Expression quantitative trait loci mapping.

Introduction

Expression Quantitative Trait loci (eQTL) are genetic loci (single nucleotide polymorphisms, SNP) whose alleles are associated with different expression levels of a specific gene. Different alleles can be associated with a decrease or increase in gene expression. Figure 1 highlights how a SNP (in red) can be associated with gene expression.

In this example, an A allele increases gene expression in a dose-response manner, with A homozygotes displaying higher levels of expression of a gene compared to G homozygotes. Additionally, alleles are not expressed uniformly, with some variants rarer than others, as represented by the relative frequency of different genotypes on the y-axis.

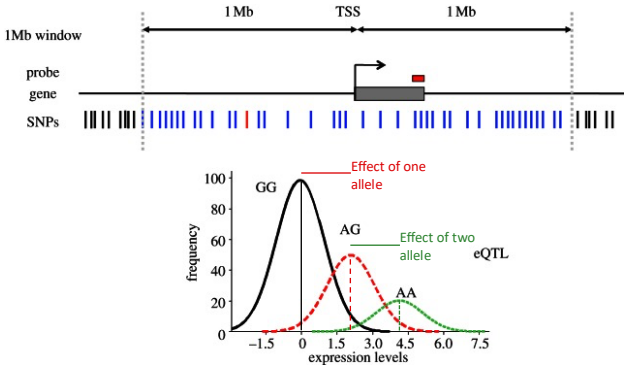

Figure 1. Representation of an eQTL effect on a gene. Figure taken from Nica, A. C. & Dermitzakis, E. T. Expression quantitative trait loci: present and future. Philos. Trans. R. Soc. London. Ser. B, Biol. Sci. ***368****, 20120362 (2013).*

Most eQTLs are found outside of coding regions[1] and can be divided into two categories:

- Cis-eQTLs are found on the same chromosome as the gene they influence, usually within a 100,000 base pair window around it. Cis eQTLs are thought to act on the gene directly through the regulation of enhancers, silencers, promoter regions or other regulatory elements of the gene.
- Trans-eQTLs can be found anywhere in the genome, further away from the gene they influence or even on other chromosomes. They are thought to influence gene expression through regulation of biological pathways.

During this practical, we will investigate how to identify eQTLs. To better understand eQTL analysis, we will start by simulating both genotype and expression data. This simulation approach will allow us to understand the structure of the data used for eQTL mapping as well as investigate information relevant to QTL mapping. After performing the simulation, we will investigate the GTEx website and see how eQTL can be used to investigate genome-wide association study (GWAS) results in real life.

Part 1: eQTL simulation

eQTL mapping.

During the lecture, we saw that QTL mapping can be performed using a simple linear regression of the following form:

$$y = \beta_0 + \beta_1 x \ + \varepsilon$$

With:

y = gene expression for the different individuals measured.
β_0 = intercept (mean effect).
β_1 = effect of an allele on the gene expression.
x = genotype value of the different individuals measured.
ε = error term of the model

To perform a QTL mapping, we will therefore have to simulate the gene expression y as well as the genotype x with the other parameters β_0 and $\beta_1$being inferred.

Genetic data:

Before simulating genetic data, we should first gain an understanding of how genotype data are represented.

> Question 1:
> Think about a single genetic locus where the allele can either be A or T.
> How would you represent four individuals whose genotypes at a specific locus are respectively:
>
> - A/A
> - A/T
> - T/A
> - T/T

Genetic expression is usually represented based on the number of alleles an individual carries at a specific locus. However, the allele used as a reference is *arbitrary*. For the previous example, we can represent the individuals based on the number of A alleles they carry: $[2\ 1\ 1\ 0]$ or based on the number of T: $[0\ 1\ 1\ 2]$.

To simulate genetic data, we, therefore, have to create several vectors containing 0, 1 or 2 representing the genotype of a single individual at different loci. Those vectors can then be arranged into a matrix of the following form:

$$\begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 2 \\ 1 & 2 & 1 & 0 \end{bmatrix}$$

With the columns representing different individuals and the rows a specific genetic locus.

Connection to the cluster:

To simulate the data, we will need to connect to the High-Performance Computing (HPC) cluster set up for this class. You should have been given the credentials necessary to connect to the cluster. Follow the information presented in the introductory guide to connect to the HPC cluster.
You can alternatively use your local machine. The simulations might, however, take a while.

