Statistics
in Medicine

Received 28 February 2015, Accepted 6 July 2015 Published online 6 September 2015 in Wiley Online Library

(wileyonlinelibrary.com) DOI: 10.1002/sim.6605

A guide to genome-wide association analysis and post-analytic interrogation

Eric Reed,[a] Sara Nunez,[a] David Kulp,[b] Jing Qian,[c]
Muredach P. Reilly[d] and Andrea S. Foulkes[a*†]

This tutorial is a learning resource that outlines the basic process and provides specific software tools for implementing a complete genome-wide association analysis. Approaches to post-analytic visualization and interrogation of potentially novel findings are also presented. Applications are illustrated using the free and open-source R statistical computing and graphics software environment, Bioconductor software for bioinformatics and the UCSC Genome Browser. Complete genome-wide association data on 1401 individuals across 861,473 typed single nucleotide polymorphisms from the PennCATH study of coronary artery disease are used for illustration. All data and code, as well as additional instructional resources, are publicly available through the Open Resources in Statistical Genomics project: http://www.stat-gen.org**. © 2015 The Authors. *Statistics in Medicine* Published by John Wiley & Sons Ltd.**

Keywords: statistical genomics; tutorial; genome-wide association (GWA) study; R code; Bioconductor; SNP filtering; sample filtering; call rate; minor allele frequency (MAF); Hardy–Weinberg equilibrium (HWE); heterozygosity; relatedness; IBD; ancestry; substructure; principal component analysis (PCA); parallel processing; imputation; Manhattan plot; Q–Q plot; lambda statistic; heatmap; regional association plot; UCSC Genome Browser

1. Introduction

This brief tutorial is intended as a learning and teaching tool, offering the fundamental computational skills for entry into the field of applied statistical genetics and bioinformatics. Unique from existing resources, this tutorial offers the framework as well as complete and extensible R scripts to perform a comprehensive genome-wide association (GWA) analysis and post-analytic interrogation. Familiarity at an elementary textbook level [1–3] with basic statistical and genetics concepts for GWA studies is assumed. The content of this tutorial builds and expands on several existing and highly recommended resources on genetic and statistical concepts, including [4, 5]. We begin the analysis after genotyping calls are made and quality control and assurance measures are taken, as described, for example, in [6, 7]. An alternative freely available software platform is PLINK, another toolset used for whole genome association analysis. Additional post-analytic interrogation is also presented in this manuscript using the UCSC Genome Browser. A companion website is available for this tutorial through the Open Resources in Statistical Genomics (ORSG) project (http://www.stat-gen.org) with all R coding examples fully embedded in .Rmd files to be edited and weaved as dynamic and reproducible reports. A complete list of external resources is provided in Supplementary Information A.

[a] *Department of Mathematics and Statistics, Mount Holyoke College, South Hadley, MA, U.S.A.*
[b] *Department of Computer Science, University of Massachusetts, Amherst, MA, U.S.A.*
[c] *Department of Biostatistics and Epidemiology, University of Massachusetts, Amherst, MA, U.S.A.*
[d] *Department of Medicine, University of Pennsylvania, Philadelphia, PA, U.S.A.*
**Correspondence to: Andrea S. Foulkes, Department of Mathematics and Statistics, Mount Holyoke College, South Hadley, MA, U.S.A.*
[†] *E-mail: afoulkes@mtholyoke.edu.*

Support for this research is provided by NIH/NHLBI R01-HL107196.

The focus of this tutorial is on GWA analysis of common variants that involves testing association of each single nucleotide polymorphism (SNP) independently and subsequently characterizing findings through a variety of visual and analytic tools. In the rare variant setting, in which interest resides in investigating variations that are present in less than 1% of the population, alternative techniques are needed that account for regional associations. The reader is referred to a rich literature that addresses rare variant analysis, including [8–10]. In the present manuscript, we focus on the analysis of data arising from population-based GWA studies of unrelated individuals where primary interest resides in identifying associations between SNPs and a single binary, for example, case or control status or quantitative phenotype. Extensive methods and tools specific to family-based investigations that account for within-family correlation structures are also available (e.g., [11, 12]). Further extensions of the tools presented to censored survival or longitudinal outcomes can be achieved through application of an alternative modeling framework in the association analysis of step 7. The data used for illustration here are limited to the 22 autosomal chromosomes, and both typed and 1000 Genomes [13] imputed SNPs are considered as potential predictor variables. Post-analytic interrogation of SNP-level findings is an essential part of GWA analysis, and first steps, including mapping positive SNP findings to gene regions, are described herein. We note that there exists a large literature on alternative analytical paradigms for simultaneous analysis

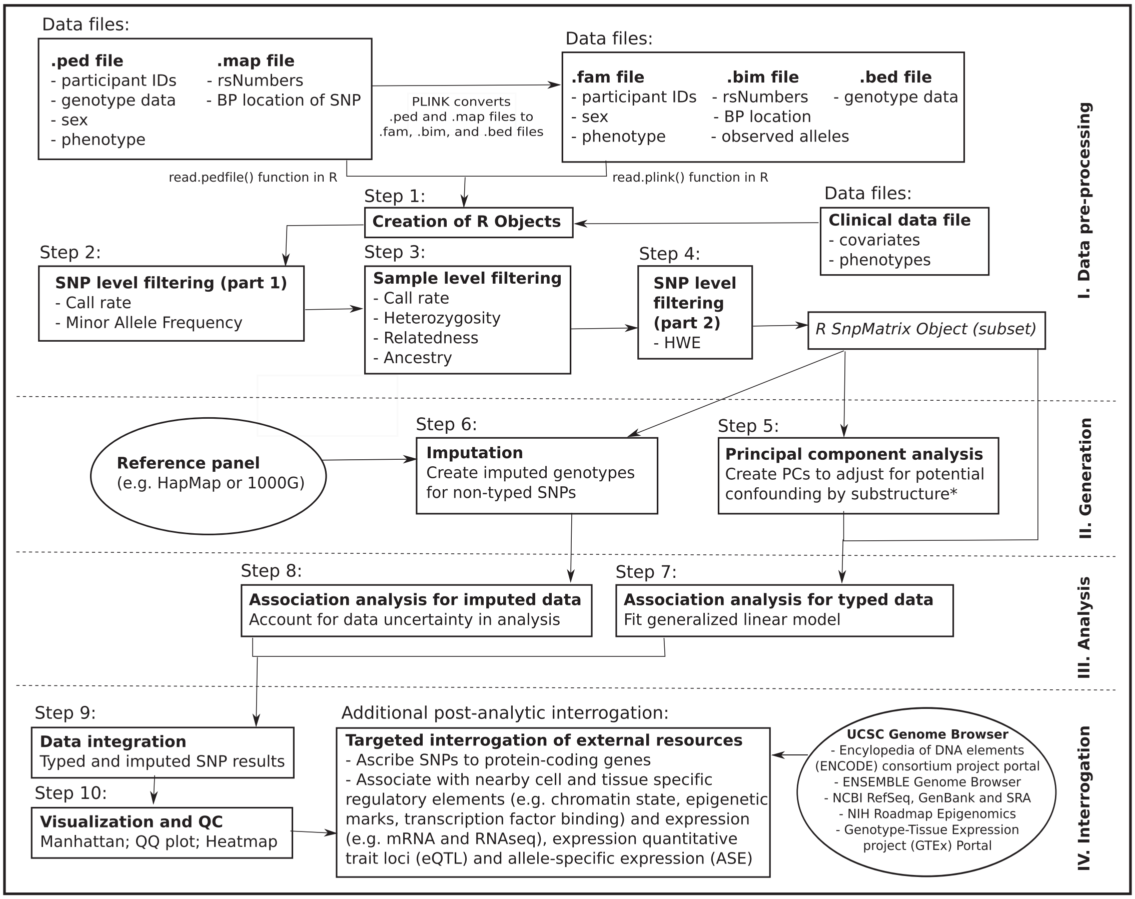
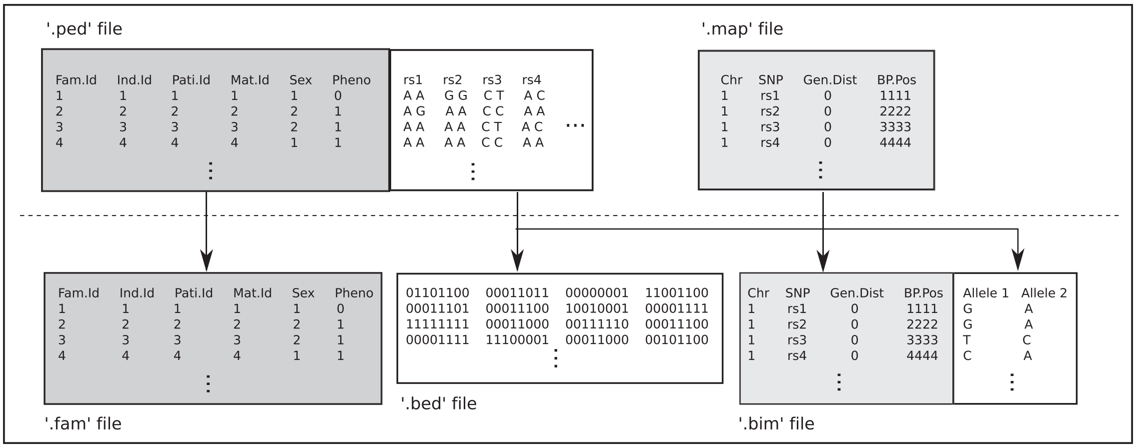

Data files:
.ped file
- participant IDs
- genotype data
- sex
- phenotype
.map file
- rsNumbers
- BP location of SNP
PLINK converts .ped and .map files to .fam, .bim, and .bed files
Data files:
.fam file
- participant IDs
- sex
- phenotype
.bim file
- rsNumbers
- BP location
- observed alleles
.bed file
- genotype data
read.pedfile() function in R
read.plink() function in R
Step 1:
Creation of R Objects
Data files:
Clinical data file
- covariates
- phenotypes
Step 2:
SNP level filtering (part 1)
- Call rate
- Minor Allele Frequency
Step 3:
Sample level filtering
- Call rate
- Heterozygosity
- Relatedness
- Ancestry
Step 4:
SNP level filtering (part 2)
- HWE
R SnpMatrix Object (subset)
I. Data pre-processing
Reference panel
(e.g. HapMap or 1000G)
Step 6:
Imputation
Create imputed genotypes for non-typed SNPs
Step 5:
Principal component analysis
Create PCs to adjust for potential confounding by substructure*
II. Generation
Step 8:
Association analysis for imputed data
Account for data uncertainty in analysis
Step 7:
Association analysis for typed data
Fit generalized linear model
III. Analysis
Step 9:
Data integration
Typed and imputed SNP results
Step 10:
Visualization and QC
Manhattan; QQ plot; Heatmap
Additional post-analytic interrogation:
Targeted interrogation of external resources
- Ascribe SNPs to protein-coding genes
- Associate with nearby cell and tissue specific regulatory elements (e.g. chromatin state, epigenetic marks, transcription factor binding) and expression (e.g. mRNA and RNAseq), expression quantitative trait loci (eQTL) and allele-specific expression (ASE)
UCSC Genome Browser
- Encylopedia of DNA elements (ENCODE) consortium project portal
- ENSEMBLE Genome Browser
- NCBI RefSeq, GenBank and SRA
- NIH Roadmap Epigenomics
- Genotype-Tissue Expression project (GTEx) Portal
IV. Interrogation

Figure 1. Genome-wide association (GWA) analysis workflow. GWA analysis is composed of 10 essential steps that fall into four broadly defined categories as illustrated in this figure. Additional detail on the structure of the data files, particularly the relationship of the .ped and .map files with the .bim, .bed, and .fam files, is provided in Figure 2. This workflow is based on a single GWA analysis and may be modified in the context of a large collaborative meta-analysis involving the combination of multiple GWA studies that require harmonization. Additional detail on typical modifications in this context is provided in Section 6. *Substructure, also referred to as population admixture and population stratification, refers to the presence of genetic diversity (e.g., different allele frequencies) within an apparently homogenous population that is due to population genetic history (e.g., migration, selection, and/or ethnic integration).

of multiple SNPs, including methods for gene-based (e.g., [14–16]) and pathway-based (e.g., [17, 18]) analysis, as well as growing literature on gene–environment interaction analysis in the context of GWA studies [19].

The PennCATH cohort data, arising from a GWA study of coronary artery disease (CAD) and cardiovascular risk factors based at University of Pennsylvania Medical Center [20], are used throughout this tutorial as an illustrative example and have been made publicly available for training use to accompany the tutorial. In this study, a total of $n = 3850$ individuals were recruited between July 1998 and March 2003. A nested case-control study of European ancestry severe angiographic CAD cases and angiographic normal controls were selected for genome-wide genotyping. De-identified data used in this tutorial are composed of $n = 1401$ individuals with genotype information across 861,473 SNPs. Corresponding clinical data, including age, sex, high-density lipoprotein (HDL)-cholesterol, low-density lipoprotein cholesterol, triglycerides, and CAD status are available as well. HDL-cholesterol, low-density lipoprotein cholesterol and triglycerides are all quantitative traits that are well-described cardiovascular disease risk factors. Notably, PennCATH is one of the core GWA studies nested within the Coronary ARtery DIsease Genome-wide Replication And Meta-analysis (CARDIoGRAM) consortium meta-data and serves as a representative regional population with no admixture [20, 21].

