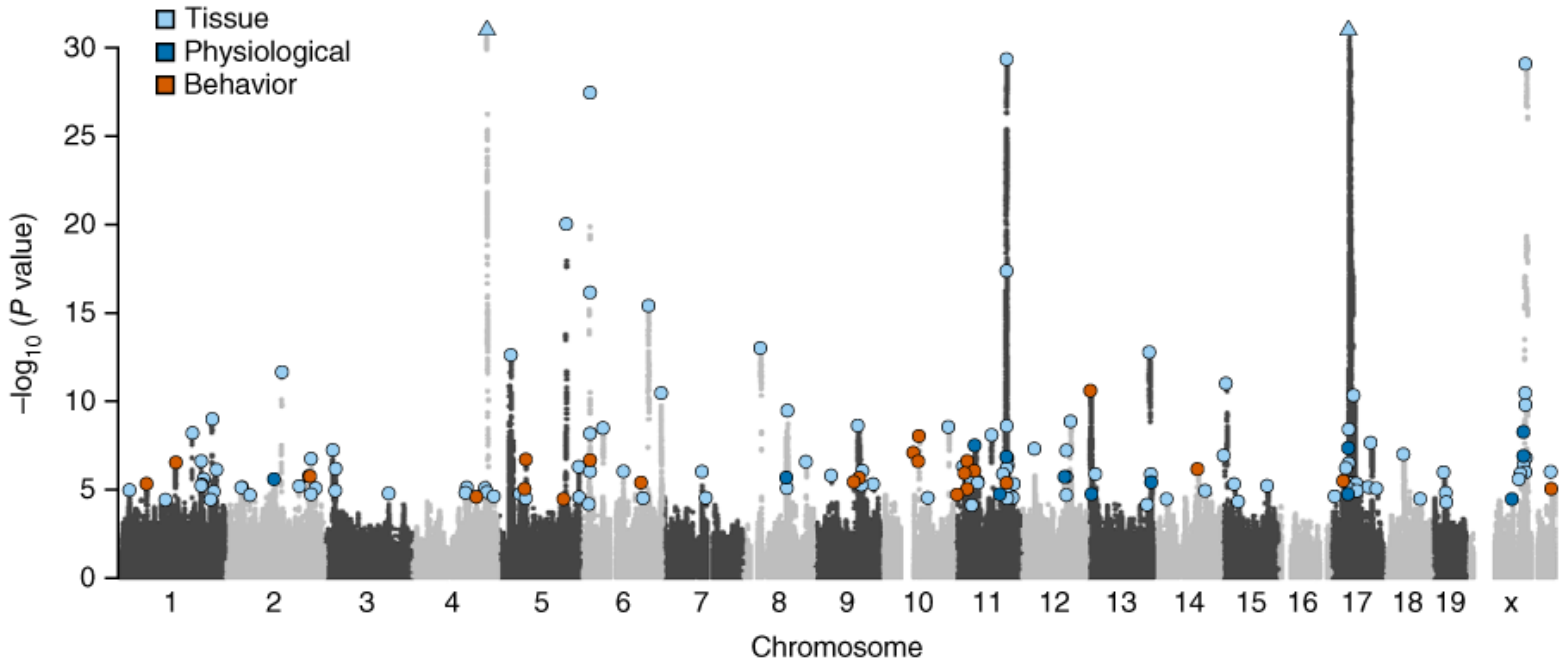
Analysis of Genetic Data 2: Mapping Genome-Wide Associations

Peter Carbonetto

Research Computing Center and the Dept. of Human Genetics
University of Chicago



Aims of workshop

1. Implement the basic steps of a genome-wide association analysis.
2. Understand how phenotype and genotype data from a genome-wide association study (GWAS) are encoded in computer files.
3. Understand some of the benefits (and complications) of using linear-mixed models (LMMs) for GWAS.
4. Use command-line tools to view and manipulate the phenotype and genotype data.
5. Learn through “live coding”—this includes learning from our mistakes!

Our research task

Our goal is identify and interpret genetic loci contributing to tibia bone development in mice.