Simulation of genotypes:

First, we will set up the HPC folders to keep your analyses organised. The following bash script allows you to create three folders:

```
cd ~
mkdir eQTLPrac
cd eQTLPrac
mkdir Genotype
mkdir Expression
mkdir eQTL
```

Files created will be stored in those folders.

Note: While the practical were designed to be run on the cluster, they can be run locally on your laptop.

The following R code can be used to simulate genetic data. Start your R session and, copy, and paste it within the R interpreter.

```
# Load all libraries needed for the practical:
library(tidyverse)
library(MASS)
library(cowplot)
library(MetBrewer)

set.seed(6543456)
frequency <- 0.5
SNP <- rbinom(5000, size = 2, frequency)
SNP_number <- 1000
indv_number <- 500
p <- runif(SNP_number, min = 0, max = 1)
genotypes <- replicate(indv_number, rbinom(SNP_number, 2, p))
rownames(genotypes) <- paste0('SNP', seq(1, nrow(genotypes)))
colnames(genotypes) <- paste0('Indv', seq(1, ncol(genotypes)))
print(nrow(genotypes))
print(ncol(genotypes))
print(genotypes[1:10,1:10])
```

Note: The *set.seed* function allows the code to be reproducible by fixing the random processes. A different seed would change the results.

Question 2:

- How many SNPs were simulated?
- How many individuals were simulated?
- Given that the SNP3 reference allele is G and the alternate allele C, what is the genotype of individual 5?

Exploration of allele frequency:

Now that we simulated genotype data, we can calculate the frequency of the alleles simulated with the following code:

```
maf = rowMeans(genotypes)/2
maf <- pmin(maf, 1-maf)

jpeg('~/eQTLPrac/Genotype/HistogramMAFsimulated.jpeg',width = 21, height = 12, res = 300, units = 'cm')
truehist(maf, main = "Histogram of minor allele frequency", col = "light grey", nbins=100)
lines(density(maf), lty = 2, col = "dark red", lwd = 3)
dev.off()
```

You can download the plot that you created by using the following command on your local machine:

scp <username>@203.101.xxx.xxx: ~/eQTLPrac/Genotype/HistogramMAFsimulated.jpeg .

Question 3:
Look at the allele frequency of the genotype data you simulated.

- What is the allele frequency occurring the most?
- Why is allele frequency important for eQTL analysis?
- What does the x-axis represent?
    - Why is it limited at 0.5?

Allele frequency is an important parameter during eQTL analysis due to the possible lack of representation of some genotypes. For QTL analyses to be possible, they need to ideally include individuals with all genotype groups or to have at least two genotype groups present (0,1 or 1,2).

Question 4:
Fill the table below using the genotype frequency derived from the Hardy-Weinberg principle for an allele A with a frequency p of 99%:

- What is the property of a linear regression that allows us to perform eQTL mapping when only 2 genotypes are present?
- Out of the three different populations in the table, which one(s) could be used to perform an eQTL mapping for alleles with a frequency of 1%?

Table 1. Proportion of the possible genotypes for a genetic loci with a minor allele frequency q of 0.01 (Calculation based on the Hardy–Weinberg principle)

	Population genotype frequency:	*1,000 individuals*	*10,000 individuals*	*100,000 individuals*
AA Frequency: p^2				
AT Frequency: $2pq$				
TT: Frequency: q^2				

To identify eQTL with a minor allele frequency of 1%, we would, therefore, need a population larger than 10,000 individuals to have access to two genotypes. With 198 heterozygous individuals expected in the population, we can perform eQTL mapping. Given that the eQTL effect is expected to be additive (the effect of two alleles is twice the effect of one allele), eQTL mapping can be performed with only 2 genotype groups available.

Currently large eQTL studies such as the eQTLgen consortium[2] contain 31,684 individuals, showing that eQTL mapping for rare variant (MAF < 1%) is currently not feasible.
Allele frequency needs to be kept in mind when performing eQTL mapping; exclusion of rare variants is often performed.

Simulation of gene expression data:

Now that we simulated genetic data, we need to create matching gene expression data.
While gene expression is not normally distributed (*RNA-sequencing and read-based sequencing technology generate discrete data that usually follow a negative binomial or Poisson distribution*), most analyses will start by normalising the data. Simulating gene expression data can be performed either at the discrete level or at the normalised level.