Genome-wide association analysis strategies typically include four broadly defined components: (i) data pre-processing; (ii) new data generation; (iii) statistical analysis; and (iv) post-analytic interrogation. A primary goal of these investigations is identifying and characterizing the association among SNPs and measures of disease progression or disease outcomes. In Sections 2 – 5 in the succeeding paragraphs, we present the key aspects of each of the core analytic components, including a description of attributes of the data, application of relevant software tools, and guidance on interpretation of findings. An overall summary of the analytic approach we follow is provided in Figure 1. Notably, this figure highlights multiple stages within each of the four broadly defined components of analysis. The resultant ten steps are as follows: (1) reading data into R to create an R object; (2) SNP-level filtering (part 1); (3) sample-level filtering; (4) SNP-level filtering (part 2); (5) principal component analysis (PCA); (6) imputation of non-typed genotypes; (7) association analysis of typed SNPs; (8) association analysis of imputed data; (9) integration of imputed and typed SNP results; and (10) visualization and quality control of association findings. Further data interrogation using external resources is also discussed. In the following sections, we elaborate on each of these steps. Notably, this workflow is typical for analysis of a single GWA study and may be modified in the context of a large collaborative meta-analysis involving the combination of multiple studies requiring harmonization. Additional detail on the analysis pipeline in this context is provided in Section 6 where we also present a broader contemporary context and additional available resources.



'.ped' file

Fam.Id	Ind.Id	Pati.Id	Mat.Id	Sex	Pheno	rs1	rs2	rs3	rs4	
1	1	1	1	1	0	A A	G G	C T	A C	
2	2	2	2	2	1	A G	A A	C C	A A	
3	3	3	3	2	1	A A	A A	C T	A C	···
4	4	4	4	1	1	A A	A A	C C	A A	
		⋮					⋮			

'.map' file

Chr	SNP	Gen.Dist	BP.Pos
1	rs1	0	1111
1	rs2	0	2222
1	rs3	0	3333
1	rs4	0	4444
	⋮		

Fam.Id	Ind.Id	Pati.Id	Mat.Id	Sex	Pheno
1	1	1	1	1	0
2	2	2	2	2	1
3	3	3	3	2	1
4	4	4	4	1	1
		⋮			

'.fam' file

```
01101100   00011011   00000001   11001100
00011101   00011100   10010001   00001111
11111111   00011000   00111110   00011100
00001111   11100001   00011000   00101100
                    ⋮
```

'.bed' file

Chr	SNP	Gen.Dist	BP.Pos	Allele 1	Allele 2
1	rs1	0	1111	G	A
1	rs2	0	2222	G	A
1	rs3	0	3333	T	C
1	rs4	0	4444	C	A
	⋮			⋮	

'.bim' file

Figure 2. Genome-wide association data files. GWA data files are typically organized into either .ped and .map files or .bim, .bed, and .fam files. Plink converts .ped and .map files into .bim, .bed, and .fam files. The later set is substantially smaller because the .bed file contains a binary version of the genotype data. R can read in either set of files although the later is preferable.

2. Data pre-processing

In the example we present, samples were genotyped using the Affymetrix 6.0 GeneChip and provided to us in .CEL format. The Birdseed calling algorithm, which is based on an expectation-maximization type algorithm [20], was applied to generate genotypes and confidence scores for each sample at every SNP. In turn, PERL and unix scripts were used to convert these to .ped and .map files. While R can read .ped and .map files, it is generally preferable to first convert these two files to .bim, .bed, and .fam files. This can be carried out using PLINK and is preferable as the conversion of the .ped file to a .bed file, a binary formatted file, results in a substantial reduction in file size. In the following texts, we describe the elements of each file type mentioned and their interrelatedness. A visual representation of the data files is provided in Figure 2.

- .ped and .map files: The .ped file contains information on each study participant including family ID, participant ID, father ID, mother ID, sex, phenotype, and the full typed genotype. Here, each SNP is bi-allelic (i.e., only two nucleotides are observed at any given SNP across study participants) and coded as a pair of nucleotides (A, C, T, or G). Notably, the ordering in the pair is non-informative in the sense that the first alleles listed for each of the two SNPs are not necessarily on the same chromosome. The .map file contains a row for each SNP with rsNumber (SNP) and corresponding chromosome (chr) and coordinate (BPPos) based on the current genome build.
- .bim, .bed, and .fam files: The .bim file contains the same information as the .map file as well as the two observed alleles at each SNP (A1 and A2) from the .ped file. It contains a row for each SNP and six columns, containing information for the chromosome number, rsNumber, genetic distance, position identifier, allele 1, and allele 2. The .bed file contains a binary version of the genotype data. This is the largest of the three files because it contains every SNP in the study, as well as the genotype at this SNP for each individual. The .fam file contains the participant identification information, including a row for each individual and six columns, corresponding the same columns described for the .ped file with the exception of the genotype data. Note that not all of these columns contain unique information. That is, in a population-based study of unrelated individuals, ‘family ID number’ and ‘individual ID number’ will be the same.
- Clinical data file: An additional ascii .txt or .csv file is typically available, which includes clinical data on each study subject. The rows of this file represent each subject, and the columns correspond to available covariates and phenotypes. There may be redundancies in this file and the data contained in the columns labeled ‘sex’ and ‘phenotype’ in the .fam file.

We begin by installing packages and setting up global parameters in R. This tutorial utilizes several packages available from Bioconductor, an open-source bioinformatic software repository. Of these, we make the most use of `snpStats`, which includes functions to read in various formats of genotype data and carry out quality control, imputation, and association analysis. SNPRelate is also well utilized and includes functions for sample-level quality control and computationally efficient principal component (PC) calculation. Other packages include functionalities for data visualization (`ggplot2`, `LDheatmap`, `postgwas`), data manipulation (`plyr`), and parallel processing (`doParallel`), as well as their dependencies.

```
# ---- packages ----
# Run this once interactively to download and install Bioconductor packages and other packages.

source("http://bioconductor.org/biocLite.R")
biocLite("snpStats")
biocLite("SNPRelate")
biocLite("rtracklayer")
biocLite("biomaRt")
install.packages(c('plyr', 'GenABEL', 'LDheatmap','doParallel', 'ggplot2', 'coin', 'igraph', 'devtools'))

library(devtools)
install_url("http://cran.r-project.org/src/contrib/Archive/postgwas/postgwas_1.11.tar.gz")
```

We additionally specify the parameters used in the data processing and analysis. Of particular note, we set the location of the GWA data set (available at https://www.mtholyoke.edu/courses/afoulkes/Data/GWAStutorial/) and specify input and output files.

```
# ---- globals ----
# Customize as needed for file locations

# Modify data.dir to indicate the location of the GWAStutorial files
# Intermediate data files will also be stored in this same location unless you set out.dir
data.dir <- '/Volumes/genome/Research/GWAS'
out.dir <- data.dir                     # may want to write to a separate dir to avoid clutter

# Input files
gwas.fn <- lapply(c(bed='bed',bim='bim',fam='fam',gds='gds'), function(n) sprintf("%s/GWAStutorial.%s", data.dir, n))
clinical.fn <- sprintf("%s/GWAStutorial_clinical.csv", data.dir)
onethou.fn = lapply(c(info='info',ped='ped'), function(n) sprintf("%s/chr16_1000g_CEU.%s", data.dir, n))
protein.coding.coords.fname <- sprintf("%s/ProCodgene_coords.csv", out.dir)

# Output files
gwaa.fname <- sprintf("%s/GWAStutorialout.txt", out.dir)
gwaa.unadj.fname <- sprintf("%s/GWAStutorialoutUnadj.txt", out.dir)
impute.out.fname <- sprintf("%s/GWAStutorial_imputationOut.csv", out.dir)
CETP.fname <- sprintf("%s/CETP_GWASout.csv", out.dir)
```

2.1. *Reading and formatting data in R (step 1)*

In order to read the .fam, .bim, and .bed files in R, we use the `read.plink()` function in the Bioconductor `snpStats` package. The `genotype` slot of the resulting list contains an SnpMatrix object, which is a matrix of genotype data with a column for each SNP and a row for each study participant.

```
# ---- step1 ----
# Reading data into R

source("globals.R")

# ---- step1-a ----
library(snpStats)

# Read in PLINK files to create list
geno <- read.plink(gwas.fn$bed, gwas.fn$bim, gwas.fn$fam, na.strings = ("-9"))

# ---- step1-b ----
# Obtain the genotype SnpMatrix object from generated list
# Note: Phenotypes and covariates will be read from the clinical data file, below
genotype <- geno$genotype
print(genotype)                  # 861473 SNPs read in for 1401 subjects

# Obtain the SNP information table from the list
genoBim <- geno$map
colnames(genoBim) <- c("chr", "SNP", "gen.dist", "position", "A1", "A2")
print(head(genoBim))

# Remove raw file to open up memory
rm(geno)
```

The clinical data (*GWAStutorial_clinical.csv*) can be read in the familiar way as a comma delimited text file.

```
# ---- step1-c ----
# Read in clinical file
clinical <- read.csv(clinical.fn,
                     colClasses=c("character", "factor", "factor", rep("numeric", 4)))
rownames(clinical) <- clinical$FamID
print(head(clinical))
```

Finally, we subset the data at this stage to include only individuals who have data available in both the genotype and phenotype files.

```
# ---- step1-d ----
# Subset genotype for individuals with clinical data
genotype <- genotype[clinical$FamID, ]
print(genotype)  # All 1401 subjects contain both clinical and genotype data
```

In the data example provided, genotype information is available for 861,473 typed SNPs across $n = 1401$ individuals with available phenotype data.

As illustrated in Figure 1, once we have read in the the genotype and clinical information, we are ready to proceed with the next steps of the GWA data pre-processing. This involves two stages of filtering the data, at SNP and sample levels, respectively. Each of these is described in more detail in the succeeding texts, accompanied by the appropriate R code for implementation. We note again that the order of analysis may vary depending on whether a single GWA analysis is being performed (as described herein) or the analyst is preparing results to be incorporated into a larger meta-analysis that requires data harmonization across multiple studies. In the latter case, the following filtering steps (steps 2, 3, and 4) may be excluded or performed centrally after analysis (steps 7 and 8) as summary level data are combined across studies.

2.2. Single nucleotide polymorphism-level filtering – part 1 (step 2)

The second data pre-processing step involve removing (also referred to as 'filtering') SNPs that will not be included analysis. Well-described reasons for this exclusion include large amounts of missing data, low variability, and genotyping errors (e.g., [22]). Typically, SNP-level filtering based on a large amount of missing data and lower variability is performed first. This is followed by sample-level filtering (see step 3 in the succeeding texts), and finally, SNP-level filtering based on possible genotyping errors (see step 4 in the succeeding texts) is performed. The rationale for this is that both sample-level relatedness and substructure (for which we filter in step 3) can influence the Hardy–Weinberg equilibrium (HWE) criterion (step 4) used for filtering SNPs based on genotyping errors. An iterative procedure that repeats the SNP and sample-level filtering until no additional samples are removed is also common. In our setting, however, no samples are filtered, deeming this loop unnecessary.

- SNP-level filtering: call rate. The call rate for a given SNP is defined as the proportion of individuals in the study for which the corresponding SNP information is not missing. In the following example, we filter using a call rate of 95%, meaning we retain SNPs for which there is less than 5% missing data. More stringent cut points (e.g., less than 5%) may be employed in smaller sample settings.
- SNP-level filtering: minor allele frequency (MAF). A large degree of homogeneity at a given SNP across study participants generally results in inadequate power to infer a statistically significant relationship between the SNP and the trait under study. This can occur when we have a very small MAF so that the large majority of individuals have two copies of the major allele. Here, we remove SNPs for which the MAF is less than 1%. In some instances, particularly small sample settings, a cut point of 5% is applied.

We filter simultaneously on call rate and MAF using the following script.

```
# ---- step2 ----
# SNP-level filtering (part 1)

# ---- step2-a ----
# Create SNP summary statistics (MAF, call rate, etc.)
snpsum.col <- col.summary(genotype)
print(head(snpsum.col))

# ---- step2-b ----
# Setting thresholds
call <- 0.95
minor <- 0.01

# Filter on MAF and call rate
use <- with(snpsum.col, (!is.na(MAF) & MAF > minor) & Call.rate >= call)
use[is.na(use)] <- FALSE                # Remove NA's as well

cat(ncol(genotype)-sum(use),"SNPs will be removed due to low MAF or call rate.\n") #203287 SNPs will be removed

# Subset genotype and SNP summary data for SNPs that pass call rate and MAF criteria
genotype <- genotype[,use]
snpsum.col <- snpsum.col[use,]

print(genotype)                           # 658186 SNPs remain
```

In the data example provided, we filter 203,287 SNPs based on call rate < 0.95 and/or MAF < 0.01.

2.3. Sample-level filtering (step 3)

The third stage of data pre-processing involves filtering samples, that is, removing individuals who we select to be excluded from analysis. Criteria for sample-level filtering are generally based on missing data, sample contamination, correlation (for population-based investigations), and racial, ethnic, or gender ambiguity or discordance. Each of these is described later. Additional detail on sample-level filtering is available in, for example, [22].