1. We will use data from this study:
   - Nicod *et al* (2016). "Genome-wide association of multiple complex traits in outbred mice by ultra-low-coverage sequencing." *Nature Genetics* **48:** 912–918.
2. To map tibia development genes, we will assess support for statistical association between tibia length (given sex and body weight) and 353,697 SNP genotypes. Specifically, we will compute *p*-values to quantify support for association.
3. We will visualize the "association signal" to identify candidate genes for tibia bone development.
4. We will see that <10% of the effort is running GEMMA; *we will spend most of our time preparing the data for GEMMA, and examining the GEMMA output in R.*

Some background on the study

- Mice are from an outbred population called “Carworth Farms White” (CFW).
- 200 measurements (e.g., blood concentrations, muscle weights) were recorded in columns of a large table. We will analyze one column—tibia length.
- Genotypes were obtained from *low-coverage* DNA sequencing (to save money!). Therefore, genotype data is lower quality.
- Scale of data is comparable to human GWAS: 2,000 samples and 350,000 SNPs.
- Linkage disequilibrium (LD) decays faster than many other mouse populations, but not as rapidly as humans, so mapping resolution will not be as good as human GWAS.
- One advantage over human GWAS: *data are publicly available.*

It's your choice

Your may choose to...

- Work through the examples on the RCC cluster.
- Work through the examples on your laptop.
- Pair up with your neighbour.
- Follow what I do on the projector.

However,

1. A few of the examples will only run on the RCC cluster.
2. The examples may not produce the exact same result on your laptop.

Software tools we will use today

1. GEMMA
2. R
3. R packages `stringr`, `data.table`, `ggplot2` and `cowplot`.
4. Basic shell commands such as `wc` and `cat`.

Outline of workshop

1. Initial setup.
2. Download genotype data.
3. Prepare phenotype data.
4. Prepare genotype data.
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. Run an LMM-based association analysis in GEMMA, and compare against the initial analysis.
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

Outline of workshop

1. **Initial setup.**
2. Download genotype data.
3. Prepare phenotype data.
4. Prepare genotype data.
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. Run an LMM-based association analysis in GEMMA, and compare against the initial association analysis.
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

Initial setup (part 1)

- WiFi
- Power outlets
- YubiKeys
- Pace, questions (e.g., keyboard shortcuts).

Initial setup (part 2)

Download the workshop packet to your laptop.

- URL: `https://github.com/rcc-uchicago/genetic-data-analysis-2`
- Open **slides.pdf** from the **docs** folder. This is useful for viewing and copying the code from the slides.
- We may also look at some of the files in the **code** folder. The code can be viewed with your favourite text editor, or browsed on GitHub.

Initial setup (part 3)

If you are using the RCC cluster, set up your cluster computing environment:

1. Connect to midway2.
   - ▷ See `https://rcc.uchicago.edu/docs/connecting`.
   - ▷ If you are using ThinLinc, use the “clipboard” to copy & paste.
2. Request a midway2 (“Broadwell”) compute node with 4 CPUs and 8 GB of memory:

```
sinteractive --partition=broadwl --mem=8G \
  --cpus-per-task=4 --time=3:00:00
```

Initial setup (part 4)

If you are using the RCC cluster, also download the workshop packet to your home directory on the cluster.

- URL: `https://github.com/rcc-uchicago/genetic-data-analysis-2`

You can run these commands to download the workshop packet:

```
cd ~
git clone https://github.com/rcc-uchicago/
  genetic-data-analysis-2.git
```

Initial setup (part 5)

Let's start up R and make sure you can display graphics from R. On the RCC cluster, load R, then start up an interactive R environment:

```
module load R
R --no-save
```

Check which version of R you are running (hopefully version 3.4.0 or greater):

```
version$version.string
```

Here is a simple test plot:

```
plot(cars$dist,cars$speed)
```

You should see a scatterplot. If not, you should start a fresh R session in a new midway2 connection (**ThinLinc is the most reliable approach**). *Note:* run `q()` at any time to exit R.

What's in the workshop packet

```
genetic-data-analysis-2
  /bin      # All executables are stored here.
  /code     # Source code used in analyses.
  /data     # Raw and processed data.
  /docs     # Additional workshop materials.
  /output   # All results are stored here.
```

Outline of workshop

1. Initial setup.
2. **Download genotype data.**
3. Prepare phenotype data.
4. Prepare genotype data.
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. Run an LMM-based association analysis in GEMMA, and compare against the initial association analysis.
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

Download data (part 1)

The genotype files are too large to fit in the git repository. So we need to download them separately.

