Practical 1 - The resemblance between relatives and estimation of (co)variance components

Introduction to software tools for these practicals

To complete the following practicals, we are going to be using a combination of the R programming language (R Core Team, 2015) and some command line tools. This will require the use of the command line. For Linux users this should be very familiar and if you are a Mac user it may be familiar too. For Windows users this means MS-DOS. If you haven't used the terminal much, then I suggest you execute one of the following listings to become familiar with basic navigation using the terminal. These sections are also critical to setting up the path structure to your practical resources. If you have a Windows machine you will need to jump to listing 2. If you are very familiar with the command line please skip the next two sections. IMPORTANT - ALL COMMANDS EXECUTED IN THE PRACTICAL ASSUME YOU ARE IN `THE SISG_AQG_2015` FOLDER ON YOUR DESKTOP.

Linux and Mac

It will be easiest if you have put the folder containing all the tools, data, and scripts on your `Desktop`. In all listings in this document the command line call will be expressed with a $ sign. This indicates that we are executing the command (not including the $) at the command line. Commands in R will be treated differently (see below). Commented lines will begin with a # and indicate comments about the commands being executed and are not to be executed. Feel free to copy line by line, however, over line copying will induce errors. If you get an error always remember to try and type the command as you may have copied some formatting. IMPORTANT - In the following listing the ~ command will not copy from the listing without error so please type it.

```
$ # Take a look at your present working directory
$ pwd
  /Users/home
$ # Your home directory is indicated with a ~. So ~/Desktop will be your desktop
$ # Change you directory so that your are in the SISG_AQG_2015 folder
$ # !!!Careful copying the next line as ~ will not copy correctly!!!!
$ cd ~/Desktop/SISG_AQG_2015
$ # When navigating through directories use the TAB key to complete a half typed path. For example
$ # once you are in ~/Desktop/SISG_AQG_2015 type
$ cd pra
$ # and then hit TAB to try and complete the line. Navigate to practical_1 using this trick
$ # Navigate back to the parent SISG_AQG_2015 by
$ cd ../
$ # Once you are back in the SISG_AQG_2015 type
$ ls
$ # This lists the contents of the folder. You can list it in row format by
$ ls -l
$ # These basic commands along with
$ mv /path/to/folder /path/to/place/of/interest # Move a folder from one directory to another
$ cp /path/to/folder/file.txt /path/to/place/of/interest # Move a file
$ cp -r /path/to/folder/ /path/to/place/of/interest # Copy a folder from one directory to another
$ # and some others will be key to these practicals
$ # IMPORTANT - We are going to add our binary folder to the PATH so that our machine know where
$ # our binaries live. If you have put the folder on the Desktop then the following should work
$ export PATH=$PATH:~/Desktop/SISG_AQG_2015/bin
$ echo $PATH
   /usr/local/Cellar/gcc/4.8.3_1/bin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin:/usr/local/bin:/usr/texbin:
   /Users/Lukescomp/Desktop/SISG_AQG_2015/bin/
$ # Check to see if the directory containing your binaries is in the PATH
$ # To make sure our binaries are all working execute the following at the terminal
$ pedstats
$ qtdt
$ plink
$ plink2
$ gcta
$ # Let's start where we would like to be
$ cd ~/Desktop/SISG_AQG_2015
```

Listing 1 Basic command line executions for Mac and Linux

IMPORTANT - For Windows users the navigation is a little bit different and the execution of commands is outlined in the below listing.

Windows

To open a command prompt in Windows, go to `Start` and then click on `Run` and typing in `CMD`. In Windows 7, just click on `Start` and begin typing `CMD`. In Windows 8, you can just right-click on the `Start` button and choose `Command Prompt`. Once you have a prompt open attempt to execute the following commands.

```
$ # Take a look at your home directory. This will be important for building other directories
$ echo %HOMEPATH%
  \Users\presentation
$ # IMPORTANT - THIS WAS MY HOME PATH YOURS WILL BE DIFFERENT
$ # Change your directory so that your are in the SISG_AQG_2015 folder
$ cd C:\Users\presentation
$ cd Desktop\SISG_AQG_2015
$ # The current directory is always displayed before the command start >
$ # When navigating through directories use the TAB key to complete a half typed path.
$ # For example, once you are in the SISG_AQG_2015 try
$ cd pra
$ # and then hit TAB to try and complete the line. Navigate to practical_1 using this trick
$ # Navigate back to the parent SISG_AQG_2015 by
$ cd ../
$ # Once you are back in the SISG_AQG_2015 type
$ dir
$ # This lists the contents of the folder.
$ # IMPORTANT - WE ARE GOING TO SET UP THE PATH FOR THE BINARIES FOR THE REST
$ # OF THE TUTORIAL. MAKE NOTE OF YOUR HOMEPATH FROM ABOVE
$ setx path "%path%;%HOMEPATH%\Desktop\SISG_AQG_2015\bin"
$ # To make sure our binaries are all working, make sure you are in the SISG_AQG_2015 folder and type the following at
$ # the command prompt. You should see the header printed to the screen for each of the programs
$ pedstats
$ qtdt
$ plink
$ plink2
$ gcta
$ cd %HOMEPATH%\Desktop\SISG_AQG_2015
```