- Sample-level filtering: call rate. Similar to SNP-level filtering based on call rate, we exclude individuals who are missing genotype data across more than a pre-defined percentage of the typed SNPs. This proportion of missingness across SNPs is referred to as the sample call rate, and we apply a threshold of 95%. That is, individuals who are missing genotype data for more than 5% of the typed SNPs are removed. A new reduced dimension `SnpMatrix` genotype object is created, which incorporates this filter.
- Sample-level filtering: heterozygosity. Heterozygosity refers to the presence of each of the two alleles at a given SNP within an individual. This is expected under HWE to occur with probability $2 * p * (1 - p)$, where p is the dominant allele frequency at that SNP (assuming a bi-allelic SNP). Excess heterozygosity across typed SNPs within an individual may be an indication of poor sample quality, while deficient heterozygosity can indicate inbreeding or other substructure in that person [23]. Thus, samples with an inbreeding coefficient $|F| = (1 - O/E) > 0.10$ are removed, where O and E are respectively the observed and expected counts of heterozygous SNPs within an individual. Note that we calculate the expected counts for each individual based on the observed SNPs for that individual.

We filter on sample call rate and heterozygosity simultaneously in the following script:

```
# ---- step3 ----
# sample-level filtering

# ---- step3-a ----
library(SNPRelate)                      # Estimating LD, relatedness, PCA
library(plyr)

# Create sample statistics (Call rate, Heterozygosity)
snpsum.row <- row.summary(genotype)

# Add the F stat (inbreeding coefficient) to snpsum.row
MAF <- snpsum.col$MAF
callmatrix <- !is.na(genotype)
hetExp <- callmatrix %*% (2*MAF*(1-MAF))
hetObs <- with(snpsum.row, Heterozygosity*(ncol(genotype))*Call.rate)
snpsum.row$hetF <- 1-(hetObs/hetExp)

head(snpsum.row)

# ---- step3-b ----
# Setting thresholds
sampcall <- 0.95    # Sample call rate cut-off
hetcutoff <- 0.1    # Inbreeding coefficient cut-off

sampleuse <- with(snpsum.row, !is.na(Call.rate) & Call.rate > sampcall & abs(hetF) <= hetcutoff)
sampleuse[is.na(sampleuse)] <- FALSE    # remove NA's as well
cat(nrow(genotype)-sum(sampleuse), "subjects will be removed due to low sample call rate or inbreeding coefficient.\n")
        # 0 subjects removed

# Subset genotype and clinical data for subjects who pass call rate and heterozygosity crtieria
genotype <- genotype[sampleuse,]
clinical<- clinical[ rownames(genotype), ]
```

Because the PennCATH data provided are pre-filtered, no additional samples are filtered based on an inbreeding coefficient $|F| > 0.10$.

- Sample-level filtering: cryptic relatedness, duplicates, and gender identity. Population-based cohort studies are often limited to unrelated individuals, and the generalized linear modeling approach described in step 7 (association analysis of typed SNPs) later assumes independence across individuals. Further discussion of alternative data structures and associated analysis tools is provided in Section 6. Importantly, in regional cohort studies (e.g., hospital-based cohort studies) of complex diseases, individuals from the same family can be recruited unintentionally. A common measure of relatedness (or duplication) between pairs of samples is based on identity by descent (IBD). An IBD kinship coefficient of greater than 0.10 may suggest relatedness, duplicates, or sample mixture. Typically, the individual of a related pair with lower genotype call rate is removed. We note that gender identity can also be checked at this stage to confirm that self-reported gender is consistent with the observed X and Y chromosomes; however, in the data example provided, sex chromosomes are not available, and thus, an example of filtering on gender identity is not provided.

  We begin by applying linkage disequilibrium (LD) pruning using a threshold value of 0.2, which eliminates a large degree of redundancy in the data and reduces the influence of chromosomal artifacts [6]. This dimension reduction step is commonly applied prior to both IBD analysis and PCA, applied in the succeeding texts for ancestry filtering, and results in large computational savings.

```
# ----step3-c ----
# Checking for Relatedness

ld.thresh <- 0.2    # LD cut-off
kin.thresh <- 0.1   # Kinship cut-off

# Create gds file, required for SNPRelate functions
snpgdsBED2GDS(gwas.fn$bed, gwas.fn$fam, gwas.fn$bim, gwas.fn$gds)
genofile <- openfn.gds(gwas.fn$gds, readonly = FALSE)

# Automatically added "-1" sample suffixes are removed
gds.ids <- read.gdsn(index.gdsn(genofile, "sample.id"))
gds.ids <- sub("-1", "", gds.ids)
add.gdsn(genofile, "sample.id", gds.ids, replace = TRUE)

# Prune SNPs for IBD analysis
set.seed(1000)
```

```
geno.sample.ids <- rownames(genotype)
snpSUB <- snpgdsLDpruning(genofile, ld.threshold = ld.thresh,
                          sample.id = geno.sample.ids, # Only analyze the filtered samples
                          snp.id = colnames(genotype)) # Only analyze the filtered SNPs
snpset.ibd <- unlist(snpSUB, use.names=FALSE)
cat(length(snpset.ibd),"will be used in IBD analysis\n")  # expect 72812 SNPs
```

This reduces the number of SNPs from 658,186 at the end of step 2 to 72,812. Next, we calculate pairwise IBD distances to search for sample relatedness. A strategy is employed that iteratively removes subjects with the highest number of pairwise kinship coefficients > 0.1.

```
# ----step3-d ----
# Find IBD coefficients using Method of Moments procedure.  Include pairwise kinship.
ibd <- snpgdsIBDMoM(genofile, kinship=TRUE,
                    sample.id = geno.sample.ids,
                    snp.id = snpset.ibd,
                    num.thread = 1)
ibdcoeff <- snpgdsIBDSelection(ibd)     # Pairwise sample comparison
head(ibdcoeff)

# ---- step3-e ----
# Check if there are any candidates for relatedness
ibdcoeff <- ibdcoeff[ ibdcoeff$kinship >= kin.thresh, ]

# iteratively remove samples with high kinship starting with the sample with the most pairings
related.samples <- NULL
while ( nrow(ibdcoeff) > 0 ) {

    # count the number of occurrences of each and take the top one
    sample.counts <- arrange(count(c(ibdcoeff$ID1, ibdcoeff$ID2)), -freq)
    rm.sample <- sample.counts[1, 'x']
    cat("Removing sample", as.character(rm.sample), 'too closely related to', sample.counts[1, 'freq'],'other samples.\n')

    # remove from ibdcoeff and add to list
    ibdcoeff <- ibdcoeff[ibdcoeff$ID1 != rm.sample & ibdcoeff$ID2 != rm.sample,]
    related.samples <- c(as.character(rm.sample), related.samples)
}

# filter genotype and clinical to include only unrelated samples
genotype <- genotype[ !(rownames(genotype) %in% related.samples), ]
clinical <- clinical[ !(clinical$FamID %in% related.samples), ]

geno.sample.ids <- rownames(genotype)

cat(length(related.samples), "similar samples removed due to correlation coefficient >=", kin.thresh,"\n")
print(genotype)                         # expect all 1401 subjects remain
```

In our example, none of the samples are filtered based on the IBD kinship coefficient >0.10.

- Sample-level filtering: ancestry. PCA is one approach to visualizing and classifying individuals into ancestry groups based on their observed genetic makeup. We do this for two reasons: First, self-reported race and ethnicity can differ from clusters of individuals that are based solely on genetic information, and second, the presence of an individual not appearing to fall within a racial/ethnic cluster may be suggestive of a sample-level error. Note that we use the subset of 72,812 SNPs after LD pruning (step 3-c) as the input for the PCA. An alternative strategy to first-stage LD pruning, which also improves computational efficiency, is the 'HapMap rooted' analysis, which involves first performing PCA in a reference panel, for example, HapMap or 1000 Genomes, and then projecting the study sample onto the resulting space. This approach is not presented herein but can be implemented with existing functionalities of the Kinship-based INference for Gwas (KING) software [24].

```
# ---- step3-f ----
# Checking for ancestry

# Find PCA matrix
pca <- snpgdsPCA(genofile, sample.id = geno.sample.ids,  snp.id = snpset.ibd, num.thread=1)

# Create data frame of first two principal components
pctab <- data.frame(sample.id = pca$sample.id,
                    PC1 = pca$eigenvect[,1],    # the first eigenvector
                    PC2 = pca$eigenvect[,2],    # the second eigenvector
                    stringsAsFactors = FALSE)

# Plot the first two principal components
plot(pctab$PC2, pctab$PC1, xlab="Principal Component 2", ylab="Principal Component 1", main = "Ancestry Plot")
```

No additional samples are filtered based on visual inspection of PCA plots. Again, we expect this as the PennCATH data provided are pre-filtered.

2.4. *Single nucleotide polymorphism-level filtering – part 2 (step 4)*

- SNP-level filtering: HWE. Violations of HWE can be an indication of the presence of population substructure or the occurrence of a genotyping error. While they are not always distinguishable, it is a common practice to assume a genotyping error and remove SNPs for which HWE is violated. If case-control status is available, we limit this filtering to analysis of controls as a violation in cases may be an indication of association. Departures from HWE are generally measured at a given SNP using a χ^2 goodness-of-fit test between the observed and expected genotypes. We remove SNPs for which the HWE test statistic has a corresponding p-value of less than 1×10^{-6} in controls.

```
# ---- step4 ----
# Hardy-Weinberg SNP filtering on CAD controls

hardy <- 10^-6      # HWE cut-off

CADcontrols <- clinical[ clinical$CAD==0, 'FamID' ]
snpsum.colCont <- col.summary( genotype[CADcontrols,] )
HWEuse <- with(snpsum.colCont, !is.na(z.HWE) & ( abs(z.HWE) < abs( qnorm(hardy/2) ) ) )
rm(snpsum.colCont)

HWEuse[is.na(HWEuse)] <- FALSE          # Remove NA's as well
cat(ncol(genotype)-sum(HWEuse),"SNPs will be removed due to high HWE.\n")  # 1296 SNPs removed

# Subset genotype and SNP summary data for SNPs that pass HWE criteria
genotype <- genotype[,HWEuse]

print(genotype)                           # 656890 SNPs remain
```

We filter out an additional 1,296 SNPs based on HWE $p < 1 \times 10^{-6}$ in CAD controls. This results in 656,890 typed SNPs to be considered in the association analysis.

3. New data generation

After completion of SNP and sample-level filtering, we generate two new types of data prior to performing our statistical analysis. The first are PCs that are intended to capture information of latent population substructure that is typically not available in self-reported race and ethnicity variables. The second are genotypes of untyped SNPs that may have a functional relationship to the outcome and therefore provide additional power for identifying association. Each of these is described in more detail in the succeeding texts.

3.1. *Creating principal components for capturing population-substructure (step 5)*

Substructure, also referred to as population admixture and population stratification, refers to the presence of genetic diversity (e.g., different allele frequencies) within an apparently homogenous population that is due to population genetic history (e.g., migration, selection, and/or ethnic integration). PCs based on observed genotype data capture information on substructure and are straightforward to generate using the `snpgdsPCA()` function in the `SNPRelate` package based on the full genotype data. Notably, we again apply LD pruning prior to the PCA. Typically, the first 10 PCs are considered as possible confounders. This number is routine, but arbitrary, and one alternative is to select the number of PCs based on a pre-defined proportion of variability that they explain. The λ-statistic is typically used to evaluate whether inclusion of PCs is necessary. This is described further in step 10 (quantile–quantile (Q–Q) plots and the λ-statistic).

```
# ---- step5 ----
# Generating principal components for modeling

# ---- step5-a ----
# Set LD threshold to 0.2
ld.thresh <- 0.2

set.seed(1000)
geno.sample.ids <- rownames(genotype)
snpSUB <- snpgdsLDpruning(genofile, ld.threshold = ld.thresh,
                          sample.id = geno.sample.ids, # Only analyze the filtered samples
                          snp.id = colnames(genotype)) # Only analyze the filtered SNPs
snpset.pca <- unlist(snpSUB, use.names=FALSE)
cat(length(snpset.pca),"\n")  # 72578 SNPs will be used in PCA analysis

pca <- snpgdsPCA(genofile, sample.id = geno.sample.ids,  snp.id = snpset.pca, num.thread=1)

# Find and record first 10 principal components
# pcs will be a N:10 matrix.  Each column is a principal component.
pcs <- data.frame(FamID = pca$sample.id, pca$eigenvect[,1 : 10],
                  stringsAsFactors = FALSE)
colnames(pcs)[2:11]<-paste("pc", 1:10, sep = "")

print(head(pcs))
```

3.2. *Imputing non-typed single nucleotide polymorphisms using 1000 Genomes data (step 6)*

Typed SNPs measured using chip array technology typically capture approximately one-million polymorphisms, which vary in at least 1% of the general population. More generally, interest lies in analyzing association of genotypes of non-typed SNPs with disease outcomes because functional (causal) variants may not be measured. Using the extensive externally derived resources on reference haplotypes and their LD structure, such as HapMap and 1000 Genomes data, we can impute the unmeasured genotype data. Three well-described and recommended stand-alone packages for SNP-level imputation are IMPUTE2, MACH, and BEAGLE. Imputed genotypes can be reported as the 'best guess' genotype or as the posterior probability of each genotype at a given location on the genome. Importantly, the uncertainty in this estimation process needs to be accounted for in the association analysis, and thus, we distinguish between genotyped and imputed data henceforth. Methods that specifically account for the uncertainty in the imputed SNP data are described in step 8 later.