- URL: `http://mtweb.cs.ucl.ac.uk/dosages`
- Download files for chromosomes 1–19 into the `data` folder.
- You may find it easier to run the bash script provided in the `data` folder:

  ```
  cd data
  bash download_genotypes.sh
  ```

  This script requires the `wget` program.

Note: If you downloaded the genotype files before the workshop, please move these files into the ‘data folder.

Download data (part 2)

After following these steps, the contents of your `data` directory should look like this:

```
$ cd data
$ ls -1
README
CFW_covariates.txt
CFW_measures.txt
download_genotypes.sh
chr1.prunedgen.final.maf001.0.98.RData
chr2.prunedgen.final.maf001.0.98.RData
...
chr19.prunedgen.final.maf001.0.98.RData
```

The data we will use in our analyses

- **CFW_measures.txt:** a large table with 200 columns (phenotypes) and 2,117 rows (CFW mice).
- **CFW_covariates.txt:** covariate data for 2,117 CFW mice.
- **listof1934miceusedforanalysis.txt:** ids of the 1,934 mice used in the final analyses of Nicod *et al* (2016).
- **chr*.prunedgen.final.maf001.0.98.RData:** genotypes for SNPs on chromosomes 1–19 used in the final analysis of Nicod *et al* (2016).

See the `README` in the `data` folder for more details.

Outline of workshop

1. Initial setup.
2. Download genotype data.
3. **Prepare phenotype data.**
4. Prepare genotype data.
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. Run an LMM-based association analysis in GEMMA, and compare against the initial association analysis.
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

Outline of phenotype data preparation

1. Examine phenotype and covariate data from the terminal.
2. Examine phenotype and covariate data in R.
3. Verify phenotype residuals.
4. Format phenotype and covariate data for GEMMA.

Explore the phenotype data from the terminal

First, let's take a quick look the phenotype data using command-line tools:

```
less -S CFW_measures.txt
wc -l CFW_measures.txt
```

Likewise, the covariate data:

```
less -S CFW_covariates.txt
wc -l CFW_covariates.txt
```

Explore the phenotype data in R (part 1)

Move to the **code** folder in the git repository, then start up R:

```
cd code
R --no-save
```

Before continuing, check your working directory:

```
getwd()   # Should be .../code
```

Explore the phenotype data in R (part 2)

Load the phenotype and covariate data, and merge these data into a single table ("data frame"), dropping the redundant second "id" column:

```
source("functions.R")
dat1 <- read.pheno("../data/CFW_measures.txt")
dat2 <- read.pheno("../data/CFW_covariates.txt")
pheno <- cbind(dat1,dat2[-1])
```

Take a quick glance at the data:

```
nrow(pheno)
ncol(pheno)
sort(names(pheno))
pheno[1:5,1:5]
```

Explore the phenotype data in R (part 3)

Select the same 1,934 samples that were used in Nicod *et al* (2016). First, we load the list of ids:

```
x <- "../data/listof1934miceusedforanalysis.txt"
ids <- read.table(x,stringsAsFactors = FALSE)
ids <- ids[[1]]
```

Then we select the rows of the table that match up with the ids:

```
rows  <- match(ids,pheno$Animal_ID)
pheno <- pheno[rows,]
nrow(pheno)
```

Explore the phenotype data in R (part 4)

Take a closer look at the data for the phenotype we are interested in—tibia length (units = mm):

```
summary(pheno$Tibia.Length)
```

Plot the distribution of tibia length (`plot.dist` was defined in `functions.R`):

```
library(ggplot2)
library(cowplot) # Optional.
p1 <- plot.dist(pheno$Tibia.Length)
print(p1)
```

Fix the axis labels, following Best Practices:

```
p1 <- p1 + labs(x = "tibia length (mm)",
                y = "number of mice")
print(p1)
```

Explore the covariate data in R (part 1)

The authors recommend using body weight as a covariate in the association analysis:

```
summary(pheno$Weight.Diss)
```

