net/en/privacy-and-security/turning-off-security-warnings/> for how to change your MacOS security setting.

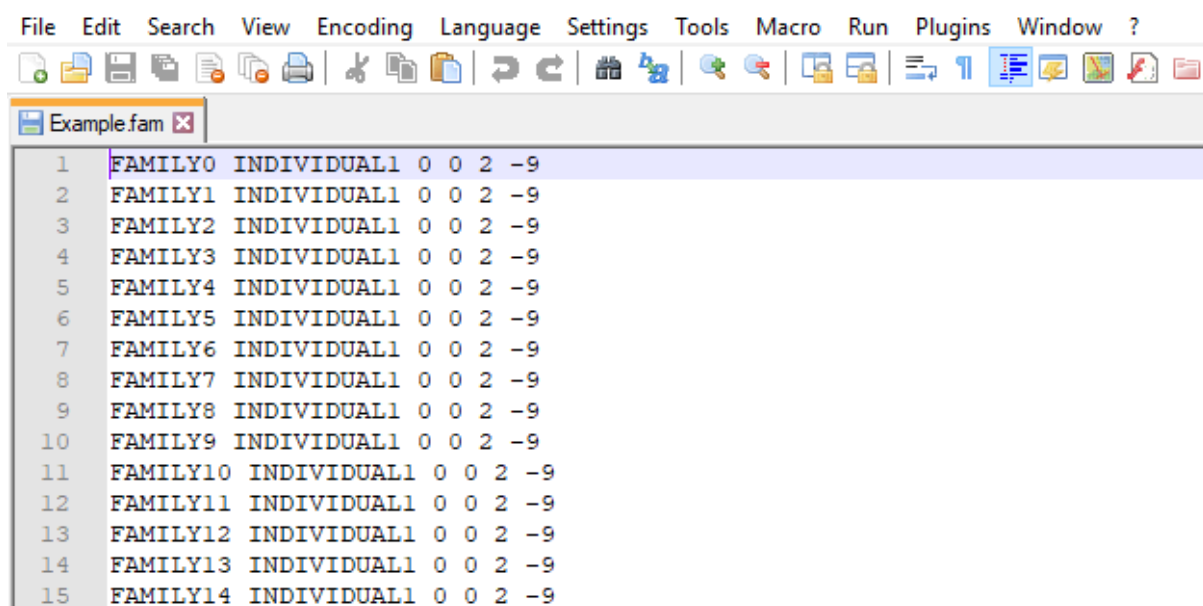
If you are not familiar with working from a command line, have a look at <https://www.computerhope.com/issues/chusedos.htm>. The most important navigation commands are: “cd XXXXX” to move into a directory, and “cd ..” to move one directory up. Alternatively, watch the first 6 minutes of this YouTube video <https://www.youtube.com/watch?v=aKRYQsKR46I> (MacOS). Windows users may want to watch <https://www.youtube.com/watch?v=MBBWVgEOewk> and the first 1:23 minutes of https://www.youtube.com/watch?v=7ABkcHLdG_A.

Data description

The emphasis in this tutorial is on getting familiar with genetic (SNP) data, and some basic functionality of PLINK. For this purpose, we use the dataset “Example”. The data for this tutorial are available on Canvas and it is assumed you have stored the data (and plink.exe) in the folder C:\Users\Niels\Documents\EUR\E&G\Tutorials. The three files Example.bed, Example.bim, and Example.fam together contain genetic information (23,825 SNPs from chromosomes 1-22) from 2,000 individuals. Only the .bim and the .fam file are plain text files that can be opened in text editors.

Let’s open the Example.fam file first. To do so, I recommend to use Notepad++ (Windows, available from <https://notepad-plus-plus.org/>) or Atom (MacOS, available from <https://mac.filehorse.com/download-atom/>) because these free text editors can handle large data files easily. Note that we are using these editors here to get a better sense of the contents of the genetic data files; We’ll edit the genetic data files using PLINK.