After imputation, a quality control step is performed to filter imputed data with high degrees of uncertainty. Common measures of uncertainty are the information content and R^2 [25]. We apply an R^2 threshold of 0.7 for inclusion in association analysis where in this case, R^2 is the value association with the linear model regressing each imputed SNP on regional typed SNPs. This is described further in the `snpStats` package documentation. Additionally, we exclude SNPs at this stage with a MAF, after assignment of the highest posterior probability genotype, of less than 0.01. For the purpose of illustration, we use the `snp.imputation()` and `impute.snps()` functions in the R package `snpStats` to impute a limited set of 1000 Genome SNPs on the same chromosome (chromosome 16) as the genotyped SNPs identified as genome-wide significant in the GWA association analysis (step 7). In practice, imputation is often performed across all chromosomes, resulting in up to 12.5 million typed and imputed SNPs on which association analysis can be performed [13].

```
# ---- step6 ----
# Genotype imputation

# ----  step6-a ----
# Read in 1000g data for given chromosome 16
thougeno <- read.pedfile(onethou.fn$ped, snps = onethou.fn$info, which=1)

# Obtain genotype data for given chromosome
genoMatrix <- thougeno$genotypes

# Obtain the chromosome position for each SNP
support <- thougeno$map
colnames(support)<-c("SNP", "position", "A1", "A2")
head(support)

# Imputation of non-typed 1000g SNPs
```

```
presSnps <- colnames(genotype)

# Subset for SNPs on given chromosome
presSnps <- colnames(genotype)
presDatChr <- genoBim[genoBim$SNP %in% presSnps & genoBim$chr==16, ]
targetSnps <- presDatChr$SNP

# Subset 1000g data for our SNPs
# "missing" and "present" are snpMatrix objects needed for imputation rules
is.present <- colnames(genoMatrix) %in% targetSnps

missing <- genoMatrix[,!is.present]
print(missing)                          # Almost 400,000 SNPs

present <- genoMatrix[,is.present]
print(present)                          # Our typed SNPs

# Obtain positions of SNPs to be used for imputation rules
pos.pres <- support$position[is.present]
pos.miss <- support$position[!is.present]

# Calculate and store imputation rules using snp.imputation()
rules <- snp.imputation(present, missing, pos.pres, pos.miss)
```

We then remove failed imputations, imputed SNPs with high degrees of associated uncertainty, and imputed SNPs with low estimated MAF.

```
# ----  step6-b ----
# Remove failed imputations
rules <- rules[can.impute(rules)]
cat("Imputation rules for", length(rules), "SNPs were estimated\n")  # Imputation rules for 197888 SNPs were estimated

# Quality control for imputation certainty and MAF
# Set thresholds
r2threshold <- 0.7
minor <- 0.01

# Filter on imputation certainty and MAF
rules <- rules[imputation.r2(rules) >= r2threshold]

cat(length(rules),"imputation rules remain after uncertain imputations were removed\n")
        # 162565 imputation rules remain after uncertain impuations were removed

rules <- rules[imputation.maf(rules) >= minor]
cat(length(rules),"imputation rules remain after MAF filtering\n")  # 162565 imputation rules remain after MAF filtering

# Obtain posterior expectation of genotypes of imputed snps
target <- genotype[,targetSnps]
imputed <- impute.snps(rules, target, as.numeric=FALSE)
print(imputed)  # 162565 SNPs were imputed
```

This analysis results in 162,565 1000 Genomes imputed SNPs on chromosome 16 that are carried forward in step 8 for association analysis. We again emphasize that the uncertainty in imputation needs to be considered in the context of association analysis, and thus, these SNPs are considered separately from the typed SNPs analyzed in step 7.

4. Genome-wide association analysis

4.1. Association analysis of typed single nucleotide polymorphisms (step 7)

Association analysis typically involves regressing each SNP separately on a given trait, adjusted for patient-level clinical, demographic, and environmental factors. The assumed underlying genetic model of association for each SNP (e.g., dominant, recessive, or additive) will impact the resulting findings; however, because of the large number of SNPs and the generally uncharacterized relationships to the outcome, a single additive model is typically selected. In this case and as illustrated in the code provided, each SNP is represented as the corresponding number of minor alleles (0, 1, or 2). Notably, coding SNP variables based on alternative models (e.g., dominant or recessive) is straightforward, and the association analysis described proceeds identically [26, 27]. In practice, a Bonferonni-corrected genome-wide significance threshold of 5×10^{-8} is used for control of the family-wise error rate. This cutoff is based on research, suggesting approximately one-million independent SNPs across the genome (e.g., [28]), so tends be applied regardless of the actual number of typed or imputed SNPs under investigation.

In our data example, we use inverse normally transformed HDL-cholesterol as the response, adjusting for age, sex, and the first 10 PCs. HDL-cholesterol is a complex trait associated with cardiovascular disease, for which age and sex are established risk factors. These two covariates and the arbitrary choice of 10 PCs are routine for cardiovascular disease trait association studies (e.g., [20, 26, 27]). Importantly, as in any model fitting procedure, it is essential to evaluate the appropriateness of model assumption and specifically the normality of the trait under study. Visual inspection of a histogram of HDL-cholesterol (code provided but plot not shown) reveals some extreme values, and therefore, an inverse normal transformation is selected. Alternative transformations, such as the log-transformation, may also be reasonable and have the advantage of maintaining the relative distance between observations. We do not emphasize this in the present tutorial as standard statistical modeling practice can be applied. The following code prepares the phenotype data for analysis.

```
# ---- step7 ----
# Association analysis of typed SNPs

library(GenABEL)

# ----step7-a ----
# Merge clincal data and principal components to create phenotype table
phenoSub <- merge(clinical,pcs)      # data.frame => [ FamID CAD sex age hdl pc1 pc2 ... pc10 ]

# We will do a rank-based inverse normal transformation of hdl
phenoSub$phenotype <- rntransform(phenoSub$hdl, family="gaussian")

# Show that the assumptions of normality met after transformation
par(mfrow=c(1,2))
hist(phenoSub$hdl, main="Histogram of HDL", xlab="HDL")
hist(phenoSub$phenotype, main="Histogram of Tranformed HDL", xlab="Transformed HDL")

# Remove unnecessary columns from table
phenoSub$hdl <- NULL
phenoSub$ldl <- NULL
phenoSub$tg <- NULL
phenoSub$CAD <- NULL

# Rename columns to match names necessary for GWAS() function
phenoSub <- rename(phenoSub, replace=c(FamID="id"))

# Include only subjects with hdl data
phenoSub<-phenoSub[!is.na(phenoSub$phenotype),]
# 1309 subjects included with phenotype data

print(head(phenoSub))
```

Running a GWA analysis in parallel, that is, simultaneously across several cores, is recommended because of the large number of models that require fitting. Each core runs the GWA analysis for a subset of the SNPs, and when computation is completed across all cores, the output is returned to its original order. In the succeeding texts, we describe a cross-platform approach to running the analysis in parallel on MAC, Unix, and Windows operating systems. The `detectCores()` function can be used to determine the available number of workers. To run the analysis in parallel, we use the `dopar()` function in the `doParallel` package, indicating the number of workers. The output of `doPar()` is an ascii text file. These are contained in the `GWAA()` function that we developed, and is available in Supplementary Information B of this manuscript.

```
# ---- step7-b ----
# Run GWAS analysis (using parallel processing)

library(plyr)
source("GWAA.R")

# Note: This function writes a file, but does not produce an R object
start <- Sys.time()
GWAA(genodata=genotype, phenodata=phenoSub, filename=gwaa.fname)
end <- Sys.time()
print(end-start)
```

In our setting, two genotyped SNPs in the cholesteryl ester transfer protein (CETP) gene region, rs1532625 and rs247617, are suggestive of association ($p < 5 \times 10^{-6}$) with respective p-values of 8.92×10^{-8} and 1.25×10^{-7}. CETP is a well-characterized gene that has been associated previously with HDL-C (e.g., [26]). More information on these SNPs and the process of post-analytic interrogation is provided in steps 9 and 10 later.

4.2. *Association analysis of imputed data (step 8)*

Several stand-alone packages can be applied to conduct association analysis of imputed SNPs using the corresponding posterior probabilities. These include, for example, MACH2qtl/dat [29], ProbABEL [30], BEAGLE [31], BIMBAM [32], and SNPTEST [25]. Reviews and comprehensive comparisons of these approaches can be found in [33, 34]. The R package `snpStats` also has functions to read in imputed data based on which imputation package was used (e.g., BEAGLE, IMPUTE, and MACH). For illustrations, we use the `single.rhs.tests()` function in R package `snpStats` using the imputation rules generated in step 6.

```
# ---- step8 ----
# Association analysis of imputed SNPs

# ---- step8-a ----
# Carry out association testing for imputed SNPs using snp.rhs.tests()
rownames(phenoSub) <- phenoSub$id

imp <- snp.rhs.tests(phenotype ~ sex + age + pc1 + pc2 + pc3 + pc4 + pc5 + pc6 + pc7 + pc8 + pc9 + pc10,
                     family = "Gaussian", data = phenoSub, snp.data = target, rules = rules)

# Obtain p values for imputed SNPs by calling methods on the returned GlmTests object.
results <- data.frame(SNP = imp@snp.names, p.value = p.value(imp), stringsAsFactors = FALSE)
results <- results[!is.na(results$p.value),]

# Write a file containing the results
write.csv(results, impute.out.fname, row.names=FALSE)

# Merge imputation testing results with support to obtain coordinates
imputeOut<-merge(results, support[, c("SNP", "position")])
imputeOut$chr <- 16

imputeOut$type <- "imputed"

# Find the -log_10 of the p-values
imputeOut$Neg_logP <- -log10(imputeOut$p.value)

# Order by p-value
imputeOut <- arrange(imputeOut, p.value)
print(head(imputeOut))
```

In total, we identify 22 imputed SNPs on chromosome 16 that are significant at a suggestive association threshold of 5×10^{-6}. Next, we select only those SNPs within the region of CETP (±5 Kb) to report. Here, we use the `map2gene()` function we developed, which is also available in Supplementary Information B, that identifies the set of SNPs that belong to a specified gene region. This function uses gene coordinates based on Genome Reference Consortium GRCh37 (hg19), provided in the file *ProCodgene_coords.csv*. Further interrogation of these SNPs and the CETP region is provided in Figures 5a and 6 as well as associated text.

```
# ---- step8-b ----
source("map2gene.R")

# Read in file containing protein coding genes coords
genes <- read.csv(protein.coding.coords.fname, stringsAsFactors = FALSE)

# Subset for CETP SNPs
impCETP <- map2gene("CETP", coords = genes, SNPs = imputeOut)

# Filter only the imputed CETP SNP genotypes
impCETPgeno <- imputed[, impCETP$SNP ]
```

At this stage, we map 70 imputed SNPs to the CETP region, of which 16 are significant at the suggestive association threshold of 5×10^{-6}.

5. Post-analytic visualization and genomic interrogation

5.1. *Data integration (step 9)*

At this stage, it is also common to ascribe SNPs to loci and report chromosome and base pair locations, also referred to as coordinates or positions. Notably, the SNP coordinate is dependent on the genome build, and in our data example, we use the Genome Reference Consortium GRCh37 (hg19) build. A typical presentation of results includes gene and locus name; SNP name; chromosome number; base pair location, according to a specified build; the coefficient estimate (or odds ratio) from the model fitting procedure; the corresponding standard error; and the associated *p*-value.

```
# ---- step9 ----
# Data Integration

# ---- step9-a ----
# Read in GWAS output that was produced by GWAA function
GWASout <- read.table(gwaa.fname, header=TRUE, colClasses=c("character", rep("numeric",4)))

# Find the -log_10 of the p-values
GWASout$Neg_logP <- -log10(GWASout$p.value)

# Merge output with genoBim by SNP name to add position and chromosome number
GWASout <- merge(GWASout, genoBim[,c("SNP", "chr", "position")])

# Order SNPs by significance
GWASout <- arrange(GWASout, -Neg_logP)
print(head(GWASout))
```

An additional step is required to combine the imputed data results and the typed SNP results. Notably, genotype imputation can involve imputing all SNPs, including both unobserved and typed SNPs. Thus, the analyst may chose to select from the imputation results only SNPs that were not typed or did not pass the SNP-level filtering. In our example (step 6), we imputed non-typed SNPs as well as SNPs that did not pass SNP-level filtering (steps 2 and 4). GWA significant SNPs in this combined set can then be further visualized and interrogated as described in step 10.

```
# ---- step9-b ----
# Combine typed and imputed
GWASout$type <- "typed"

GWAScomb<-rbind.fill(GWASout, imputeOut)
head(GWAScomb)
tail(GWAScomb)

# Subset for CETP SNPs
typCETP <- map2gene("CETP", coords = genes, SNPs = GWASout)

# Combine CETP SNPs from imputed and typed analysis
CETP <- rbind.fill(typCETP, impCETP)[,c("SNP","p.value","Neg_logP","chr","position","type","gene")]
print(CETP)
```