For simplicity, this practical will simulate data normally distributed data.

```
genesTotal <- 1000
geneswithQTL <- 50
geneswithoutQTL <- genesTotal - geneswithQTL
# Select the SNPs associated with each of the gene:
SNPs <- rownames(genotypes)
SNPswithQTL <- sample(SNPs, size = geneswithQTL)
SNPswithoutQTL <- SNPs[-which(SNPs %in% SNPswithQTL)]

expMatrixAssociated <- do.call(cbind, lapply(SNPswithQTL,
                                             function(i) {
                  #Simulate expression of three different cellTypes:
                   meanCT1 <- c(rnorm(mean = rnorm(mean = 3, n = 1), n = 1, ),
                                rnorm(mean = rnorm(mean = 5, n = 1), n = 1),
                                rnorm(mean = rnorm(mean = 7, n = 1), n = 1))
                   meanCT2 <- c(rnorm(mean = 3, n = 1),
                                rnorm(mean = 3, n = 1),
                                rnorm(mean = 3, n = 1))
                   meanCT3 <- c(rnorm(mean = rnorm(mean = 4, n = 1), n = 1),
                                rnorm(mean = rnorm(mean = 2, n = 1), n = 1),
                                rnorm(mean = rnorm(mean = 0, n = 1), n = 1))
      yWithQTLCT1 <- rnorm(indv_number, meanCT1[factor(genotypes[i,])])
      yWithQTLCT2 <- rnorm(indv_number, meanCT2[factor(genotypes[i,])])
      yWithQTLCT3 <- rnorm(indv_number, meanCT3[factor(genotypes[i,])])
      df <- data.frame(ct1 = yWithQTLCT1,
                       ct2 = yWithQTLCT2,
                       ct3 = yWithQTLCT3)
# Create a bulk dataframe as the sum of the expression of the three cell types:
      df$bulk <- rowSums(df, na.rm = T)
 return(df)
}))
# Add columns to the matrix:
colnames(expMatrixAssociated) <- paste0('Gene',
                                        rep(1:geneswithQTL, each = 4),
                                        '_',
                                        colnames(expMatrixAssociated))
# Simulate genes without QTLs:
expMatrixNotAssociated <- do.call(cbind, lapply(SNPswithoutQTL,
                       function(i) {
                        meanForAlleles  <- c(rnorm(1,10))
      yWithQTLCT1 <- rnorm(indv_number, meanForAlleles)
      yWithQTLCT2 <- rnorm(indv_number, meanForAlleles)
      yWithQTLCT3 <- rnorm(indv_number, meanForAlleles)
      df <- data.frame(ct1 = yWithQTLCT1,
                       ct2 = yWithQTLCT2,
                       ct3 = yWithQTLCT3)
      df$bulk <- rowSums(df, na.rm = T)
  return(df)
}))
# Add colunms to the matrix:
colnames(expMatrixNotAssociated) <- paste0('Gene',
                                           rep(1:geneswithoutQTL, each = 4),
                                           '_',
                                           colnames(expMatrixNotAssociated))
```

Question 5:
Run the code above and answer the following questions:

- How many genes were simulated?
    - How many of those genes were associated with SNPs?
- How many cell types were simulated?
- How was the bulk expression created?
    - What omics technology does the bulk data corresponds? Cell type data?

The following code will generate plots showing the association between SNPs and genotypes. You can change the code (by changing the name of the SNPs and gene in red) to visually inspect the association between different SNPs and genes.

```
expMatrix <- cbind(expMatrixNotAssociated %>% dplyr::select(contains('bulk')),
            expMatrixAssociated %>% dplyr::select(contains('bulk')))
colnames(expMatrix) <- paste0('Gene', 1:ncol(expMatrix))



### Test for SNPs:
SNPassociationPlot <- function(expMatrix, SNPID, GeneID) {
  ggplot(data.frame(snp = genotypes[SNPID,], y = expMatrix[,GeneID]),
         aes(x = factor(snp), y = y)) +
    ggtitle(paste0('Association between ', GeneID, ' and ', SNPID)) +
    geom_boxplot(fill = 'dark red') + geom_point(col = 'dark grey') +
    xlab("Reference allele count") +
    theme_minimal() +  theme(plot.title = element_text(hjust = 0.5))
}

p1 <- SNPassociationPlot(expMatrix, SNPID = 'SNP182', GeneID = 'Gene5')
p2 <- SNPassociationPlot(expMatrix, SNPID = 'SNP243', GeneID = 'Gene2')
p3 <- SNPassociationPlot(expMatrix, SNPID = 'SNP921', GeneID = 'Gene2')
p4 <- SNPassociationPlot(expMatrix, SNPID = 'SNP564', GeneID = 'Gene2')
p <- cowplot::plot_grid(p1,p2,p3,p4)
 ggsave(p, filename = '~/eQTLPrac/Expression/AssociationPlot.jpeg', width = 14, height=14, dpi =
300)
```