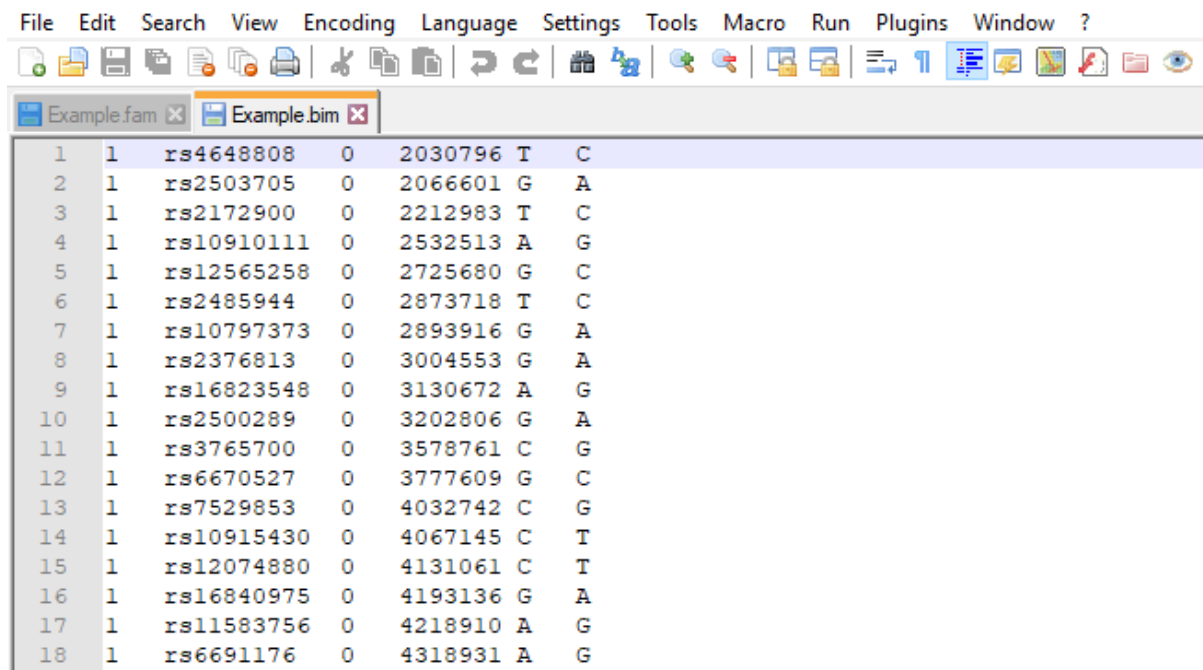
When you open the Example.fam, you should see the following:



Line	FID	IID	PAT	MAT	Sex	Phenotype
1	FAMILY0	INDIVIDUAL1	0	0	2	-9
2	FAMILY1	INDIVIDUAL1	0	0	2	-9
3	FAMILY2	INDIVIDUAL1	0	0	2	-9
4	FAMILY3	INDIVIDUAL1	0	0	2	-9
5	FAMILY4	INDIVIDUAL1	0	0	2	-9
6	FAMILY5	INDIVIDUAL1	0	0	2	-9
7	FAMILY6	INDIVIDUAL1	0	0	2	-9
8	FAMILY7	INDIVIDUAL1	0	0	2	-9
9	FAMILY8	INDIVIDUAL1	0	0	2	-9
10	FAMILY9	INDIVIDUAL1	0	0	2	-9
11	FAMILY10	INDIVIDUAL1	0	0	2	-9
12	FAMILY11	INDIVIDUAL1	0	0	2	-9
13	FAMILY12	INDIVIDUAL1	0	0	2	-9
14	FAMILY13	INDIVIDUAL1	0	0	2	-9
15	FAMILY14	INDIVIDUAL1	0	0	2	-9

The above figure gives an impression of the contents of Example.fam. The first column gives the family ID (FID) of the individual, the second column gives the (within-family) individual ID (IID) of the individual. The third column gives the paternal ID of the individual (PAT, 0=unknown), the fourth column the maternal ID (MAT, 0=unknown), the fifth column Sex (1=male; 2=female; other=unknown), the sixth column the phenotype (dependent variable). You can see that in this dataset all individuals are unrelated: They come all from different families, and their parents are all unknown. All individuals in the sample are female, and the phenotype equals -9 for every individual in the dataset (in PLINK, -9 means “missing”).

Let's open Example.bim now (see the figure below). The first column of the BIM file contains the chromosome number, the second column the SNP identifier ("rs-number"), the third column the genetic distance (a measure for the position of SNP in the genome, it is not frequently used so therefore it is set to 0 in this dataset), and the fourth column the basepair position (the position of the SNP on a chromosome). A SNP has only two possible alleles in the population [A/C/G/T], one is given in column five and the other one is given in column six.



Line	Chromosome	rs-id	Distance	Position	Alleles
1	1	rs4648808	0	2030796	T C
2	1	rs2503705	0	2066601	G A
3	1	rs2172900	0	2212983	T C
4	1	rs10910111	0	2532513	A G
5	1	rs12565258	0	2725680	G C
6	1	rs2485944	0	2873718	T C
7	1	rs10797373	0	2893916	G A
8	1	rs2376813	0	3004553	G A
9	1	rs16823548	0	3130672	A G
10	1	rs2500289	0	3202806	G A
11	1	rs3765700	0	3578761	C G
12	1	rs6670527	0	3777609	G C
13	1	rs7529853	0	4032742	C G
14	1	rs10915430	0	4067145	C T
15	1	rs12074880	0	4131061	C T
16	1	rs16840975	0	4193136	G A
17	1	rs11583756	0	4218910	A G
18	1	rs6691176	0	4318931	A G

Unfortunately, we cannot open the Example.bed file in a text editor, because it is in binary format. We will use PLINK to reformat this file in such a way that it can be opened.

Running PLINK with example data

Let's launch plink and make sure that it reads in the binary genetic data file. You can do this with the `--bfile` flag (**note the two dashes**). After the `--bfile` flag you give the name of your datafile. The `--bfile` option takes a single parameter, the root of the input file names, and will search for three files: a .bed file, a .bim file, and a .fam file with this root name. In other words, using "`--bfile Example`" implies that Example.bed, Example.bim, and Example.fam should exist in your working directory.