Listing 2 Basic command prompt for Windows

For Windows users the following practicals are written with Linux or Unix users in mind. It will be up to you to be able to adapt the path structure, which will be very similar usually a / replacing the but we will be here to help.

Installation and opening of R

If you haven't already installed R please go to the <`http://cran.wustl.edu/`> and choose the download as per your operating system. Follow the installation instructions. Once installed run the application. You should now have an R console open. You can choose to run all R commands in these practicals using this window but it is best if you save you work in a script for easy execution later. I have prepared some template scripts in each of the `practical_` folders. In the `practical_1` folder there should be a `sisg_aqg_2015_p1.R` script template, open this and type you commands here. Remember to save your work intermittently.

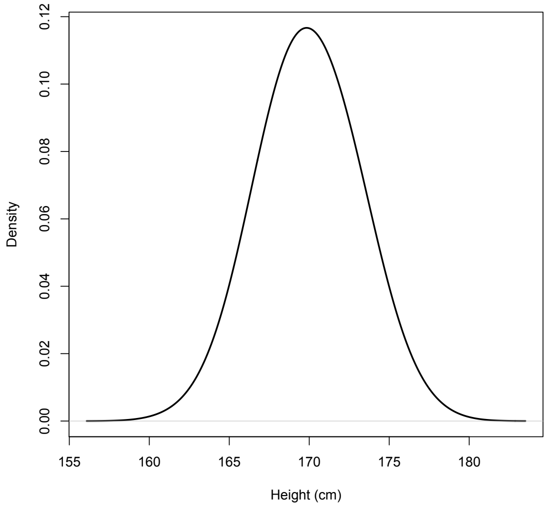
R functions

We begin with some useful R functions. In this document the R console call shall be indicated with the greater than symbol >. Any subsequent output from the call will appear below the R command with no >. Comment lines will begin with a # and are used to assist in the readability and understanding of the code. If you are unfamiliar with R, then the `?` command is useful; follow the `?` with any function name (e.g., `?rnorm`) and a manual on the implementation of this function will open. If you would like to copy from the listing windows throughout these documents straight to R then it is advised to do so line-by-line as formatting will be copied as well if the whole listing is highlighted.

The normal distribution is a key continuous distribution and thus we will begin with a simulation from this distribution via the R function `rnorm`.

```
# Take a look at the function structure and arguments
> args(rnorm)
  function (n, mean = 0, sd = 1)
  NULL
# Simulate a pseudo phenotype called height. Mean 170 cm and SD 2.8 cm
> height <- rnorm(1000, 170, 2.8)
# Calculate the mean and standard deviation of the simulated phenotype and plot
> mean(height)
  170.018
> sd(height)
  2.84313
> # REMEMBER that the your mean and sd will be different to mine as these are pseudo-random draws from
> # the distribution. Please watch the copying of this line into R
> plot(density(height, adjust = 3), main = ’’")
```

The final line produces the below plot of the simulated height distribution.