Examine the relationship between tibia length and body weight:

```
p2 <- draw.scatterplot(pheno$Weight.Diss,
                       pheno$Tibia.Length) +
  labs(x = "body weight (g)",
       y = "tibia length (mm)")
print(p2)
```

Explore the covariate data in R (part 2)

The authors also recommend using sex as a covariate in the association analysis:

```
pheno <- transform(pheno,Sex = factor(Sex))
summary(pheno$Sex)
```

A boxplot is the standard approach to visualize the relationship between a continuous variable (tibia length) and a categorical variable (sex):

```
p3 <- draw.boxplot(pheno$Sex,
                   pheno$Tibia.Length) +
  labs(x = "sex",y = "tibia length (mm)")
print(p3)
```

Verify the phenotype residuals

Compute the phenotypes after removing the linear effects of sex and body weight:

```
fit <- lm(Tibia.Length ~ Sex + Weight.Diss,pheno)
r   <- resid(fit)
```

Plot the distribution of the residuals:

```
plot.dist(r) + labs(x = "residuals")
```

Is the normal distribution a good fit for the residuals? We can verify this more rigorously:

```
check.normal.quantiles(r)
draw.qqplot(r)
```

Save phenotype data for GEMMA

Now that we have explored and verified the phenotype data, we are ready to save the data in a format usable by GEMMA. First write the phenotype data to `pheno.txt`:

```
write.table(pheno$Tibia.Length,
            "../data/pheno.txt",
            row.names = FALSE,
            col.names = FALSE)
```

Save covariate data for GEMMA

Write the covariate data to file `covariates.txt` in the format used by GEMMA. This requires that we convert sex to a numeric value:

```
sex <- as.numeric(pheno$Sex) - 1
unique(sex)
write.table(data.frame(1,sex,pheno$Weight.Diss),
            "../data/covariates.txt",sep = " ",
            row.names = FALSE,col.names = FALSE)
```

Also, a column of ones is needed for an intercept in the GEMMA regression analysis.

Preparing the phenotype data: take-home points

Most association analyses will involve these steps:

1. Carefully checking phenotype and covariate data for possible issues, and resolving these issues (e.g., filtering out problematic data, transforming phenotypes).
2. Aligning phenotypes to genotypes.
3. Reformatting the data.

In this tutorial, we performed these steps in R.

Outline of workshop

1. Initial setup.
2. Download genotype data.
3. Prepare phenotype data.
4. **Prepare genotype data.**
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. Run an LMM-based association analysis in GEMMA, and compare against the initial association analysis.
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

Prepare the genotype data for GEMMA (part 1)

Processing the genotype data is more complicated, so I have provided a script to create, for each chromosome, a SNP annotation file and a genotype file in the BIMBAM format. Run this script in R:

```
source("format.genotypes.for.gemma.R")
```

- This will take a few minutes to run. Once it is done, exit R.

Let's look at the GEMMA files:

```
less chr01.map.txt
wc -l chr01.map.txt
less -S chr01.geno.txt
wc -l chr01.geno.txt
```

What do the numeric values in `chr01.geno.txt` *represent?*

Prepare the genotype data for GEMMA (part 2)

The genotype data need to be combined into one file. This can be easily done with the `cat` command, which combines by lines (rows):

```
cat chr*.geno.txt > geno.txt
head geno.txt | less -S
tail geno.txt | less -S
wc -l geno.txt
```

We do the same for the SNP annotations:

```
cat chr*.map.txt > map.txt
wc -l map.txt
```

Why didn't we combine the genotypes inside the R script?

Preparing the genotype data: take-home points

- Typically this is much more work—*we were fortunate that the authors provided the post-processing, post-QC (quality control) genotype data.*
- A **critical step** in this script was to make sure that the samples in the genotype and phenotype files are listed in the same order. *A common mistake is to save the phenotype and genotype samples in different orders, which will lead to an incorrect association analysis!*
- Why is it easier to make this mistake in a GEMMA analysis?

Outline of workshop

1. Initial setup.
2. Download genotype data.
3. Prepare phenotype data.
4. Prepare genotype data.
5. **Run a basic association analysis in GEMMA, and assess quality of results.**
6. Run an LMM-based association analysis in GEMMA, and compare against the initial analysis.
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

Set up GEMMA

Download the GEMMA 0.96 to the `bin` folder.