Let's type: `plink --bfile Example --out test`

(with "`--out`", we specify the root of the output file names; in this case a logfile only)

Note: On some MacBooks, you need to type `./plink` instead of `plink`.³

If PLINK runs properly, then you should see the following:

³ If PLINK does not run on your Mac, please check whether you navigated your Terminal already to the right directory. You can do this using the "pwd" command in Terminal. If it doesn't give "C:\Users\Niels\Documents\EUR\E&G\Tutorials" then first move to this directory and try again.

```
Opdrachtprompt
C:\Users\Niels\Documents\EUR\E&G\Tutorials>plink --bfile Example
PLINK v1.90b6.18 64-bit (16 Jun 2020)      www.cog-genomics.org/plink/1.9/
(C) 2005-2020 Shaun Purcell, Christopher Chang  GNU General Public License v3
Logging to plink.log.
Options in effect:
  --bfile Example

Warning: No output requested. Exiting.

plink <input flag(s)...> [command flag(s)...] [other flag(s)...]
plink --help [flag name(s)...]

Commands include --make-bed, --recode, --flip-scan, --merge-list,
--write-snp-list, --list-duplicate-vars, --freqx, --missing, --test-mishap,
--hardy, --mendel, --ibc, --impute-sex, --indep-pairphase, --r2, --show-tags,
--blocks, --distance, --genome, --homozyg, --make-rel, --make-grm-gz,
--rel-cutoff, --cluster, --pca, --neighbour, --ibs-test, --regress-distance,
--model, --bd, --gxe, --logistic, --dosage, --lasso, --test-missing,
--make-perm-pheno, --tdt, --qfam, --annotate, --clump, --gene-report,
--meta-analysis, --epistasis, --fast-epistasis, and --score.

"plink --help | more" describes all functions (warning: long).
C:\Users\Niels\Documents\EUR\E&G\Tutorials>
```

Now we know that PLINK is running, so let's *request some output*. We are going to do two things. 1) Reformat the dataset in such a way that we can open it with a text editor (with "--recode"). 2) Using --out, we give the dataset the new name "Example_non_binary".

```
plink --bfile Example --recode --out Example_non_binary
```

```
C:\Users\Niels\Documents\EUR\E&G\Tutorials>plink --bfile Example --recode --out
Example_non_binary
PLINK v1.90b6.18 64-bit (16 Jun 2020)      www.cog-genomics.org/plink/1.9/
(C) 2005-2020 Shaun Purcell, Christopher Chang  GNU General Public License v3
Logging to Example_non_binary.log.
Options in effect:
  --bfile Example
  --out Example_non_binary
  --recode

8114 MB RAM detected; reserving 4057 MB for main workspace.
23825 variants loaded from .bim file.
2000 people (0 males, 2000 females) loaded from .fam.
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 2000 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Total genotyping rate is 0.970001.
23825 variants and 2000 people pass filters and QC.
Note: No phenotypes present.
--recode ped to Example_non_binary.ped + Example_non_binary.map ... done.
C:\Users\Niels\Documents\EUR\E&G\Tutorials>
```

From the screen, you can see that there are 2,000 individuals in your dataset, with information about 23,825 genetic variants (SNPs). With the --recode option, two new files were created in your directory: Example_non_binary.ped and Example_non_binary.map. Let's open the MAP file first:

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?				
Example.fam x Example.bim x Example_non_binary.ped x Example_non_binary.map x				
1	1	rs4648808	0	2030796
2	1	rs2503705	0	2066601
3	1	rs2172900	0	2212983
4	1	rs10910111	0	2532513
5	1	rs12565258	0	2725680
6	1	rs2485944	0	2873718
7	1	rs10797373	0	2893916
8	1	rs2376813	0	3004553
9	1	rs16823548	0	3130672
10	1	rs2500289	0	3202806
11	1	rs3765700	0	3578761
12	1	rs6670527	0	3777609
13	1	rs7529853	0	4032742

You can see that the MAP file is identical to the BIM file, except for the fact that it misses the fifth and sixth column from the BIM file. This information (and the information of the FAM file) is now included in the PED file. Let's open the PED file now (your computer may protest a bit, because of the large file size – where the binary BED file is ~11MB the PED file is ~186MB):