The *rnorm* function can be used to simulate a phenotype and SNP data.

```
> y.means  <- c(5, 15, 20)
> y.sd     <- 5
> # Generate a SNP. Sample from the binomial distribution with success probability 0.4
> # Success probability corresponds to the MAF
> snp      <- rbinom(1000, 2, 0.4)
> # Generate Y given the SNP and bind to data frame
> y        <- rnorm(1000, y.means[factor(snp)], y.sd)
> snp.data <- data.frame(cbind(y, snp))
```

The ggplot2 library can be used over the standard plotting library to generate superior graphics. If the ggplot2 library is not installed then it can be included via the `install.packages("ggplot2")` command. This package uses an unfamiliar syntax when compared with the standard plotting library and may take some getting used to if you haven’t used it before.

```
> # Plot the phenotype as a factored boxplot of the underlying genotype counts
> # Load ggplot2
> library(ggplot2)
> p <- ggplot(snp.data, aes(x = factor(snp), y = y))
> p + geom_boxplot() + xlab("Reference allele count")
```

Listing 3 Using `ggplot`

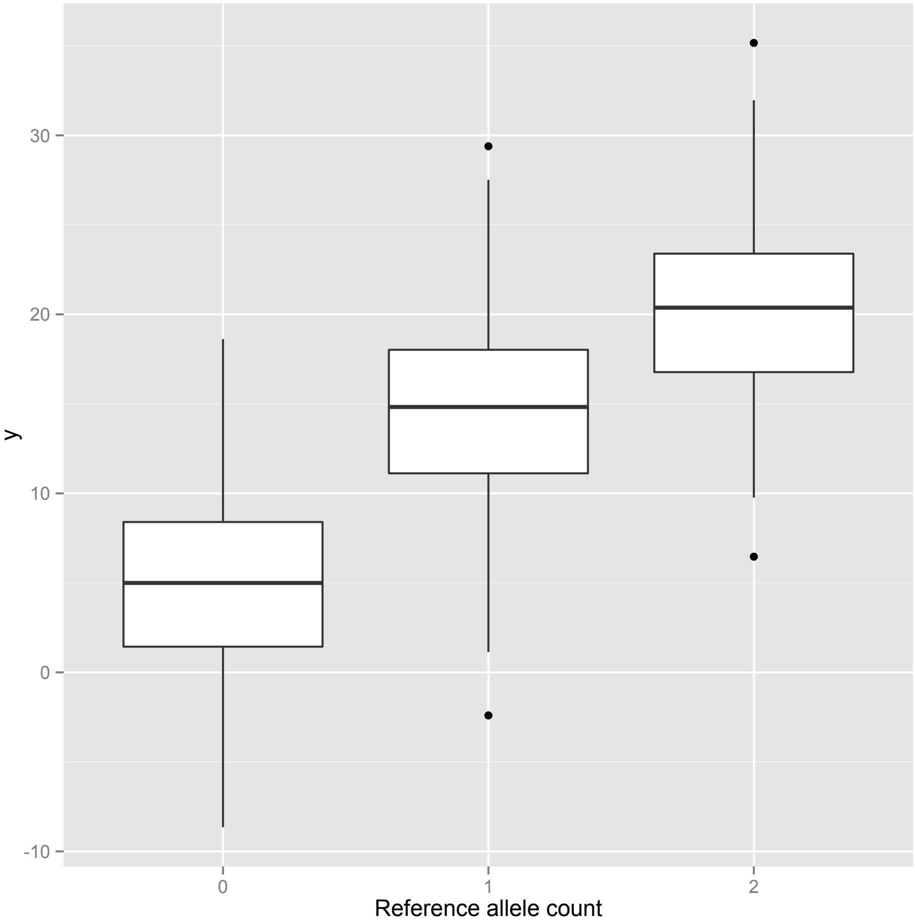


Figure 1 Boxplot of phenotype subsetted by the reference allele count.

The multivariate extension of `rnorm` is `mvrnorm` and is contained in the `MASS` package. Naturally, this function can be used to simulate data from a multivariate normal distribution.

```
> library(MASS)
> args(mvrnorm)
  function (n = 1, mu, Sigma, tol = 1e-06, empirical = FALSE, EISPACK = FALSE)
  NULL
> # Simulate a correlated height and bmi pseudo phenotype
> # Generate a set of means and a covariance matrix. Covariance of 0.1
> means   <- c(170, 24)
> cov.mat <- matrix(c(1, 0.1, 0.1, 1), nrow = 2, ncol = 2)
> height.bmi <- data.frame(mvrnorm(1000, means, cov.mat))
> # Give the data frame column headings. This is a requirement for ggplot
> # Please watch quotes
> names(height.bmi) <- c("height", ’bmi")
> # Following output won’t match exactly as it is pseudo-random
> height.bmi[1:10, ]
    height      bmi
  167.7047 24.56114
  170.5013 23.45098
  170.8352 25.21894
  169.7476 26.58169
  170.9120 23.17956
  171.6698 24.32220
  169.2741 23.84742
  170.9006 24.58450
  169.2104 25.13864
  170.6774 23.83173
> # Look at the mean and covariance to check the function
> colMeans(height.bmi)
     height       bmi
  169.99983  24.06207
> cov(height.bmi)
              height        bmi
  height 1.02247286 0.08060026
  bmi    0.08060026 1.01609569
> cor(height.bmi$height, height.bmi$bmi)
> cor(height.bmi$height, height.bmi$bmi)
  [1] 0.07907574
> # Plot the joint distribution of the variables
> p <- ggplot(height.bmi, aes(x = height, y = bmi))
> p + geom_point() + geom_smooth(method = "lm")
```