Download the plot created using the following command:

```
scp <username>@203.101.xxx.xxx:~/eQTLPrac/Expression/AssociationPlot.jpeg .
```

Question 6:
Based on the plot that you generated answer the following questions:

- What is the mean expression of the gene you simulated?
    - What is the unit of gene expression?
- Which SNP (if any) is associated with the expression of Gene 5 ?
    - How would you identify SNP statistically associated with gene expression?

While identifying SNP and gene expression pairs visually is already time-consuming, the human genome is composed of 3.2 billion base pairs and roughly 20,000 genes, rendering it impossible. We need to use the linear regression that we previously described and filter the results based on significance.

eQTL mapping, simple linear regression:

Now that we have simulated both gene expression and genotype, we will use linear regression to identify significant eQTL:

```
# Expression:
GeneID='Gene1'
# Set the first test:
Association <- summary(lm(expMatrix[,GeneID]~genotypes['SNP1',]))
Association <- as.data.frame(Association$coefficients)[2,]
rownames(Association) <- 'SNP1'

Association <- data.frame()
for(SNPID in rownames(genotypes)){
  test <- summary(lm(expMatrix[,GeneID]~genotypes[SNPID,]))
  test <- as.data.frame(test$coefficients)[2,]
  rownames(test) <- SNPID
  Association <- rbind(Association, test)
}
colnames(Association) <- c("Estimate", "Std.Error", "t_value", "P")
Association %>% arrange(P) %>% head() %>% print()
# Change the SNP in the following code:
signifQTL <- SNPassociationPlot(expMatrix, SNPID = 'ChangeSNP', GeneID = 'Gene1')
ggsave(signifQTL,
       filename = '~/eQTLPrac/Expression/SignificantQTL.jpeg', width = 7, height=7, dpi = 300)
```

Question 7:

- What SNP (if any) is significantly associated with Gene 1?
    - Fill the gap in the code with the significantly associated SNP and investigate visually the association.
- Modify the previous code to investigate associations with Gene 982.

The association test between the gene and 1000 SNPs for 500 individuals that we just performed took only a few seconds. However, this toy example does not represent the scale of the human genome with its more than 3 billion base pairs. eQTL analyses quickly result in prohibitive computation time as we increase the number of SNPs and individuals tested.

Software such as matrixeQTL[3] and fastQTL[4] have been developed to decrease the computational resources and time necessary for eQTL analyses. While we will not go into details on their inner working here, the underlying mechanisms of that software remain similar to the analysis performed within this practical. Methodology used to improve computational efficiency ranges from limiting the SNPs tested for a gene to the closest SNPs to developing mathematical approximations to computationally heavy calculations.

QTL mapping with interaction:

Our simulation of gene expression data was based on the presence of three different cell types measured. This simulation represents the working of bulk-RNA sequencing. We will now see what those QTLs look like when we decompose them across cell type.

Question 8:

- Modify and run the code below to visually inspect Gene 1 with the SNP that you previously found to be significantly associated.
    - Does the eQTL identified previously represent a specific cell type?

```
set.seed(58944)

expMatrixCT <- cbind(expMatrixAssociated %>% dplyr::select(!contains('bulk')),
    expMatrixNotAssociated %>% dplyr::select(!contains('bulk')))
colnames(expMatrixCT) <- paste0('Gene', rep(1:1000, each = 3), '_',
                        c('ct1', 'ct2', 'ct3'))
expMatrixCT <- expMatrixCT %>% mutate(Indv = paste0('Indv',
                                    seq(1, indv_number)))

expMatrixCTlonger <- expMatrixCT %>% pivot_longer(cols = -Indv,
                      names_to = 'geneID',
                      values_to = 'expression') %>%
  mutate(gene = str_split(geneID, '_', simplify = T)[,1],
       cellType = str_split(geneID, '_', simplify = T)[,2])

genotypeToMerge <- t(genotypes) %>% as.data.frame() %>%
  mutate(Indv = rownames(.))
expMatrixCTlonger <- left_join(expMatrixCTlonger, genotypeToMerge, by = 'Indv')
# Plot gene 1:
# Change the code in Red to inspect the gene of interest:
ggplot(expMatrixCTlonger %>% filter(gene == 'Gene1'),
      aes(x = factor(SNP1), y = expression, fill = cellType)) +
  geom_point(position = position_jitterdodge()) +
  facet_wrap(~cellType) +
  geom_boxplot() +
  theme_minimal() +
  scale_fill_manual(values  = met.brewer( n = 3, 'Hokusai1')) -> interactionPlot
ggsave(interactionPlot,
      filename = '~/eQTLPrac/Expression/interactionQTL.jpeg', width = 7, height=7, dpi = 300)
```

As we can see, the eQTL observed previously was produced by a different expression in gene between the different cell types. We do not observe any significant eQTL when the cell type information is known.

We will now investigate eQTL when different cell types are included.
Our previous linear regression can be written with interaction between cell type and genotype as follows:

$$y = \beta_0 + \beta_1 x_1 \ + \beta_2 x_2 + \beta_3 x_1 x_2$$

With:
y = *gene expression for the different individuals measured.*
β_0 = *intercept (mean effect).*
β_1 = *effect of an allele on the gene expression.*
x_1 = *genotype value of the different individuals measured.*
x_2 = cell type of origin
β_2 = overall effect of cell type on gene expression
$\boldsymbol{\beta_3}$ ***= Effect of cell type on the observed effect of genotype.***
ε = error term of the model

Using this model, our term of interest will be $\boldsymbol{\beta_3}$, representing the effect of one cell type compared to the overall effect of the genotype.

Question 9: Run the code below and modify the plotting function to output a significant interaction for gene 999

- What is the interaction observed between cell type, your significant SNP and gene 999?
- What does the bulk QTL data look like for your significant SNP and gene 999, is it a significant eQTL?

```
interactionResults <- data.frame()
for (snp in SNPs) {
  test <- expMatrixCTlonger %>% filter(gene == 'Gene985') %>%
    dplyr::select(expression, gene, cellType, snp) %>% mutate(variant = snp)
  colnames(test) <- c('expression', 'gene', 'cellType', 'genotype', 'variant')

  lmTest <- broom::tidy(summary(lm(expression ~ genotype +
                             cellType + genotype*cellType,
           data = test)))
  lmTest$SNP <- snp
  interactionResults <- rbind(interactionResults, lmTest)
}

interactionResults %>%
  filter(str_detect(term, ':')) %>%
  arrange(p.value) %>% head()
# Change the SNP value in Red to the most significant SNP:
ggplot(expMatrixCTlonger %>% filter(gene == 'Gene985'),
       aes(x = factor(SNP), y = expression, fill = cellType)) +
  geom_point(position = position_jitterdodge()) +
  facet_wrap(~cellType) +
  geom_boxplot() +
  theme_minimal() +
  scale_fill_manual(values  = met.brewer( n = 3, 'Hokusai1')) -> signifInteractionQLT

ggsave(signifInteractionQLT,
       filename = '~/eQTLPrac/Expression/SignificantQTL.jpeg', width = 7, height=7, dpi = 300)
```

eQTL associations can be driven by biological factors such as cell type proportion or environmental factors. Accurate mapping of eQTL using interaction allows for a better investigation of GWAS results or of causality between phenotypes (see SMR practicals).

Part 2: Real world eQTL:

Genotype-Tissue Expression (GTEx):

We will now investigate real-world eQTLs data. For this, we will go to the GTEx website. You can access it through this link (https://gtexportal.org/home/).
The GTEx consortium collected post-mortem samples for 948 donors. We know that eQTLs are dynamic and evolve over time and with exposure to the environment. Characteristics such as sex, age or disease status can influence eQTL association and are therefore important.

Question 10:
- On the GTEx website, look for the sample characteristics that could influence eQTL association study.
    - *Hint: Navigate to the Tissue & Sample statistics page*

eQTL are influenced by both age[5], sex[6] and ancestry[7]; the observed unbalanced number of males and females, as well as a largely white and aging (84.6% white, 68.1% of samples older than 50) cohort, therefore, need to be taken into account when performing eQTL analysis. Additionally, the cohort can be split in half with younger donors succumbing to traumatic injury while older donors displaying non-traumatic pathologies.