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Example.fam x Example.bim x Example_non_binary.ped x
1 FAMILY0 INDIVIDUAL1 0 0 2 -9 T T G A T C A G 0 0 0 0 G A G A G G A A G G G C
2 FAMILY1 INDIVIDUAL1 0 0 2 -9 T T A A C C A G G C T C G A 0 0 A G G G C G G C
3 FAMILY2 INDIVIDUAL1 0 0 2 -9 T T 0 0 T T A G G C T T A A A A A A A G G G C
4 FAMILY3 INDIVIDUAL1 0 0 2 -9 T T A A 0 0 A A C C T T G A G G G G A C G G G
5 FAMILY4 INDIVIDUAL1 0 0 2 -9 T T G G T C G G G C T T A A G G G G A A C C G C
6 FAMILY5 INDIVIDUAL1 0 0 2 -9 C C G G C C A A G C T T G G G A A G G A G G G C
7 FAMILY6 INDIVIDUAL1 0 0 2 -9 T C G A 0 0 A A G C T C G A G A G G A A G G G C
8 FAMILY7 INDIVIDUAL1 0 0 2 -9 T T G G T T A G G G C C A A G A A G G G C C C C
9 FAMILY8 INDIVIDUAL1 0 0 2 -9 0 0 G G T C G G G G C C 0 0 G A A G G A C C G C
10 FAMILY9 INDIVIDUAL1 0 0 2 -9 T C A A T C A G G C C C A A G G A A A C G G G
11 FAMILY10 INDIVIDUAL1 0 0 2 -9 T C G A T T A G C C T C G A A A G G G A C G G C
12 FAMILY11 INDIVIDUAL1 0 0 2 -9 T T A A C C G G C C C C 0 0 G A A G A G G G C
```

The first 6 columns are the same as the columns in the FAM file Each set of two columns represent a SNP

Each row in the PED file represents a genotyped individual. The first 6 columns give general information about the individual (the contents of the FAM file: FID, IID, PAT, MAT, SEX, PHENO), and the next columns give specific information about the SNPs in the data. Specifically, from the seventh column onwards, each pair of two columns represent a SNP, in the order of the MAP file. Thus, column 7 and 8 represent the first SNP in the MAP file (rs4648808, check this in Example_non_binary.map), column 9 and 10 the second (rs2503705, check this), etc.

The binary (BED/BIM/FAM) files are the default option to work with in PLINK, but in some cases the non-binary files (PED/MAP) are useful to work with because they can be more easily read in by other software tools. In case you have non-binary files, you can convert them to binary files with PLINK (this may be useful for the simulation exercise at the end of this tutorial). Let's practice this conversion with our dataset Example_non_binary (and let's call the new dataset Example_binary). Of course, the "Example" dataset is exactly the same as the dataset "Example_binary" you are now going to create. Note that in this command we are using --file rather than --bfile, because we ask PLINK to work with non-binary files.

```
plink --file Example_non_binary --make-bed --out Example_binary
```

Three files are created with this command: The binary file that contains the raw genotype data Example_binary.bed but also a revised map file Example_binary.bim which contains two extra columns that give the allele names for each SNP, and Example_binary.fam which contains just the first six columns of Example_non_binary.ped. Again, you can open the .bim and .fam files in a text editor -- but do not try to open the .bed file (if you nevertheless try, you will see that the contents of the file are not readable ;-)).

At this point in the Tutorial, you may want to clean up your working directory a bit by deleting the files Example_binary.bed, Example_binary.bim, Example_binary.fam, Example_binary.log, Example_non_binary.log, Example_non_binary.map, and Example_non_binary.ped. We will not use these files anymore in the Tutorial.

Filtering data

We continue this tutorial with the binary dataset “Example”, and focus now on filtering SNPs and individuals. Genotyping individuals is an error-prone process, and there are many options to filter your data, see <https://www.cog-genomics.org/plink/1.9/filter> for an overview. One could loosely divide the filters in general filters based on properties of the data, and specific filters based on data contents (that is, what you know about the individuals and SNPs in the data). *We practice with these filters, as that may help you to complete the simulation exercise at the end of this tutorial.*

General filters

The most commonly used general filters are: --maf (minor allele frequency filter), --geno (missingness per SNP filter), and --mind (missingness per individual filter). If you do not impose filters, than these missing rates and allele frequency filters are automatically set to exclude nobody/no SNPs. Try to filter the example data on a maximum SNP missingness of 0.03 (3%), maximum individual missingness of SNPs of 0.05 (5%) and a MAF of at least 0.05 (5%) with the following command, and store the filtered file in a file with the name “Example_filtered” (with --make-bed and --out).

```
plink --bfile Example --make-bed --out Example_filtered --geno 0.03
--mind 0.05 --maf 0.05
```

```
C:\Users\Niels\Documents\EUR\E&G\Tutorials>plink --bfile Example --make-bed --out Example_filtered --geno 0.03 --mind 0.05 --maf 0.05
PLINK v1.90b6.18 64-bit (16 Jun 2020)          www.cog-genomics.org/plink/1.9/
(C) 2005-2020 Shaun Purcell, Christopher Chang GNU General Public License v3
Logging to Example_filtered.log.
Options in effect:
  --bfile Example
  --geno 0.03
  --maf 0.05
  --make-bed
  --mind 0.05
  --out Example_filtered