- URL: `https://github.com/genetics-statistics/GEMMA/releases`

On the RCC cluster, you can run these commands to install the GEMMA 0.96 Linux binary, and check that it works:

```
cd bin
wget http://bit.ly/2I67fBV -O gemma.linux.gz
gunzip gemma.gz
mv gemma.linux gemma
chmod 700 gemma
./gemma
```

Basic association analysis in GEMMA (part 1)

Now that we have installed GEMMA, and we have prepared the phenotype and genotype data in the formats accepted by GEMMA, we are now ready to run our first association analysis.

```
cd data
../bin/gemma -p pheno.txt -c covariates.txt \
  -a map.txt -g geno.txt -notsnp -lm 2 \
  -outdir ../output -o tibia
```

This should take a few minutes to complete.

- The `-notsnp` option is included here to prevent any automatic processing of the genotype data.

Basic association analysis in GEMMA (part 2)

This GEMMA command creates two files:

- **tibia.log.txt:** A brief summary of the association analysis.
- **tibia.assoc.txt:** The full table of association results (*p*-values, effect size estimates, *etc*).

Use `less -S` or your favourite text editor to inspect these files.

Assess “genomic inflation” (part 1)

The q-q plot is commonly used to assess inflation of small *p*-values, which could indicate a high rate of false positive associations. This test has many well-known problems, but is nonetheless useful as a simple diagnostic.

- Since we will be jumping back and forth between R and the command-line shell, it is better at this point to set up a second session, and run R separately in this session (as before, make sure that plotting to the screen works).
- **ThinLinc** is the recommended choice for running R on midway2.
- Since the computation in R will not be intensive, you do not need to run `sinteractive`; running R from a login node is okay.

Assess “genomic inflation” (part 2)

In R, use `read.table` to load the results of the GEMMA association analysis:

```
getwd() # Should be .../code
gwscan <- read.table("../output/tibia.assoc.txt",
                     as.is = "rs",header = TRUE)
head(gwscan)
nrow(gwscan)
```

Assess “genomic inflation” (part 3)

Plot the observed *p*-values against the expected *p*-values under the null distribution:

```
library(ggplot2)
library(cowplot) # Optional.
source("functions.R")
plot.inflation(gwscan$p_lrt)
```

Next, we will attempt an LMM-based association analysis, and compare this q-q plot against the LMM q-q plot.

- The LMM reduce inflation due to population structure and “hidden” relatedness.
- *I recommend taking a screenshot of this plot before continuing.*

Outline of workshop

1. Initial setup.
2. Download genotype data.
3. Prepare phenotype data.
4. Prepare genotype data.
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. **Run an LMM-based association analysis in GEMMA, and compare against the initial analysis.**
7. Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.

LMM association analysis in GEMMA

First, generate a “realized relatedness” matrix from the (centered) genotypes. This may take 10–15 minutes.

```
cd data
../bin/gemma -p pheno.txt -g geno.txt \
  -a map.txt -gk 1 -outdir ../output \
  -o tibia
```

Next, fit an LMM separately for each SNP:

```
../bin/gemma -p pheno.txt -c covariates.txt \
  -a map.txt -g geno.txt -notsnp \
  -k ../output/tibia.cXX.txt \
  -lmm 2 -outdir ../output -o tibia-lmm
```

This may take over 30 minutes to run, *so I have provided a compressed file in the* `output` *directory containing the result of this command.*

Assess “genomic inflation” in LMM analysis

In your R session, follow the same steps as before to load the GEMMA results and create a genomic inflation plot:

```
gwscan <-
  read.table("../output/tibia-lmm.assoc.txt",
             as.is = "rs",header = TRUE)
plot.inflation(gwscan$p_lrt)
```

Compare against the previous inflation plot—*did the LMM reduce inflation of p-values?*

Association analysis: take-home points

- LMMs are now one of the most widely used approaches to reduce inflation of false positive associations due to population structure and hidden relatedness.
- However, LMMs can be computationally intensive, especially for data sets with many samples.
- Another issue: LMMs can *overcorrect* inflation—this is particularly a problem in mouse GWAS due to long-range LD patterns. One solution is a “leave-one-chromosome-out” strategy (this was used in the Nicod *et al* paper).