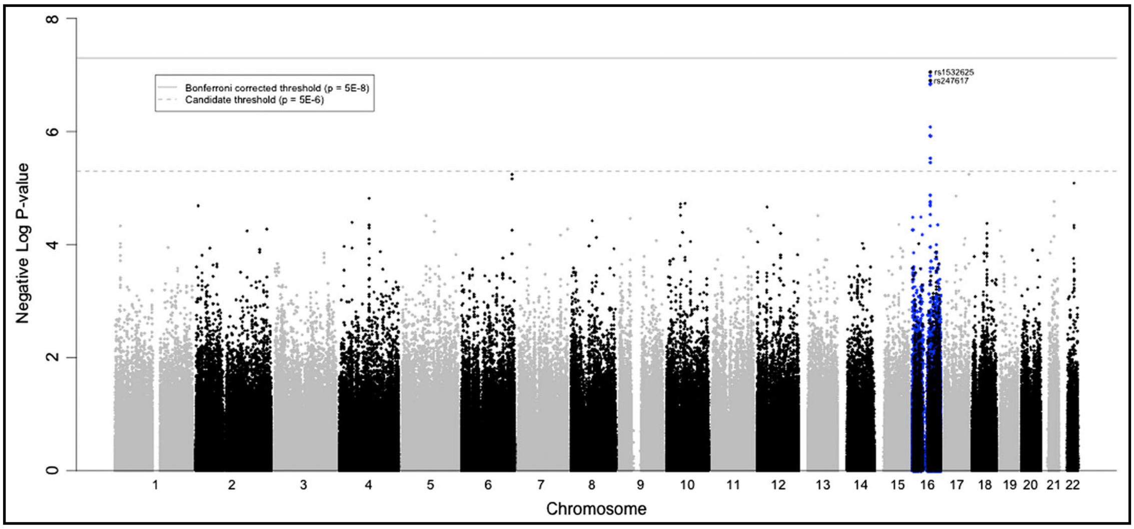


Figure 3. Manhattan plot of genome-wide association analysis results. This figure illustrates the level of statistical significance (*y*-axis), as measured by the negative log of the corresponding *p*-value, for each single nucleotide polymorphism (SNP). Each typed SNP is indicated by a grey or black dot. SNPs are arranged by chromosomal location (*x*-axis). Imputation was performed on chromosome 16 only using 1000 Genomes data, and imputed SNPs are indicated by blue dots. None of the SNPs reached the Bonferroni level of significance ($p < 5 \times 10^{-8}$ – solid horizontal line); however, two typed SNPs and 22 imputed SNPs (on chromosome 16) were suggestive of association ($p < 5 \times 10^{-6}$ – dashed horizontal line).

5.2. Visualization and Quality Control (step 10)

Several plots allow us both to visualize the GWA analysis findings and to perform quality control checks at the same time. Specifically, as elaborated in each section below we are interested in identifying data inconsistencies, potential systemic biases, and consistency of our findings with previously reported results. We describe three visualization tools in the succeeding texts. In addition to these visualization approaches, association analysis using other genetic models can be a useful sensitivity analysis.

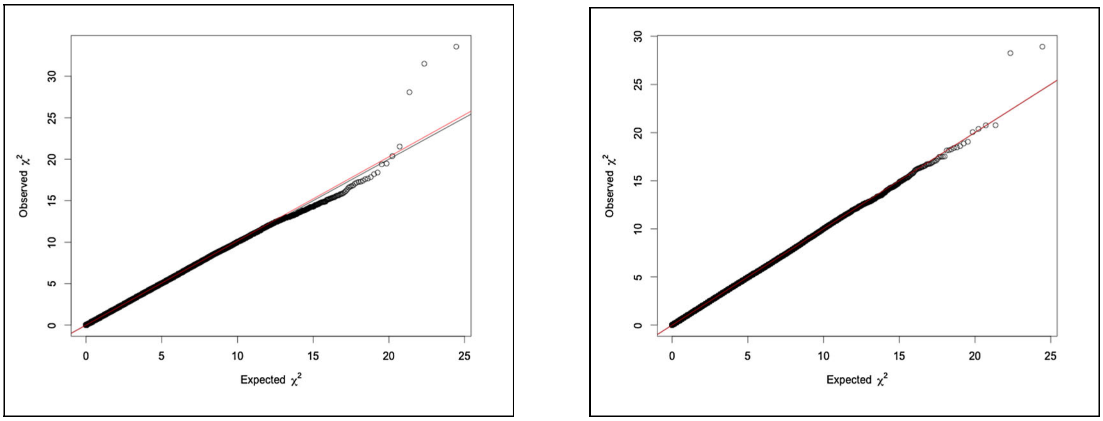
- Manhattan plots. Manhattan plots are used to visualize GWA significance level by chromosome location as shown in Figure 3. Here, each dot corresponds to a single SNP. The x-axis represents gene coordinates, and the numbers shown correspond to chromosome numbers. The y-axis is the negative of the log p-value, so that large values correspond to small p-values. The solid horizontal line indicates the Bonferonni corrected significance threshold ($-\log(5 \times 10^{-8})$). The dotted horizontal line is a less stringent suggestive association threshold ($-\log(5 \times 10^{-6})$) that we use as an indicator of a suggestive association and requiring further validation, similar to the approach taken in [26]. Visual inspection of this plot allows for identification of SNPs with relatively small p-values that are in regions with relatively large and non-significant p-values, suggesting potentially spurious findings. Multiple signals in the CETP region suggest that this may be a true signal. This plot is generated using the `GWAS_Manhattan()` that we developed and is available in Supplementary Information B.

```
# ---- step10 ----
# Visualizing and QC of GWA findings

source("GWAS_ManhattanFunction.R")
par(mfrow=c(1,1))

# ---- step10-a ----
# Create Manhattan Plot
GWAS_Manhattan(GWAScomb)
```

- Q–Q plots and the λ-statistic. Q–Q plots are used to visualize the relationship between the expected and observed distributions of SNP-level test statistics, as illustrated in Figure 4a and b. Figure 4a uses observed test statistics based on a model that does not adjust for potential confounders, while Figure 4b is based on a model that includes the first 10 PCs we generated in step 5 discussed earlier as well as sex and age. We see here that the tail of the distribution is brought closer to the $y = x$ line after accounting for potential confounding by population substructure in the modeling framework. If



(a) Unadjusted model; $\lambda = 1.0142$ **(b)** Adj for PCs, age and sex; $\lambda = 1.0032$

Figure 4. Quantile–quantile plots for quality control check and visualizing crude association. Quantile–quantile plots illustrate the relationship between observed (y-axis) and expected (x-axis) test statistics and are used as a tool for visualizing appropriate control of population substructure and the presence of association. The left panel (a) is based on an unadjusted model, where the deviation is below expected, while the right panel (b) is based on a model adjusted for potential confounders, which brings the tail closer to the $y = x$ line. The extreme observed statistics are suggestive of association. Data generally falling on the $y = x$ lines suggests no clear systemic bias. Unstandardized λ's are reported. PCs, principal components.

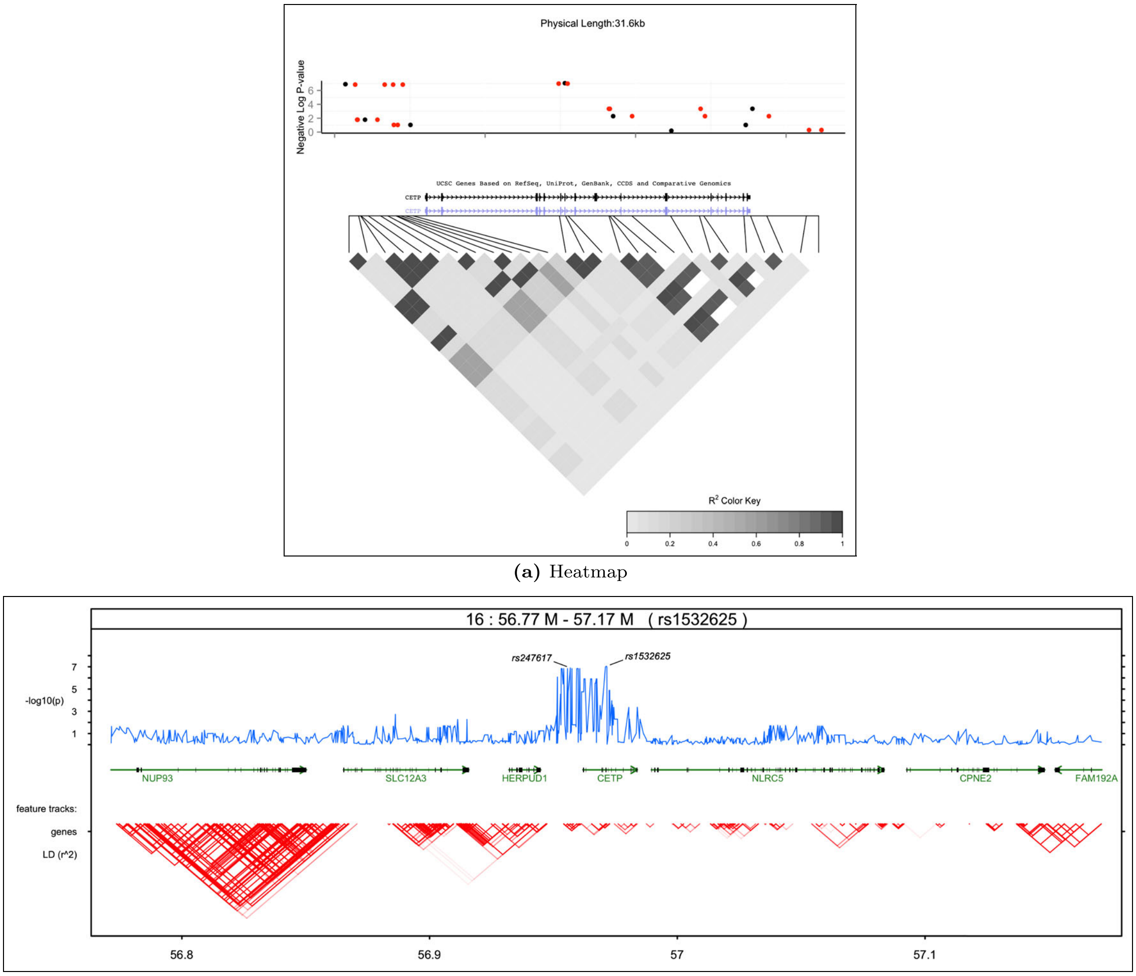
the data in this figure were shifted up or down from the $y = x$ line, then we would consider possible systemic bias. Finally, a slight deviation in the upper right tail from the $y = x$ line suggests crudely that some form of association is present in the data.

```
# ---- step10-b ----
# Rerun the GWAS using unadjusted model
phenoSub2 <- phenoSub[,c("id","phenotype")] # remove all extra factors, leave only phenotype

GWAA(genodata=genotype, phenodata=phenoSub2, filename=gwaa.unadj.fname)
GWASoutUnadj <- read.table(gwaa.unadj.fname, header=TRUE, colClasses=c("character", rep("numeric",4)))

# Create QQ plots for adjusted and unadjusted model outputs
par(mfrow=c(1,2))
lambdaAdj <- estlambda(GWASout$t.value^2,plot=TRUE,method="median")
lambdaUnadj <- estlambda(GWASoutUnadj$t.value^2,plot=TRUE,method="median")
cat(sprintf("Unadjusted lambda: %s\nAdjusted lambda: %s\n", lambdaUnadj$estimate, lambdaAdj$estimate))
```

The degree of deviation from this line is measured formally by the λ-statistic [35, 36], where a value close to 1 suggests appropriate adjustment for potential substructure. While λ is improved after adjusting for PCs (from $\lambda = 1.014$ to $\lambda = 1.0032$), a dramatic difference in values is not observed as



(a) Heatmap

(b) Regional association plot

Figure 5. Heatmap and regional association plots. Heatmap (top) illustrating linkage disequilibrium (LD) between typed (black) and imputed (red) single nucleotide polymorphisms (SNPs) in the cholesteryl ester transfer protein (CETP) region. A total of two typed SNPs and 16 imputed SNPs are significant at the less stringent 5×10^{-6} threshold; however, the heat map only illustrates imputed SNPs with a posterior probability of 1 for the associated genotype. We observe the presence of two distinct LD blocks within the CETP gene region, with high levels of LD between SNPs within each block and lower LD between SNPs across the the two blocks. A related regional association plot (bottom) illustrates association levels and LD for a larger window surrounding CETP.

this PennCATH sample is from a relatively homogenous population. In general, the goal is to achieve a value of λ close to one; $\lambda > 1.2$ suggests stratification, and typically, additional PCs are included in this setting, and in some cases, the study is eliminated from inclusion in subsequent meta-analysis. Calculation of a standardized λ that accounts for sample size is particularly useful in the context of contrasting values across studies for inclusion in meta-analysis. We apply the following code to generate standardized λ's for the unadjusted and adjusted models, resulting in $\lambda = 1.0108$ and 1.000632 for the unadjusted and adjusted models, respectively. For binary traits, the standardization approach described in [37] can be applied.

```
# ---- step10-c ----
# Calculate standardized lambda
lambdaAdj_1000<-1+(lambdaAdj$estimate-1)/nrow(phenoSub)*1000
lambdaUnadj_1000<-1+(lambdaUnadj$estimate-1)/nrow(phenoSub)*1000
cat(sprintf("Standardized unadjusted lambda: %s\nStandardized adjusted lambda: %s\n", lambdaUnadj_1000, lambdaAdj_1000))
```