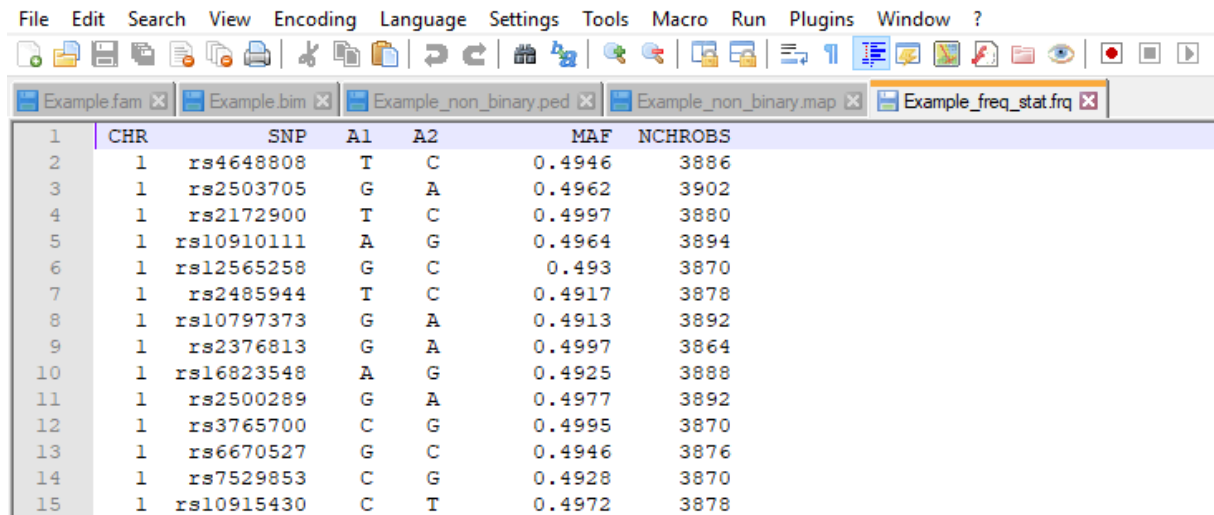
8114 MB RAM detected; reserving 4057 MB for main workspace.
23825 variants loaded from .bim file.
2000 people (0 males, 2000 females) loaded from .fam.
0 people removed due to missing genotype data (--mind).
Using 1 thread (no multithreaded calculations invoked).
Before main variant filters, 2000 founders and 0 nonfounders present.
Calculating allele frequencies... done.
Total genotyping rate is 0.970001.
11054 variants removed due to missing genotype data (--geno).
0 variants removed due to minor allele threshold(s)
(--maf/--max-maf/--mac/--max-mac).
12771 variants and 2000 people pass filters and QC.
Note: No phenotypes present.
--make-bed to Example_filtered.bed + Example_filtered.bim +
Example_filtered.fam ... done.
```

You can see that 11,054 SNPs are removed based on SNP missingness (--geno) and 0 for low allele frequency (--maf). No individual is removed because of high SNP missingness (--mind), hence the new dataset set includes 2,000 individuals and 12,771 SNPs.

Summary statistics: Allele frequencies

Let's have a more in depth look at the allele frequencies in the data. The following command generates a file called `Example_freq_stat.frq` which contains the minor allele frequency and allele codes for each SNP.