Simulating data to mimic twins

We would like to simulate 300 pairs of twins with a mean height of 170 cm and a SD of 2.8 cm. We also induce a correlation between the two sets of twins.

```
# Generate the mean vector and covariance matrix
> twin.means   <- c(170, 170)
> twin.cov.mat <- 2.8 * matrix(c(1, 0.7, 0.7, 1), nrow = 2,
                               ncol = 2)
> twins        <- data.frame(mvrnorm(300, twin.means,
                             twin.cov.mat))
> names(twins) <- c("twin1", "twin2")
> twins[1:4, ]
     twin1    twin2
  170.8381 169.5010
  172.6826 171.7345
  172.9329 173.7101
  167.7240 171.2131
> cor(twins$twin1, twins$twin2)
  0.6848547
> cov(twins)
           twin1    twin2
  twin1 2.302733 1.695159
  twin2 1.695159 2.660606
```

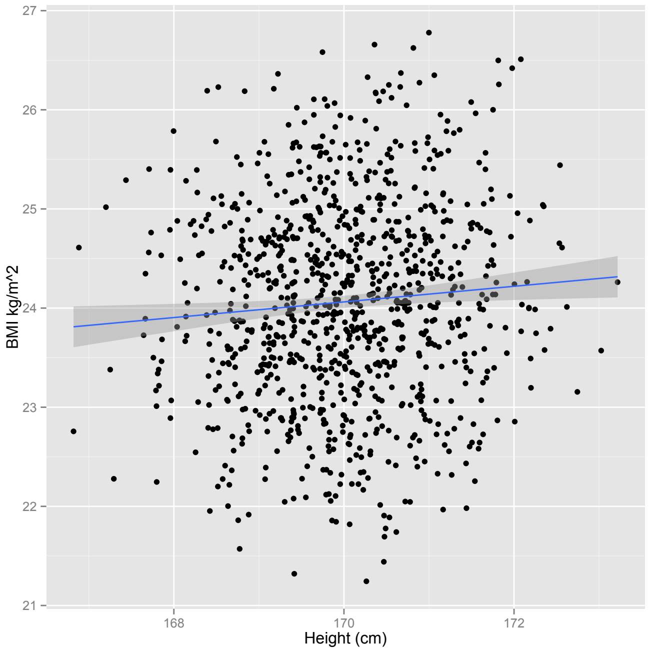
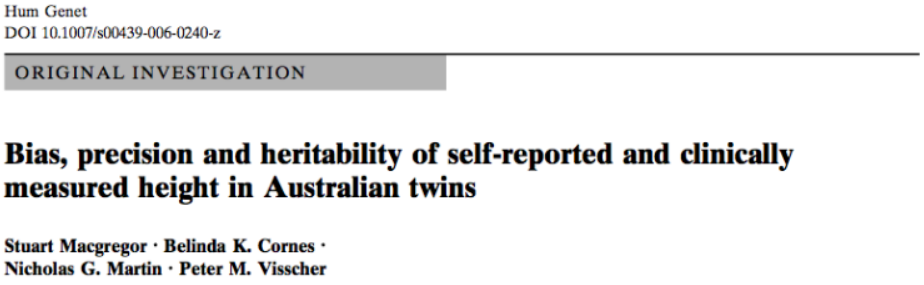


Figure 2 Plot of joint distribution of height and BMI

Real data from twins

The twin data that we will use, was part of the following publications (Macgregor et al., 2006) and (Liu et al., 2010).