- Heatmaps and regional association plots. Heatmaps and regional association plots are typically used in the context of GWA analysis to visualize LD patterns between GWA significant SNPs and SNPs in nearby regions, which have been previously reported. LD is typically measured by r^2 or D', which are both transformations of the scalar D, defined as the difference between the joint probability of the two major alleles and the product of the two marginal probabilities, where the adjustment is based on allele frequencies. A more detailed description of these quantities can be found in [3]. An example of a heatmap from our data analysis is provided in Figure 5a where we include the most significant SNP from our analysis (rs1532625) and SNPs in a nearby gene, CETP. The degree of shading indicates the amount of LD so that darker squares indicates higher LD. This figure is generated using the following code snippet.

```
# ---- step10-d ----
library(LDheatmap)
library(rtracklayer)

# Add "rs247617" to CETP
CETP <- rbind.fill(GWASout[GWASout$SNP == "rs247617",], CETP)

# Combine genotypes and imputed genotypes for CETP region
subgen <- cbind(genotype[,colnames(genotype) %in% CETP$SNP], impCETPgeno)     # CETP subsets from typed and imputed SNPs

# Subset SNPs for only certain genotypes
certain <- apply(as(subgen, 'numeric'), 2, function(x) { all(x %in% c(0,1,2,NA)) })
subgen <- subgen[,certain]

# Subset and order CETP SNPs by position
CETP <- CETP[CETP$SNP %in% colnames(subgen),]
CETP <- arrange(CETP, position)
subgen <- subgen[, order(match(colnames(subgen),CETP$SNP)) ]

# Create LDheatmap
ld <- ld(subgen, subgen, stats="R.squared") # Find LD map of CETP SNPs

ll <- LDheatmap(ld, CETP$position, flip=TRUE, name="myLDgrob", title=NULL)

# Add genes, recombination
llplusgenes <- LDheatmap.addGenes(ll, chr = "chr16", genome = "hg19", genesLocation = 0.01)

# Add plot of -log(p)
library(ggplot2)

plot.new()
llQplot2<-LDheatmap.addGrob(llplusgenes, rectGrob(gp = gpar(col = "white")),height = .34)
pushViewport(viewport(x = 0.483, y= 0.76, width = .91 ,height = .4))

grid.draw(ggplotGrob({
  qplot(position, Neg_logP, data = CETP, xlab="", ylab = "Negative Log P-value", xlim = range(CETP$position),
        asp = 1/10, color = factor(type), colour=c("#000000", "#D55E00")) +
    theme(axis.text.x = element_blank(),
          axis.title.y = element_text(size = rel(0.75)), legend.position = "none",
          panel.background = element_blank(),
          axis.line = element_line(colour = "black")) +
    scale_color_manual(values = c("red", "black"))
}))
```

A regional association plot, provided in Figure 5b, provides similar information for a broader region of the genome. In this case, the blue line at the top represents the SNP-level p-values, the green segments indicate gene regions, and the red lines indicate LD, where we have specified to only include lines betweens SNPs with $r^2 > 0.8$. The following code is used to generate this figure.

```
# ---- step10-e ----
# Create regional association plot

library(postgwas)

# Create data.frame of most significant SNP only
snps<-data.frame(SNP=c("rs1532625"))

# Change column names necessary to run regionalplot function
GWAScomb <- rename(GWAScomb, replace=c(p.value="P", chr="CHR", position="BP"))


# Edit biomartConfigs so regionalplot function
# pulls from human genome build 37/hg19

myconfig <- biomartConfigs$hsapiens
myconfig$hsapiens$gene$host <- "grch37.ensembl.org"
myconfig$hsapiens$gene$mart <- "ENSEMBL_MART_ENSEMBL"
myconfig$hsapiens$snp$host <- "grch37.ensembl.org"
myconfig$hsapiens$snp$mart <- "ENSEMBL_MART_SNP"


# Run regionalplot using HAPMAP data (pop = CEU)
regionalplot(snps, GWAScomb, biomart.config = myconfig, window.size = 400000, draw.snpname = data.frame(
  snps = c("rs1532625", "rs247617"),
  text = c("rs1532625", "rs247617"),
  angle = c(20, 160),
  length = c(1, 1),
  cex = c(0.8)
),
ld.options = list(
  gts.source = 2,
  max.snps.per.window = 2000,
  rsquare.min = 0.8,
  show.rsquare.text = FALSE
),
out.format = list(file = "png", panels.per.page = 4))
```

5.3. *Additional data interrogation using external resources*

Reporting SNP-level findings from association analysis is much more meaningful when a context for the findings is also presented. For example, investigators may want to know whether a statistically significant SNP is within a protein-coding gene, intergenic, or close to a regulatory element (e.g., a methylation mark) in specific tissues or cell types that are relevant to the disease under investigation. Possible external types of data that may be relevant are provided in Supplementary Information C, Table I. These fall into eight general categories, following roughly an order representing the process from DNA information to regulation to expression: (i) SNP; (ii) gene elements; (iii) chromatin state; (iv) epigenetic marks; (v) transcription factor binding; (vi) RNA expression; (vii) SNP–mRNA association; and (viii) other -omics data. We emphasize that this table is not intended to be comprehensive; rather, it provides a glimpse at the vast amount of external data resources available. Data associated with each of the listed categories are available from a wide range of sources (for example, column 2, Supplementary Information C, Table I) and are generally based on a variety of technologies. The UCSC Genome Browser provides a well-devised suite of integrated bioinformatic tools and databases, including many derived from the resources listed in Supplementary Information C, Table I, which allow for further interrogation of GWA findings.

In this section, we provide a very brief introduction to the genome browser, with particular focus on how to view and interpret standard tracks, visualize data corresponding to these tracks, and create custom tracks using new data. To begin, we go to the genome browser gateway at http://genome.ucsc.edu/cgi-bin/hgGateway. We then specify assembly `Feb. 2009 (GRCh37/hg19)`, type the name of our most significant SNP, `rs1532625`, in the field *search term*, and then select *submit*. On the next page, we select `rs1532625 at chr16:57005051-57005551` under the first heading. This choice is elaborated in the succeeding texts. The content of the next page will vary depending on the tracks remaining open at the end of the current user's last session. However, all users will see in the bottom half of this page several classes of tracks, with multiple choices within them. We make the following selections and then press *refresh* either after each change or after all fields have been selected: `Genes and Gene Predictions: UCSC Genes` select *pack*; `mRNA and EST: Human mRNAs` select *dense*; `Regulation: ENCODE Regulation` select *show*; and `Variation: common SNPs(142)` select *pack*. All other fields should be marked *hide*. On the next screen, we zoom out `100x` by pressing the corresponding grey button at the top of the image to acquire a better picture of the entire region.

The resulting image is illustrated in Figure 6. Note that your tracks could be illustrated in a different order. Next is a summary of the elements of this figure. We note first that this is an image of the genetic region surrounding the `rs1532625` SNP (highlighted in a black box in Figure 6) we entered and that the tracks illustrated (also based on our selections) are differentiated by the grey vertical rectangles on the left-hand side of the figure.

Table I. Example data types and select resources for post-analytic interrogation. *Listed resources are intended to provide primary examples and are not comprehensive. [a]National Center for Biotechnology Information (NCBI) dbSNP; [b]ENSEMBL Genome Browser; [c]NCBI RefSeq; [d]NCBI GenBank; [e]The encyclopedia of DNA elements (ENCODE) Project; [f]NIH Roadmap Epigenomics Project; [g]GTex Portal; [h]NCBI Sequence Read Archive (SRA); [i]The Universal Protein Resource Knowledgebase (UniProtKB); [j]The Human Metabolome Database (HMDB).

Example data types	Select data sources*	UCSC genome browser navigation
DNA level data (non-somatic; genEric to all cells):		
I. Coordinates, e.g.		
(1) SNPs	NCBI dbSNP[a], ENSEMBL[b]	Variation: Common SNPs(141)
(2) Insertions and delations (INDELs)		
(3) Copy number variants (CNVs)		
II. Gene elements, e.g.		
(1) Protein-coding genes	NCBI RefSeq[c], NCBI GenBank[d], ENSEMBL[b]	Gene and Gene Predictions: UCSC Genes
(2) Non-protein-coding genes	NCBI RefSeq[c], NCBI GenBank[d], ENSEMBL[b]	Gene and Gene Predictions: UCSC Genes
Cell and tissue-specific regulation:		
III. Chromatin state, e.g.		
(1) DNA hypersensitivity (DNase-Seq)	ENCODE[e], ENSEMBL[b]	Regulation: ENCODE Regulation
(2) FAIRE sequencing	ENCODE[e], ENSEMBL[b]	Regulation: ENC DNase/FAIRE
IV. Epigenetic marks, e.g.		
(1) Methylation promoter marks	ENCODE[e], NIH Roadmap Epigenomics[f]	Regulation: ENCODE Regulation
(2) Methylation enhancer marks	ENCODE[e], NIH Roadmap Epigenomics[f]	Regulation: ENCODE Regulation
(3) Acetylation marks (e.g. #H3K27Ac histone mark)	ENCODE[e], NIH Roadmap Epigenomics[f]	Regulation: ENCODE Regulation
V. Transcription factor binding, e.g.		
(1) ChipSeq data	ENCODE[e], ENSEMBL[b], custom	Regulation: ENCODE Regulation
Cell and tissue-specific expression:		
VI. RNA expression, e.g.		
(1) historic mRNA	NCBI GenBank[d]	mRNA and EST: Human mRNAs
(2) genome-wide cell-specific RNA data (e.g. RNAseq)	ENCODE[e], GTex Portal[g], NCBI SRA[h]	Expression: ENC RNA-seq
VII. SNP-mRNA association, e.g.		
(1) Expression quantitative trait locis (eQTL)	GTex Portal[g], custom	N/A
(2) Allelic imbalance (AI); allele specific expression (ASE)	GTex Portal[g], custom	N/A
Biomarkers endophenotype:		
VIII. Other -omics data, e.g.		
(1) Proteomic (e.g. pQTLs)	UniProtKB[i]	N/A
(2) Metabolomic	HMDB[j]	N/A

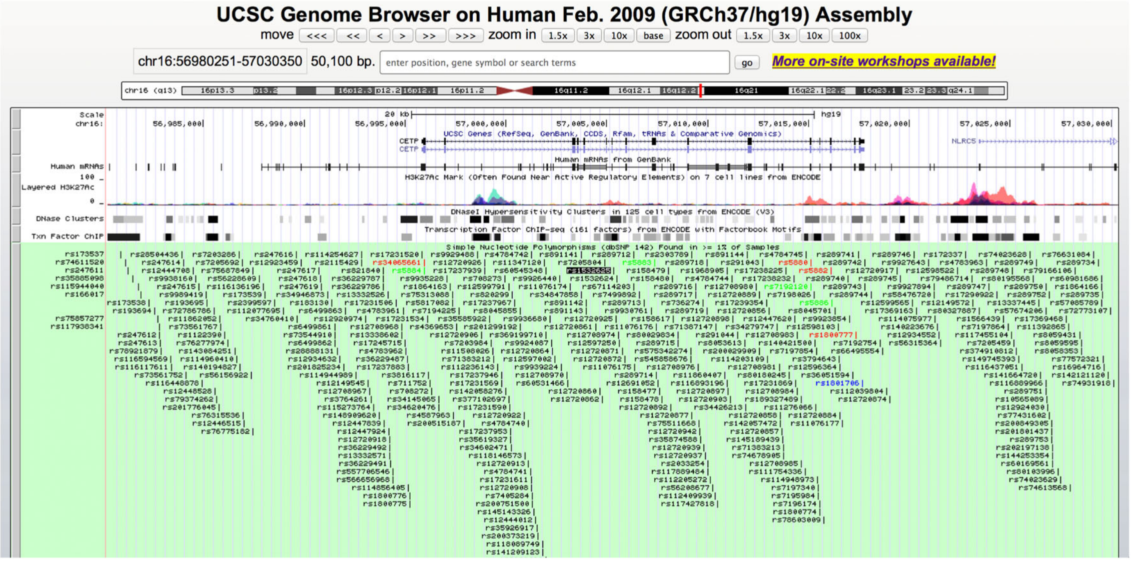

UCSC Genome Browser on Human Feb. 2009 (GRCh37/hg19) Assembly
move <<< << < > >> >>> zoom in 1.5x 3x 10x base zoom out 1.5x 3x 10x 100x
chr16:56980251-57030350 50,100 bp.
enter position, gene symbol or search terms
go
More on-site workshops available!
chr16 (q13)
16p13.3
16p12.3
16p11.2
16q11.2
16q12.1
16q21
16q22.1
16q23.1
Scale
chr16:
20 kb
hg19
56,985,000
56,990,000
56,995,000
57,000,000
57,005,000
57,010,000
57,015,000
57,020,000
57,025,000
57,030,000
UCSC Genes (RefSeq, GenBank, CCDS, Rfam, tRNAs & Comparative Genomics)
CETP
CETP
NLRC5
Human mRNAs from GenBank
Human mRNAs
100 _
Layered H3K27Ac
0 _
H3K27Ac Mark (Often Found Near Active Regulatory Elements) on 7 cell lines from ENCODE
DNaseI Hypersensitivity Clusters in 125 cell types from ENCODE (V3)
DNase Clusters
Transcription Factor ChIP-seq (161 factors) from ENCODE with Factorbook Motifs
Txn Factor ChIP
Simple Nucleotide Polymorphisms (dbSNP 142) Found in >= 1% of Samples
rs1532625

Figure 6. UCSC Genome Browser with specified tracks open.