Outline of workshop

1. Initial setup.
2. Download genotype data.
3. Prepare phenotype data.
4. Prepare genotype data.
5. Run a basic association analysis in GEMMA, and assess quality of results.
6. Run an LMM-based association analysis in GEMMA, and compare against the initial analysis.
7. **Visualize the genome-wide association signal in R to (a) map tibia loci and (b) identify plausible tibia genes.**

Visualizing the association signal

1. Summarize the genome-wide association signal for tibia length.
2. Take a closer look at the strongest association signal.
3. Based on the association signal, identify candidate genes.
4. *Optional:* Visualize the relationship between tibia length and the top SNP association.

Plot genome-wide scan for tibia length

Using GEMMA, we computed association *p*-values at 353,697 SNPs on chromosomes 1–19. Now let's plot these *p*-values to get a visual summary of the association results.

- I've written a function that uses `ggplot2` to create a "Manhattan plot" showing the results of our genome-wide association analysis.

Create the "Manhattan plot":

```
plot.gwscan(gwscan)
```

On which chromosomes do we observe the strongest association signals?

Examine the association signal on chr. 6 (part 1)

Above, our plot gave us a *genome-wide* overview of the regions of the genome contributing to variation in BMD.

- Next, let's try to get a finer-scale view of the strongest association signal on chromosome 6.

Create a data frame with the chromsome 6 association results.

```
gwscan.chr6 <- subset(gwscan,chr == 6)
```

Next, plot the *p*-values against the base-pair position of the SNPs.

```
plot.region.pvalues(gwscan.chr6) +
  scale_x_continuous(breaks = seq(0,200,10))
```

Examine the association signal on chr. 6 (part 2)

It appears that the strongest association signal is isolated to positions of 145 Mb and greater. Let's zoom in on that region:

```
dat <- subset(gwscan.chr6,ps > 145e6)
plot.region.pvalues(dat,size = 2) +
  scale_x_continuous(breaks = seq(145,150,0.5))
```

Let's identify the SNP with the strongest association to tibia length:

```
i <- which.min(gwscan.chr6$p_lrt)
gwscan.chr6[i,]
```

Examine the association signal on chr. 6 (part 3)

Based on the pattern of p-values, what would you propose as an approximate candidate region?

- *Exercise:* Use the UCSC Genome Browser (`http://genome.ucsc.edu`) to identify possible tibia bone development genes.

Visualize genotype-phenotype relationship (optional)

Above, we identified the SNP that is most strongly associated with tibia length.

- Here, let's look closely at the relationship between tibia length and the genotype at this SNP.

Load the GEMMA phenotype data:

```
y <- read.table("../data/pheno.txt")[[1]]
```

Visualize genotype-phenotype relationship (part 2)

Load the GEMMA genotype data, and select the genotypes for the top SNP. To obtain discrete genotype values (0, 1 or 2), round to the nearest integer.

```
library(data.table)
geno <- fread("../data/chr06.geno.txt",sep = " ",
              stringsAsFactors = FALSE)
class(geno) <- "data.frame"
geno        <- geno[-(1:3)]
geno        <- as.matrix(geno)
x           <- round(geno[i,-(1:3)])
```

Visualize genotype-phenotype relationship (part 3)

If we convert the genotype to a “factor” (categorical variable), we can use the boxplot function we used earlier:

```
x          <- factor(x,0:2)
levels(x) <- c("GG","AG","AA")
p4 <- draw.boxplot(x,y) +
  labs(x = "genotype",y = "tibia length (mm)")
print(p4)
```

Further, we can quickly quantify the relationship using the `lm` function in R:

```
summary(lm(y ~ x,data.frame(x = x,y = y)))
```

Visualizing the association signal: take-home points

- Today we implemented the basic steps of a genome-wide association analysis.
- We saw that most of the effort was getting the data ready for GEMMA, and carefully inspecting the GEMMA results.
- Did not discuss how to determine an appropriate significance threshold. This is a complicated question and beyond the scope of this workshop. **My advice:** first take an *exploratory approach* and identify the strongest association signals, then later assess significance.

Recap

Optional exercise: Repeat the same data processing and analysis steps to map genetic loci for LDL concentrations in blood (units = mmol/L); see the `"Bioch.LDL"` column in the phenotype table.