Hum Genet
DOI 10.1007/s00439-006-0240-z

ORIGINAL INVESTIGATION

Bias, precision and heritability of self-reported and clinically measured height in Australian twins

Stuart Macgregor · Belinda K. Cornes ·
Nicholas G. Martin · Peter M. Visscher

Genome-Wide Association Study of Height and Body Mass Index in Australian Twin Families

Jimmy Z. Liu,[1] Sarah E. Medland,[1] Margaret J. Wright,[1] Anjali K. Henders,[1] Andrew C. Heath,[2] Pamela A. F. Madden,[2] Alexis Duncan,[2] Grant W. Montgomery,[1] Nicholas G. Martin[1] and Allan F. McRae[1*]

We are going to use these data to look at some of the ideas presented in the lectures. First, these data must be read into R with the key read function `read.table`, which is a base R function.

```
> # Make sure you are in the correct directory if you haven't done so already. Please watch out for copying $~$ in the next line
> setwd("~/Desktop/SISG_AQG_2015/")
> twin.data <- read.table("practical_1/data/twin_height_bmi.txt", header = T)
> dim(twin.data)
  5438    7
> head(twin.data)
  dob  ht_t1   bmi_t1  ht_t2      bmi_t2 sex twin
 1902 165.10 25.86451 173.00 24.05693475   1   MZ
 1903 175.26 22.74054 143.00 33.74248129   1   MZ
 1910 164.00 28.62879 165.00 27.54820937   1   MZ
 1906 167.64 21.46733 167.00 18.28677973   1   MZ
 1911 168.00 23.03005 167.64 19.94805977   1   MZ
 1912 172.72 25.15385 166.00 28.30599506   1   MZ
```

With these data we will be interested in answering the following questions:

- What is the phenotypic correlation for both height and BMI for the dizygotic (DZ) and monozygotic (MZ) twins?
- Is the phenotypic correlation the same for both male and females?
- What is the estimate of the heritability (narrow sense) given we have estimated these correlations?

It is expected that you try to answer these questions using the R functions introduced above. The below code is a partial solution to the above questions and uses some key functions that you may want to use to answer the above questions. Run this code and attempt to understand the output but we will explore these concepts in the next section with the `pedstats` program.

```
> # Monozygotic twins
> twin.mz <- twin.data[twin.data$twin == "MZ", ]
> cor(twin.mz $ht_t1, twin.mz $ht_t2)
  0.918698
> p <- ggplot(twin.mz, aes(x = ht_t1, y = ht_t2))
> p + geom_point() + geom_smooth(method = "lm")
> # Dizygotic twins
> twin.dz <- twin.data[twin.data$twin == "DZ", ]
> cor(twin.dz$ht_t1, twin.dz$ht_t2)
  0.7503631
> p <- ggplot(twin.dz, aes(x = ht_t1, y = ht_t2))
> p + geom_point() + geom_smooth(method = "lm")
```

Pedigree statistics

We will further explore the resemblance between relatives by looking at pedigree data using the software `pedstats` (Wigginton and Abecasis, 2005). Open the terminal or command prompt and execute the following basic option command.

```
> $ pedstats -d practical_1/data/height_bmi.dat -p practical_1/data/height_bmi.ped
```

Listing 4 System run of pedstats

The command in listing 4 will only run if you are in the parent `SISG_AQG_2015`. Some other frequently used flags for this software are `--pair`, `--byFamily`, `--bySex`, and `--pdf` and yes they mix − flags with −− flags perhaps to distinguish optional extras from standard commands.

The `pedstats` program requires two files: the `.ped` file that contains the relationships between individuals along with phenotypic data and optional genotype data. The second file is the `.dat` file, which describes the contents of the pedigree file. For further explanation of the file formats, the author website `http://csg.sph.umich.edu//abecasis/PedStats/` gives a good explanation. Examples of both of these can be explored with R.

```
> # Example of a .ped data file
> ped <- read.table("practical_1/data/height_bmi.ped", header = F)
> dim(ped)
  47633    11
> names(ped) <- c("FAMILY", "PERSON", "FATHER", "MOTHER", "SEX")
> colnames(ped)[6:11] <- c("TWIN_STATUS", "HEIGHT","NI_1", "BMI",
                          "NI_2", "NI_3")
> head(ped)
    FAMILY PERSON FATHER MOTHER SEX TWIN_STATUS  HEIGHT  NI_1    BMI  NI_2   NI_3
      1      1      3      4   1          MZ     179.000 0.082 18.814 2.935 -1.440
      1      2      3      4   1          MZ     180.340 0.373 24.408 3.195 -0.252
      1      3      0      0   1           0           x     x      x     x      x
      1      4      0      0   2           0           x     x      x     x      x
      2      5      0      0   2           0     167.640 0.682 23.841 3.171 -0.186
      1      6      3      4   1           0     185.420 0.993 24.936 3.216  0.008
> # Example of a .dat data file
> dat <- read.table("practical_1/data/height_bmi.dat")
> dim(dat)
   7 2
> head(dat)
  V1            V2
  Z       Zygosity
  T         height
  T height_z_score
  T            BMI
  T         logBMI
  T    BMI_z_score
```