- Variation: common SNPs(142). The bottom-most track with the heading `'Simple Nucleotide Polymorphisms dbSNP141) Found in >= 1% of Samples'` lists all of the common SNPs by rsNumber that are in this region, in order of their location on the genome. We see that the input SNP, `rs1532625`, is highlighted with a black box. By clicking on this box, the investigator can retrieve additional information about this SNP, including the major and minor alleles and their frequencies, average heterozygosity, and the chromosome and coordinate location based on the current build. Additional information on a track can be found by pressing the grey rectangular vertical bar on the main browser window corresponding to the track. For example, for this track, we find a description of the source of the data (dbSNP build 142.)
- Genes and gene predictions: UCSC genes. The first track at the top of Figure 6 titled `'UCSC Genes'` illustrates all protein-coding and non-protein coding genes that are in close proximity to the SNP we entered. We note based on this figure that our SNP `rs1532625` falls in the protein-coding gene CETP. Additional information about the display conventions and the configuration can be found by pressing the grey rectangular vertical bar on the left-hand side of this track. Additional information about each gene, including the full gene name, coordinates, size, number of exons, and prior GWA evidence, can be retrieved by clicking on the abbreviated gene names in the browser window.
- mRNA and EST: human mRNAs. The next track entitled `'Human mRNAs from GenBank'` provides historical information on whether there have been any reports (indicated by a vertical bar) of the presence of mRNA corresponding to sites on the genome across all tissues and cell types. By clicking on the title, an expanded view of this track is provided (not shown), allowing the user to find additional information. Consider CETP, for example, for which we expect to see mRNA expression in cells relevant to HDL production and/or regulation, such as liver tissue. By selecting mRNA M30185 near the start of CETP, we learn that indeed, the mRNA was found in liver tissue.
- Regulation: ENCODE regulation. The next three tracks entitled '`H3K27Ac Mark`', '`DNaseI Hypersensitivity Clusters in 125 cell types from ENCODE (V3)`', and '`Transcription Factor ChIP-seq (161 factors) from ENCODE with Factorbook Motifs`' all provide information about the presence of cell and tissue-specific regulatory elements. For example, H3K27Ac is a histone mark indicating the degree of acetylation of lysine 27 of the H3 histone protein, which in turn influences how accessible chromatin is for transcription. The color coding of the density plots in this track corresponds to different cell lines. DNA hypersensitivity is a more general measure of whether chromatin is open for transcription, while transcription factor ChIP-seq data provides very specific information about whether given proteins can bind to the specified DNA regions.

It is possible to obtain the data corresponding to each track. As an example, consider the 'common SNPs' track, click on the corresponding grey vertical rectangle to the left of this track, and then select `view table schema` on the next page. If we scroll down, under the heading

Sample Rows, we see all of the data fields associated with this track. Note also that the metadata about this table, including `Database: hg19` and `Primary Table: snp142Common`, are available at the very top of the screen (not shown). These data can be downloaded by selecting `Tools -> Table Browser` on the top menu and then indicating the appropriate fields, including `assembly: Feb. 2009 (GRCh37/hg19); group: Variation; track: Common SNPs(142);` and `table: snp142Common`. The `'get output'` tab at the bottom of these fields displays the data as an ascii formatted file.

We also note that it is possible to create a custom track that is displayed and linked to the information in this browser. To do this, first, we need to create what is a called a BED track file (different than the .bed file discussed in Section 2 in the preceding texts) containing all of the data contributing to this track. A BED track file must include the following five columns: chromosome number, start location, end location (one greater than the start location for individual SNPs), identifier, score, and chromosomal strand, for which the SNP is recognized on the browser. These are included as columns $2-6$ in the table schema discussed earlier. Once we have a properly formatted BED file, we can input it directly into the genome browser as a custom track. In order to add this track to the genome browser, click `'Add Custom Tracks'`, from the main browser window, and upload the new file. This will bring up a page with the details of our new custom track. Click `'go to genome browser'` in order to see the new custom track in the browser.

6. Broader contemporary context and discussion

This tutorial presents fundamental analysis concepts and tools for performing a single GWA analysis and beginning the process of post-analytic interrogation. Increasingly, GWA analysis results are being combined across a large number of studies to improve power for novel discoveries. For example, the Global Lipids Genetics Consortium recently reported the results of a meta-analysis of 188,577 individuals across 60 studies, resulting in discovery of 62 novel loci for blood lipids [38, 39]. Likewise, the CARDIoGRAM consortium and the CARDIoGRAMplusC4D consortium metadata (which include the PennCATH data used throughout this tutorial) include GWA study results based on 194,427 individuals and contributed to the discovery of 46 loci associated with coronary artery disease [21, 27]. An overview of methods for GWA meta-analysis can be found, for example, in [40], with study-specific details typically provided in the Supplementary Information of associated manuscripts (e.g., [21, 38]). Importantly, depending on consortium data harmonization procedures, we see variation in the extent and timing of SNP and sample-level filtering, as well as the criteria for including PCs and other covariates in the final model fitting procedure. Thus, flexibility in the step-by-step procedure described herein may be required.

Traditionally, a two-stage design was used, with replication of top suggestive findings ($p < 5 \times 10^{-6}$) in a large independent study sample of like design and like ethnicity [41, 42]. A threshold for significance is set in the second stage based on the number of SNPs carried forward and typically required that the SNP met the widely held genome-wide significance for all common SNPs ($p < 5 \times 10^{-8}$) in a combined meta-analysis. However, often in contemporary studies, GWA data are available simultaneously in several studies, and a meta-analysis is performed on all SNPs across all studies in the second stage, and the significance threshold of $p < 5 \times 10^{-8}$ is applied. Typically, additional replications are sought in different ethnicities and in study designs that are not identical, for example, different age groups and with different traits that mark the same disease, in order to evaluate generalizability.

Several analytic strategies have been developed, which serve to complement the single-SNP level testing approach described in this tutorial, including gene-level testing strategies that require raw genotype data [16, 43] and gene-level testing approaches that instead leverage summary output (in the form of test statistics or p-values) of the GWA analysis presented herein [15, 44]. A broad assortment of sophisticated analytic methods has also been described for gene set enrichment or biological pathway analysis [18, 45–48]. Additional methods have been described to address the unique challenges inherent in rare variant analysis [8–10] in which the low frequencies of mutations can result in insufficient power to assess significance without regional context. Finally, linear mixed models have been described as an alternative strategy for GWA analysis, which can account for family relatedness and population substructure [49–54]. An additional recommended resource for more in-depth post-processing of GWA findings, including gene and network-based analysis, is provided in [55].

Defining the best practices for GWA data pre-processing, analysis, and post-analytic interrogation within a framework that is logical and comprehensive for statisticians is essential for standardizing methods and ensuring reproducible and comparable findings across studies. This tutorial outlines the key features that are integral to GWA studies, and provides the R code that can been applied to implement each of these features accurately. We emphasize the use of R as GWA studies are typically part of a larger data analytic investigation (e.g., gene-based analysis as described earlier), and it is straightforward to integrate the R code provided into larger statistical coding efforts. Alternative open-source, freely available, high-performance programing languages, for example, Julia, which was designed specifically for parallelism and cloud computing [56], may ultimately serve to provide additional functionalities in this big data analytic realm, particularly as post-analytic interrogation becomes more integrated with primary GWA analysis.

Additional resources:

1000 Genomes (http://www.1000genomes.org/)
BEAGLE (http://faculty.washington.edu/browning/beagle/beagle.html)
BIMBAM (http://stephenslab.uchicago.edu/software.html#bimbam)
Bioconductor (http://www.bioconductor.org/)
dbSNP build 141(http://ftp.ncbi.nih.gov/snp)
Encyclopedia of DNA elements (ENCODE) Project (https://www.encodeproject.org/)
ENSEMBL Genome Browser (http://www.ensembl.org/)
GenABEL project (http://www.genabel.org/)
Genome Browser gateway (http://genome.ucsc.edu/cgi-bin/hgGateway)
GTex Portal (http://www.gtexportal.org/home/)
Human Metabolome Database (HMDB) (http://www.hmdb.ca/)
HapMap (http://hapmap.ncbi.nlm.nih.gov/)
IMPUTE2 (https://mathgen.stats.ox.ac.uk/impute/impute_v2.html)
Julia (http://julialang.org)
KING (http://people.virginia.edu/~wc9c/KING/)
MACH/MACH2qtl/dat (http://www.sph.umich.edu/csg/abecasis/MACH/index.html)
NCBI GenBank (http://www.ncbi.nlm.nih.gov/genbank/)
NCBI RefSeq (http://www.ncbi.nlm.nih.gov/refseq/)
NIH Roadmap Epigenomics Project (http://www.roadmapepigenomics.org/)
NCBI Sequence Read Archive (SRA) (http://www.ncbi.nlm.nih.gov/sra)
R (http://cran.r-project.org/)
PLINK (http://pngu.mgh.harvard.edu/~purcell/plink/)
ProbABEL (http://www.genabel.org/packages/ProbABEL)
snpStats (http://bioconductor.org/packages/release/bioc/html/snpStats.html)
SNPTEST (https://mathgen.stats.ox.ac.uk/genetics_software/snptest/snptest.html)
UCSC Genome Browser (http://genome.ucsc.edu/))
Universal Protein Resource Knowledgebase (UniProtKB) (http://www.uniprot.org/)

References

1. Laird NM, Lange C. *The Fundamentals of Modern Statistical Genetics*. Springer-Verlag: New York, 2011.
2. Ziegler A, Konig IR, Pahlke F. *A Statistical Approach to Genetic Epidemiology: Concepts and Applications, with an e-Learning Platform* 2nd Updated edition. Wiley-Blackwell, 2010.
3. Foulkes AS. *Applied Statistical Genetics with r*. Springer-Verlag: New York, 2009.
4. Clarke GM, Anderson CA, Pettersson FH, Cardon LR, Morris AP, Zondervan KT. Basic statistical analysis in genetic case-control studies. *Nature Protocols* 2011; **6**(2):121–133.
5. Balding DJ. A tutorial on statistical methods for population association studies. *Nature Reviews Genetics* 2006; **7**(10): 781–791.
6. Laurie CC, Doheny KF, Mirel DB, Pugh EW, Bierut LJ, Bhangale T, Boehm F, Caporaso NE, Cornelis MC, Edenberg HJ, Gabriel SB, Harris EL, Hu FB, Jacobs KB, Kraft P, Landi MT, Lumley T, Manolio TA, McHugh C, Painter I, Paschall J, Rice JP, Rice KM, Zheng X, Weir BS. Quality control and quality assurance in genotypic data for genome-wide association studies. *Genetic Epidemiology* 2010; **34**(6):591–602.
7. Gogarten SM, Bhangale T, Conomos MP, Laurie CA, McHugh CP, Painter I, Zheng X, Crosslin DR, Levine D, Lumley T, Nelson SC, Rice K, Shen J, Swarnkar R, Weir BS, Laurie CC. GWASTools: an R/Bioconductor package for quality control and analysis of genome-wide association studies. *Bioinformatics* 2012; **28**(24):3329–3331.
8. Li B, Leal SM. Methods for detecting associations with rare variants for common diseases: application to analysis of sequence data. *American Journal of Human Genetics* 2008; **83**(3):311–321.
9. Liu DJ, Peloso GM, Zhan X, Holmen OL, Zawistowski M, Feng S, Nikpay M, Auer PL, Goel A, Zhang H, Peters U, Farrall M, Orho-Melander M, Kooperberg C, McPherson R, Watkins H, Willer CJ, Hveem K, Melander O, Kathiresan S, Abecasis GR. Meta-analysis of gene-level tests for rare variant association. *Nature Genetics* 2014; **46**(2):200–204.