```
plink --bfile Example --freq --out Example_freq_stat
```



1	CHR	SNP	A1	A2	MAF	NCHROBS
2	1	rs4648808	T	C	0.4946	3886
3	1	rs2503705	G	A	0.4962	3902
4	1	rs2172900	T	C	0.4997	3880
5	1	rs10910111	A	G	0.4964	3894
6	1	rs12565258	G	C	0.493	3870
7	1	rs2485944	T	C	0.4917	3878
8	1	rs10797373	G	A	0.4913	3892
9	1	rs2376813	G	A	0.4997	3864
10	1	rs16823548	A	G	0.4925	3888
11	1	rs2500289	G	A	0.4977	3892
12	1	rs3765700	C	G	0.4995	3870
13	1	rs6670527	G	C	0.4946	3876
14	1	rs7529853	C	G	0.4928	3870
15	1	rs10915430	C	T	0.4972	3878

The resulting file `Example_freq_stat.frq` contains 6 columns. The first column includes the chromosome number, the second the SNP identifier, the third one the reference allele, the fourth one the other allele, the fifth column the minor allele frequency, and the sixth column the number of observations (number of alleles). In this dataset, for most SNPs, the allele frequency is close to 0.50 (50%).

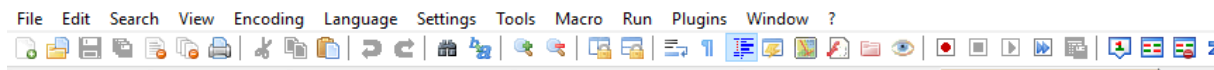
You might have expected that the NCHROBS should have been 4,000 (2,000 individuals with each 2 alleles). The reason that it is a bit lower (e.g., 3,886 for rs4648808) is that information for some SNPs is missing for some individuals. Therefore, SNP missingness is the next thing we are going to inspect.

Summary statistics: Missingness rates

Next, we will generate some simple summary statistics on rates of missing data in the file, using the `--missing` option:

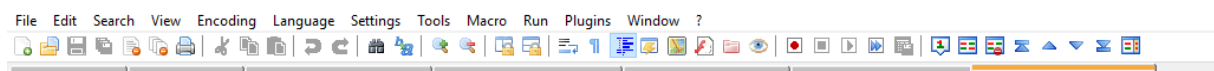
```
plink --bfile Example --missing --out Example_miss_stat
```

This command generates the files `Example_miss_stat.imiss` (individual missingness information), and `Example_miss_stat.lmiss` (locus (SNP) missingness information). The `.lmiss` file looks as follows:



	CHR	SNP	N_MISS	N_GENO	F_MISS
1					
2	1	rs4648808	57	2000	0.0285
3	1	rs2503705	49	2000	0.0245
4	1	rs2172900	60	2000	0.03
5	1	rs10910111	53	2000	0.0265
6	1	rs12565258	65	2000	0.0325
7	1	rs2485944	61	2000	0.0305
8	1	rs10797373	54	2000	0.027
9	1	rs2376813	68	2000	0.034
10	1	rs16823548	56	2000	0.028
11	1	rs2500289	54	2000	0.027
12	1	rs3765700	65	2000	0.0325
13	1	rs6670527	62	2000	0.031
14	1	rs7529853	65	2000	0.0325

For each SNP, we see the number of individuals with missing information for this SNP (N_MISS), the total number of genotyped individuals (N_GENO) and the proportion of individuals missing (F_MISS). If we take a look at the .imiss file, we see something very similar:



	FID	IID	MISS_PHENO	N_MISS	N_GENO	F_MISS
1						
2	FAMILY0	INDIVIDUAL1	Y	702	23825	0.02946
3	FAMILY1	INDIVIDUAL1	Y	694	23825	0.02913
4	FAMILY2	INDIVIDUAL1	Y	733	23825	0.03077
5	FAMILY3	INDIVIDUAL1	Y	775	23825	0.03253
6	FAMILY4	INDIVIDUAL1	Y	689	23825	0.02892
7	FAMILY5	INDIVIDUAL1	Y	691	23825	0.029
8	FAMILY6	INDIVIDUAL1	Y	701	23825	0.02942
9	FAMILY7	INDIVIDUAL1	Y	704	23825	0.02955
10	FAMILY8	INDIVIDUAL1	Y	727	23825	0.03051
11	FAMILY9	INDIVIDUAL1	Y	710	23825	0.0298
12	FAMILY10	INDIVIDUAL1	Y	672	23825	0.02821
13	FAMILY11	INDIVIDUAL1	Y	709	23825	0.02976
14	FAMILY12	INDIVIDUAL1	Y	641	23825	0.0269
15	FAMILY13	INDIVIDUAL1	Y	731	23825	0.03068
16	FAMILY14	INDIVIDUAL1	Y	655	23825	0.02749
17	FAMILY15	INDIVIDUAL1	Y	654	23825	0.02745

For each individual (the combination of FID and IID), the final column (F_MISS) gives the actual (1-)genotyping rate for that individual. We see that the genotyping rate is ~0.97 (97%) for the first person (Individual 1 from Family 0) in this dataset.

Specific filters

PLINK also has functionality to filter out very specific parts of the data. For example, with the --chr option you can select a specific chromosome (or several chromosomes), and with --snp you can select a specific SNP. Here, we practice with the most often used filters:

SNP filters:

- extract: normally accepts a text file with a list of SNPs (usually one per line, but it's okay for them to just be separated by spaces), and removes all unlisted SNPs.

- exclude: does the same for all listed SNPs.

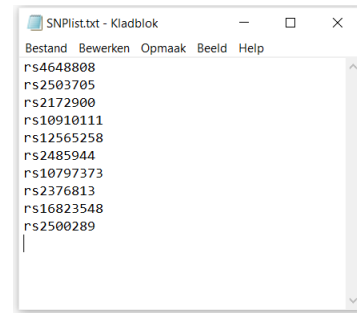
Individual filters:

- keep: accepts a space/tab-delimited text file with family IDs in the first column and individual IDs in the second column, and removes all unlisted samples from the current analysis.

- remove: does the same for all listed samples

Let's create a text file (with notepad or Notepad++) with the following ten lines:

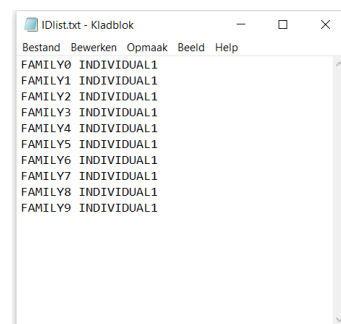
rs4648808
rs2503705
rs2172900
rs10910111
rs12565258
rs2485944
rs10797373
rs2376813
rs16823548
rs2500289



Save this file as SNPlist.txt in your working directory C:\Users\Niels\Documents\EUR\E&G\Tutorials.

Let's also create another text file (with notepad, Notepad++, or Atom) with the following ten lines:

FAMILY0 INDIVIDUAL1
FAMILY1 INDIVIDUAL1
FAMILY2 INDIVIDUAL1
FAMILY3 INDIVIDUAL1
FAMILY4 INDIVIDUAL1
FAMILY5 INDIVIDUAL1
FAMILY6 INDIVIDUAL1
FAMILY7 INDIVIDUAL1
FAMILY8 INDIVIDUAL1
FAMILY9 INDIVIDUAL1



Save this file as IDlist.txt in your working directory C:\Users\Niels\Documents\EUR\E&G\Tutorials.

With these two new files we construct two new datafiles. The first dataset only includes the above indicated 10 SNPs and 10 individuals. The second datasets includes all individuals and SNPs except the indicated 10 SNPs and 10 individuals.