Sample characteristics, therefore, need to be considered when performing QTL mapping. You can read the landmark GTEx publication in 2020[8] to observe which sample characteristics were corrected for when testing for QTL associations.

Investigation of GWAS signal:

We will now investigate a real example of an eQTL association. For this, we will start by looking at a genome-wide association study of lipids published in 2013[9]:

Discovery and refinement of loci associated with lipid levels (https://www.nature.com/articles/ng.2797)

Question 11:
- Read the abstract of the GWAS paper, what is the goals of this paper?

This paper aimed to identify the genetic control of blood lipid levels. As such, they identified associations between SNP and blood lipid levels. They then mapped those SNPs to the closest genes, concluding on their role on blood lipid levels.

We will investigate how eQTL can give us more information regarding the genetic control of blood low-density lipoprotein (LDL) cholesterol.

Open the Supplementary figures from the paper and go to the supplementary table 3.

Question 12:

- Finds the gene with the strongest negative effect on LDL blood levels.
    - What is the impact of each alternate allele?
    - If the average person has an LDL blood level of 209.7mg/dL, what would be the expected LDL level of an individual with a genotype of GG at locus rs6511720?

Let's investigate the effect of rs6511720, the genetic loci associated with the highest decrease in LDL blood levels. Search the GTEx website for rs6511720 and answer the following question:

Question 13:

- With which genes is rs6511720 associated?
- In which tissues are those association located?
- Do you think that a change in gene expression is responsible for the association observed between LDL levels and rs6511720?

We will now look at genetic loci associated with LDL cholesterol levels. rs12916 is associated with HMGCR, a gene coding for HMG-CoA reductase an enzyme playing a central role in cholesterol synthesis. Let's investigate eQTL associated with rs12916, search the GTEx website for rs12916.

Question 14:

- In which tissue is rs12916 associated with HMGCR?
- Where does the SNP fall? (*hint: open the IGV browser*)
- Do you think that a change in gene expression is responsible for the association observed between LDL levels and rs12916?

In conclusion, eQTL can help interpreting the functional significance of GWAS signals. They can provide biological interpretation of non-coding variants helping to hint at the mechanisms underlying complex traits and diseases.

Extension question:

The code used to generate bulk RNA-seq at the start of this practical assumes an equal mixture of all three cell types. This does not represent the biology of most organs or tissues and was used as a simplified illustration of the interaction between biological factors and genotype.

- Modify the original code generating bulk data to include reads coming from different proportions of cell types.
- How does this influence the identification of QTL at the bulk or single-cell level?

References:

1. Liu, H. *et al.* Epigenome-augmented eQTL-hotspots reveal genome-wide transcriptional programs in 36 human tissues. *Brief. Bioinform.* **25**, bbae109 (2024).
2. Võsa, U. *et al.* Large-scale cis- and trans-eQTL analyses identify thousands of genetic loci and polygenic scores that regulate blood gene expression. *Nat. Genet.* **53**, 1300–1310 (2021).
3. Shabalin, A. A. Matrix eQTL: ultra fast eQTL analysis via large matrix operations. *Bioinformatics* **28**, 1353–1358 (2012).
4. Ongen, H., Buil, A., Brown, A. A., Dermitzakis, E. T. & Delaneau, O. Fast and efficient QTL mapper for thousands of molecular phenotypes. *Bioinformatics* **32**, 1479–1485 (2016).
5. Yamamoto, R. *et al.* Tissue-specific impacts of aging and genetics on gene expression patterns in humans. *Nat. Commun.* **13**, 5803 (2022).
6. Oliva, M. *et al.* The impact of sex on gene expression across human tissues. *Science* **369**, (2020).
7. Zeng, B. *et al.* Multi-ancestry eQTL meta-analysis of human brain identifies candidate causal variants for brain-related traits. *Nat. Genet.* **54**, 161–169 (2022).
8. The GTEx Consortium atlas of genetic regulatory effects across human tissues. *Science* **369**, 1318–1330 (2020).
9. Willer, C. J. *et al.* Discovery and refinement of loci associated with lipid levels. *Nat. Genet.* **45**, 1274–1283 (2013).