10. Hu YJ, Berndt SI, Gustafsson S, Ganna A, Hirschhorn J, North KE, Ingelsson E, Lin D-Y, Genetic Investigation of Anthropometric Traits (GIANT) Consortium. Meta-analysis of gene-level associations for rare variants based on single-variant statistics. *American Journal of Human Genetics* 2013; **93**(2):236–248.
11. Chen MH, Yang Q. GWAF: an R package for genome-wide association analyses with family data. *Bioinformatics* 2010; **26**(4):580–581.
12. Ott J, Kamatani Y, Lathrop M. Family-based designs for genome-wide association studies. *Nature Reviews Genetics* 2011; **12**(7):465–474.
13. Abecasis GR, Altshuler D, Auton A, Brooks LD, Durbin RM, Gibbs RA, Hurles ME, McVean GA. A map of human genome variation from population-scale sequencing. *Nature* 2010; **467**(7319):1061–1073.
14. Li MX, Gui HS, Kwan JS, Sham PC. GATES: a rapid and powerful gene-based association test using extended Simes procedure. *The American Journal of Human Genetics* 2011; **88**(3):283–293.
15. Liu JZ, McRae AF, Nyholt DR, Medland SE, Wray NR, Brown KM, Hayward NK, Montgomery GW, Visscher PM, Martin NG, Macgregor S, Mann GJ, Kefford RF, Hopper JL, Aitken JF, Giles GG, Armstrong BK. A versatile gene-based test for genome-wide association studies. *The American Journal of Human Genetics* 2010; **87**(1):139–145.
16. Huang H, Chanda P, Alonso A, Bader JS, Arking DE. Gene-based tests of association. *PLoS Genetics* 2011; **7**(7):e1002177.
17. Wang L, Jia P, Wolfinger RD, Chen X, Grayson BL, Aune TM, Zhao Z. An efficient hierarchical generalized linear mixed model for pathway analysis of genome-wide association studies. *Bioinformatics* 2011; **27**(5):686–692.
18. Segre AV, Groop L, Mootha VK, Daly MJ, Altshuler D. Common inherited variation in mitochondrial genes is not enriched for associations with type 2 diabetes or related glycemic traits. *PLoS Genetics* 2010; **6**(8).
19. Thomas D. Gene–environment-wide association studies: emerging approaches. *Nature Reviews Genetics* 2010; **11**(4): 259–272.
20. Reilly MP, Li M, He J, Ferguson J, Stylianou I, Mehta N, Burnett M, Devaney J, Knouff C, Thompson J, Horne B, Stewart A, Assimes T, Wild P, Allayee H, Nitschke P, Patel R. Identification of ADAMTS7 as a novel locus for coronary atherosclerosis and association of ABO with myocardial infarction in the presence of coronary atherosclpmid21378990erosis: two genome-wide association studies. *Lancet* 2011; **377**(9763):383–392.
21. Schunkert H, Konig IR, Kathiresan S, Reilly MP, *et al.* Large-scale association analysis identifies 13 new susceptibility loci for coronary artery disease. *Nature Genetics* 2011; **43**(4):333–338.
22. Anderson CA, Pettersson FH, Clarke GM, Cardon LR, Morris AP, Zondervan KT. Data quality control in genetic case-control association studies. *Nature Protocols* 2010; **5**(9):1564–1573.
23. Wright S. Coefficients of inbreeding and relationship. *American Naturalist* 1922; **56**:330–338.
24. Manichaikul A, Mychaleckyj JC, Rich SS, Daly K, Sale M, Chen WM. Robust relationship inference in genome-wide association studies. *Bioinformatics* 2010; **26**(22):2867–2873.
25. Marchini J, Howie B, Myers S, McVean G, Donnelly P. A new multipoint method for genome-wide association studies by imputation of genotypes. *Nature Genetics* 2007; **39**(7):906–913.
26. Edmondson AC, Braund PS, Stylianou IM, Khera AV, Nelson CP, Wolfe ML, Derohannessian SL, Keating BJ, Qu L, He J, Tobin MD, Tomaszewski M, Baumert J, Klopp N, Doring A, Thorand B, Li M, Reilly MP, Koenig W, Samani NJ, Rader DJ. Dense genotyping of candidate gene loci identifies variants associated with high-density lipoprotein cholesterol. *Circulation: Cardiovascular Genetics* 2011; **4**:145–155.
27. Deloukas P, Kanoni S, Willenborg C, Farrall M, Assimes TL, Thompson JR, Ingelsson E, Saleheen D, Erdmann J, Goldstein BA, Stirrups K, Konig IR, Cazier JB, Johansson A, Hall AS, Lee JY, Willer CJ, Chambers JC, Esko T, Folkersen L, Goel A, Grundberg E, Havulinna AS, Ho WK, Hopewell JC, Eriksson N, Kleber ME, Kristiansson K, Lundmark P, Lyytikainen LP, Rafelt S, Shungin D, Strawbridge RJ, Thorleifsson G, Tikkanen E, Van Zuydam N, Voight BF, Waite LL, Zhang W, Ziegler A, Absher D, Altshuler D, Balmforth AJ, Barroso I, Braund PS, Burgdorf C, Claudi-Boehm S, Cox D, Dimitriou M, Do R, Doney AS, El Mokhtari N, Eriksson P, Fischer K, Fontanillas P, Franco-Cereceda A, Gigante B, Groop L, Gustafsson S, Hager J, Hallmans G, Han BG, Hunt SE, Kang HM, Illig T, Kessler T, Knowles JW, Kolovou G, Kuusisto J, Langenberg C, Langford C, Leander K, Lokki ML, Lundmark A, McCarthy MI, Meisinger C, Melander O, Mihailov E, Maouche S, Morris AD, Muller-Nurasyid M, Nikus K, Peden JF, Rayner NW, Rasheed A, Rosinger S, Rubin D, Rumpf MP, Schafer A, Sivananthan M, Song C, Stewart AF, Tan ST, Thorgeirsson G, van der Schoot CE, Wagner PJ, Wells GA, Wild PS, Yang TP, Amouyel P, Arveiler D, Basart H, Boehnke M, Boerwinkle E, Brambilla P, Cambien F, Cupples AL, de Faire U, Dehghan A, Diemert P, Epstein SE, Evans A, Ferrario MM, Ferrieres J, Gauguier D, Go AS, Goodall AH, Gudnason V, Hazen SL, Holm H, Iribarren C, Jang Y, Kahonen M, Kee F, Kim HS, Klopp N, Koenig W, Kratzer W, Kuulasmaa K, Laakso M, Laaksonen R, Lee JY, Lind L, Ouwehand WH, Parish S, Park JE, Pedersen NL, Peters A, Quertermous T, Rader DJ, Salomaa V, Schadt E, Shah SH, Sinisalo J, Stark K, Stefansson K, Tregouet DA, Virtamo J, Wallentin L, Wareham N, Zimmermann ME, Nieminen MS, Hengstenberg C, Sandhu M S, Pastinen T, Syvanen AC, Hovingh GK, Dedoussis G, Franks PW, Lehtimaki T, Metspalu A, Zalloua PA, Siegbahn A, Schreiber S, Ripatti S, Blankenberg SS, Perola M, Clarke R, Boehm BO, O'Donnell C, Reilly MP, Marz W, Collins R, Kathiresan S, Hamsten A, Kooner JS, Thorsteinsdottir U, Danesh J, Palmer CN, Roberts R, Watkins H, Schunkert H, Samani NJ. Large-scale association analysis identifies new risk loci for coronary artery disease. *Nature Genetics* 2013; **45**(1): 25–33.
28. Li MX, Yeung JM, Cherny SS, Sham PC. Evaluating the effective numbers of independent tests and significant p-value thresholds in commercial genotyping arrays and public imputation reference datasets. *Human Genetics* 2012; **131**(5): 747–756.
29. Li Y, Willer C, Sanna S, Abecasis G. Genotype imputation. *Annual Review of Genomics and Human Genetics* 2009; **10**:387–406.
30. Aulchenko YS, Struchalin MV, van Duijn CM. ProbABEL package for genome-wide association analysis of imputed data. *BMC Bioinformatics* 2010; **11**:134.
31. Browning BL, Browning SR. A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. *The American Journal of Human Genetics* 2009; **84**(2):210–223.

32. Servin B, Stephens M. Imputation-based analysis of association studies: candidate regions and quantitative traits. *PLoS Genetics* 2007; **3**(7):e114.
33. Pei YF, Zhang L, Li J, Deng HW. Analyses and comparison of imputation-based association methods. *PLoS One* 2010; **5**(5):e10827.
34. Marchini J, Howie B. Genotype imputation for genome-wide association studies. *Nature Reviews Genetics* 2010; **11**(7):499–511.
35. Devlin B, Roeder K, Wasserman L. Genomic control, a new approach to genetic-based association studies. *Theoretical Population Biology* 2001; **60**(3):155–166.
36. Devlin B, Roeder K. Genomic control for association studies. *Biometrics* 1999; **55**(4):997–1004.
37. Freedman ML, Reich D, Penney KL, McDonald GJ, Mignault AA, Patterson N, Gabriel SB, Topol EJ, Smoller JW, Pato CN, Pato MT, Petryshen TL, Kolonel LN, Lander ES, Sklar P, Henderson B, Hirschhorn JN, Altshuler D. Assessing the impact of population stratification on genetic association studies. *Nature Genetics* 2004; **36**(4):388–393.
38. Teslovich TM, Musunuru K, Smith AV, Teslovich TM, Musunuru K, Smith AV, Edmondson AC, Stylianou IM, Koseki M, Pirruccello JP, Ripatti S, Chasman DI, Willer CJ, Johansen CT, Fouchier SW, Isaacs A, Peloso GM, Barbalic M, Ricketts SL, Bis JC, Aulchenko YS, Thorleifsson G, Feitosa MF. Biological, clinical and population relevance of 95 loci for blood lipids. *Nature* 2010; **466**:707–713.
39. Willer CJ, Schmidt EM, Sengupta S, Peloso GM, Gustafsson S, Kanoni S, Ganna A, Chen J, Buchkovich ML, Mora S, *et al*. Discovery and refinement of loci associated with lipid levels. *Nature Genetics* 2013; **45**(11):1274–1283.
40. Evangelou E, Ioannidis JP. Meta-analysis methods for genome-wide association studies and beyond. *Nature Reviews Genetics* 2013; **14**(6):379–389.
41. Wang H, Thomas DC, Pe'er I, Stram DO. Optimal two-stage genotyping designs for genome-wide association scans. *Genetic Epidemiology* 2006; **30**(4):356–368.
42. Skol A D, Scott L J, Abecasis G R, Boehnke M. Optimal designs for two-stage genome-wide association studies. *Genetic Epidemiology* 2007; **31**(7):776–788.
43. Wu MC, Kraft P, Epstein MP, Taylor DM, Chanock SJ, Hunter DJ, Lin X. Powerful SNP-set analysis for case-control genome-wide association studies. *American Journal of Human Genetics* 2010; **86**(6):929–942.
44. Qian J, Reed E, Nunez S, Qu L, Reilly MP, Foulkes AS. Testing class-level genetic associations using single-element summary statistics; **under review**.
45. Mootha VK, Lindgren CM, Eriksson KF, Subramanian A, Sihag S, Lehar J, Puigserver P, Carlsson E, Ridderstrale M, Laurila E, Houstis N, Daly MJ, Patterson N, Mesirov JP, Golub TR, Tamayo P, Spiegelman B, Lander ES, Hirschhorn JN, Altshuler D, Groop LC. PGC-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature Genetics* 2003; **34**(3):267–273.
46. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, Mesirov JP. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National academy of Sciences of the United States of America* 2005; **102**: 15545–15550.
47. Peng G, Luo L, Siu H, Zhu Y, Hu P, Hong S, Zhao J, Zhou X, Reveille JD, Jin L, Amos CI, Xiong M. Gene and pathway-based second-wave analysis of genome-wide association studies. *European Journal of Human Genetics* 2010; **18**(1): 111–117.
48. Weng L, Macciardi F, Subramanian A, Guffanti G, Potkin SG, Yu Z, Xie X. SNP-based pathway enrichment analysis for genome-wide association studies. *BMC Bioinformatics* 2011; **12**:99.
49. Yu J, Pressoir G, Briggs WH, Vroh Bi I, Yamasaki M, Doebley JF, McMullen MD, Gaut BS, Nielsen DM, Holland JB, Kresovich S, Buckler ES. A unified mixed-model method for association mapping that accounts for multiple levels of relatedness. *Nature Genetics* 2006; **38**(2):203–208.
50. Lippert C, Listgarten J, Liu Y, Kadie CM, Davidson RI, Heckerman D. FaST linear mixed models for genome-wide association studies. *Nature Methods* 2011; **8**(10):833–835.
51. Price AL, Zaitlen NA, Reich D, Patterson N. New approaches to population stratification in genome-wide association studies. *Nature Reviews Genetics* 2010; **11**(7):459–463.
52. Kang HM, Sul JH, Service SK, Zaitlen NA, Kong SY, Freimer NB, Sabatti C, Eskin E. Variance component model to account for sample structure in genome-wide association studies. *Nature Genetics* 2010; **42**(4):348–354.
53. Zhou X, Stephens M. Genome-wide efficient mixed-model analysis for association studies. *Nature Genetics* 2012; **44**(7):821–824.
54. Zhou X, Stephens M. Efficient multivariate linear mixed model algorithms for genome-wide association studies. *Nature Methods* 2014; **11**(4):407–409.
55. Hiersche M, Ruhle F, Stoll M. Postgwas: advanced GWAS interpretation in R. *PLoS One* 2013; **8**(8):e71775.
56. Bezanson J, Edelman A, Karpinski S, Shah VB. Julia: a fresh approach to numerical computing. *arXiv preprint arXiv:1411.1607* 2014.

Supporting information

Additional supporting information may be found in the online version of this article at the publisher's web-site. Complete R scripts and data associated with this tutorial, as well as additional instructional resources, are available for download as part of the Open Resources for Statistical Genomics (ORSG) project http://www.stat-gen.org.