We will explore the data a little more with some R functions from the `kinship2` package.

```
> # Install the kinship package
> install.packages("kinship2")
> library("kinship2")
# Generate a pedigree plot of the first two families
> ped.sub <- ped[1:12, ]
> p <- pedigree(ped.sub$PERSON, ped.sub$FATHER,
                ped.sub$MOTHER, ped.sub$SEX)
> plot.pedigree(p)
```

Exercise using `pedstats`

If you had trouble executing the command in listing 4, then you should make sure that you have set your working directory correctly and that everything you need to execute the command is there i.e., `.ped`, the `.bed` files, and the `pedstats` binary are where they should be. IMPORTANT - `pedstats` does not allow a specification of an 'out' directory and thus all output files will be dumped in the `SISG_AQG_2015`

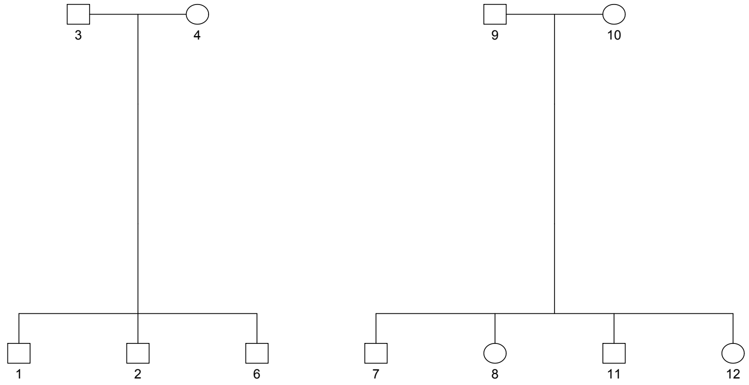


Figure 3 Pedigree plot of first two families from `.ped` file

folder. Once you are finished running one command check in this folder for output and then copy to your `practical_1/results` folder.

Attempt to answer the following questions using `pedstats`.

- What is the full sib correlation for height?
- What is an estimate of the heritability from this correlation?
- Repeat for BMI
- Repeat both of these with z-scores
- Why are the sibling correlation (subsequently the heritability estimates) results different for height but similar for BMI?

Some hints include, try using `--bySex` and compare with previous results. Additionally do a check with `--pdf`, which saves a set of plots to the current working directory. Listing 5 gives an example of using these extra functions.

```
$ pedstats -d practical_1/data/height_bmi.dat -p practical_1/data/height_bmi.ped --bySex
```

Listing 5 Example commands for pedstats

Putting it all together

- Plot the pairwise phenotypic correlation of relatives as a function of their additive genetic relationship (twice the kinship coefficient) for height and BMI
- What conclusions can be drawn about the resemblance between relatives from genetic and non-genetic factors for height and BMI?

References

Jimmy Z Liu, Sarah E Medland, Margaret J Wright, Anjali K Henders, Andrew C Heath, Pamela AF Madden, Alexis Duncan, Grant W Montgomery, Nicholas G Martin, and Allan F McRae. Genome-wide association study of height and body mass index in australian twin families. *Twin Research and Human Genetics*, 13(02):179–193, 2010.

Stuart Macgregor, Belinda K Cornes, Nicholas G Martin, and Peter M Visscher. Bias, precision and heritability of self-reported and clinically measured height in australian twins. *Human genetics*, 120(4):571–580, 2006.

R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2015. URL `http://www.R-project.org`.

Janis E Wigginton and Gonçalo R Abecasis. Pedstats: descriptive statistics, graphics and quality assessment for gene mapping data. *Bioinformatics*, 21(16):3445–3447, 2005.