```
plink --bfile Example --extract SNPlist.txt --keep IDlist.txt --out
Example_subset1 --make-bed
```

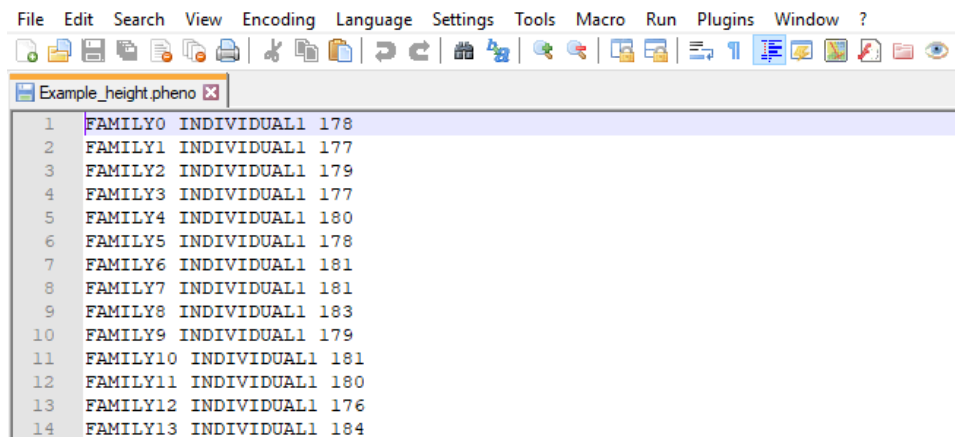
```
plink --bfile Example --exclude SNPlist.txt --remove IDlist.txt --
out Example_subset2 --make-bed
```

Have a look at the resulting FAM and BIM files (number of rows in it), to verify that your commands worked properly. *To clean up your working directory, you may now want to delete the genetic data files you created as well as the files containing the allele frequencies and missingness rates.*

Phenotype file

Let's have a look at the phenotype now. The default place where you can find the phenotype is the sixth column of the PED file (non-binary genetic data), and the sixth column of the FAM file (binary genetic data). In our data, the phenotype is missing for every individual ("-9"). However, phenotypes can also be stored in separate text files. It is common to use .pheno as extension for these files, e.g. Height.pheno. For this tutorial, you've downloaded "Example_height.pheno" from Canvas and stored it in your working directory. This file includes for every individual in your dataset body height in centimeters.

Open “Example_height.pheno” in a text editor; You’ll see that the structure of a phenotype file is:
FID IID PHENOTYPE:



```

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Example_height.pheno
1 FAMILY0 INDIVIDUAL1 178
2 FAMILY1 INDIVIDUAL1 177
3 FAMILY2 INDIVIDUAL1 179
4 FAMILY3 INDIVIDUAL1 177
5 FAMILY4 INDIVIDUAL1 180
6 FAMILY5 INDIVIDUAL1 178
7 FAMILY6 INDIVIDUAL1 181
8 FAMILY7 INDIVIDUAL1 181
9 FAMILY8 INDIVIDUAL1 183
10 FAMILY9 INDIVIDUAL1 179
11 FAMILY10 INDIVIDUAL1 181
12 FAMILY11 INDIVIDUAL1 180
13 FAMILY12 INDIVIDUAL1 176
14 FAMILY13 INDIVIDUAL1 184

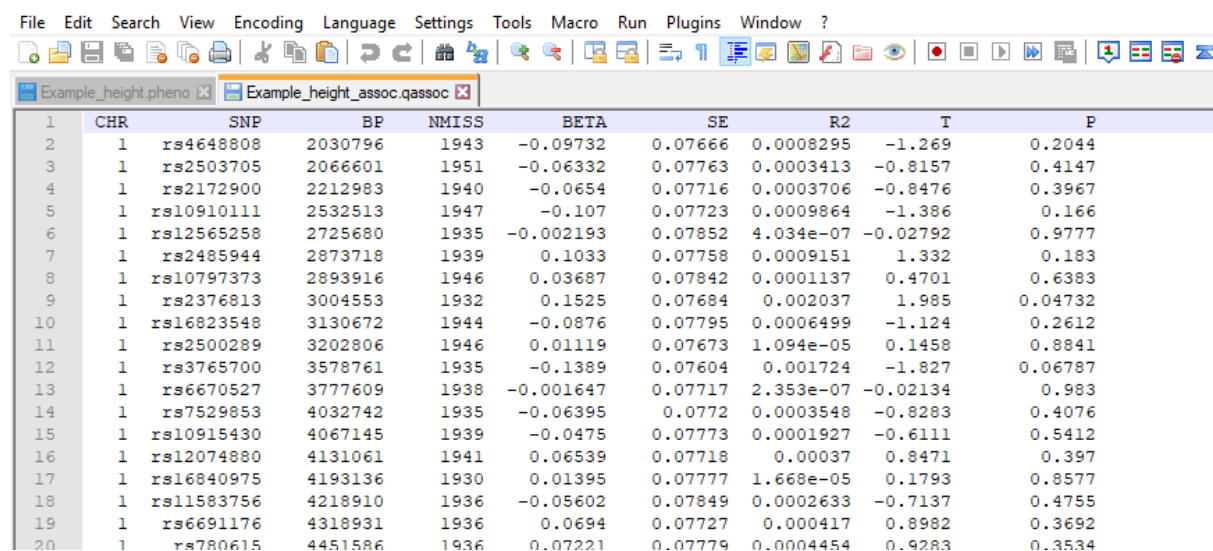
```

Basic association analysis

Let's now perform a basic association (correlation) analysis between all SNPs and the phenotype (in Tutorial 4, we will practice extensively with more advanced (Genome-Wide Association) models). Importantly, the association between each SNP and the phenotype is analyzed, hence in total 23,825 correlations will be computed with the “--assoc” command. With the “--pheno” flag we make sure PLINK uses our phenotype file in the analysis:

```
plink --bfile Example --assoc --pheno Example_height.pheno --out Example_height_assoc
```

This analysis generates a file Example_height_assoc.qassoc (see figure below).



	CHR	SNP	BP	NMISS	BETA	SE	R2	T	P
1	1	rs4648808	2030796	1943	-0.09732	0.07666	0.0008295	-1.269	0.2044
2	1	rs2503705	2066601	1951	-0.06332	0.07763	0.0003413	-0.8157	0.4147
3	1	rs2172900	2212983	1940	-0.0654	0.07716	0.0003706	-0.8476	0.3967
4	1	rs10910111	2532513	1947	-0.107	0.07723	0.0009864	-1.386	0.166
5	1	rs12565258	2725680	1935	-0.002193	0.07852	4.034e-07	-0.02792	0.9777
6	1	rs2485944	2873718	1939	0.1033	0.07758	0.0009151	1.332	0.183
7	1	rs10797373	2893916	1946	0.03687	0.07842	0.0001137	0.4701	0.6383
8	1	rs2376813	3004553	1932	0.1525	0.07684	0.002037	1.985	0.04732
9	1	rs16823548	3130672	1944	-0.0876	0.07795	0.0006499	-1.124	0.2612
10	1	rs2500289	3202806	1946	0.01119	0.07673	1.094e-05	0.1458	0.8841
11	1	rs3765700	3578761	1935	-0.1389	0.07604	0.001724	-1.827	0.06787
12	1	rs6670527	3777609	1938	-0.001647	0.07717	2.353e-07	-0.02134	0.983
13	1	rs7529853	4032742	1935	-0.06395	0.0772	0.0003548	-0.8283	0.4076
14	1	rs10915430	4067145	1939	-0.0475	0.07773	0.0001927	-0.6111	0.5412
15	1	rs12074880	4131061	1941	0.06539	0.07718	0.00037	0.8471	0.397
16	1	rs16840975	4193136	1930	0.01395	0.07777	1.668e-05	0.1793	0.8577
17	1	rs11583756	4218910	1936	-0.05602	0.07849	0.0002633	-0.7137	0.4755
18	1	rs6691176	4318931	1936	0.0694	0.07727	0.000417	0.8982	0.3692
19	1	rs780615	4451586	1936	0.07221	0.07779	0.0004454	0.9283	0.3534
20	1								

The columns in this file are: Chromosome, SNP identifier, Basepair position, Number of non-missing individuals for the correlation analysis, Regression coefficient, Standard error of the coefficient, The regression r-squared (multiple correlation coefficient), t-statistic for regression of phenotype on the SNP, Asymptotic significance value for coefficient. Check that in your output rs1953405 has the lowest *p*-value of all SNPs (2.034e-05 = 0.00002034). If we adopt a significance level of 5% in the analysis, then there are 148 significant SNPs in total. However, if we perform so many statistical

tests then we may question whether it is fair to keep the significance level at 5. *You'll hear more about this in Lecture 4.*

Showing you master PLINK

After going through all the steps in this Tutorial, you should have a basic understanding of PLINK functionality and the structure of genetic (SNP) data. If you want to make yourself more familiar with PLINK, try to complete the following simulation exercise. The goal is to construct a PED and MAP yourself, and to merge your simulated data with the Example data. If you get stuck, note that on Canvas a step-by-step demonstration is available (*Showing you master PLINK - Step by step demonstration.pptx*).

Simulate genetic data for 200 individuals and 3 SNPs (for example in Excel). As SNP identifiers, you should use the first 3 SNPs in Example.bim (rs4648808, rs2503705, and rs2172900). Generate (fake) Family IDs and Individual IDs. You can fill the paternal and maternal ID columns with zeroes (unrelated individuals). Make sure that the minor allele frequency in your data is uniformly distributed on the interval [0.10-0.50]. Make sure that PLINK can handle your simulated data (for example, copy the simulated data in Excel to a plain text editor such as Notepad and store it with the correct extensions [ped/map]). Check with PLINK that your data generation process was correct, for example by inspecting the range of allele frequencies using `--freq`. Try to merge your simulated data with the Example data using the `--bmerge` option (<https://www.cog-genomics.org/plink/1.9/data#merge>).