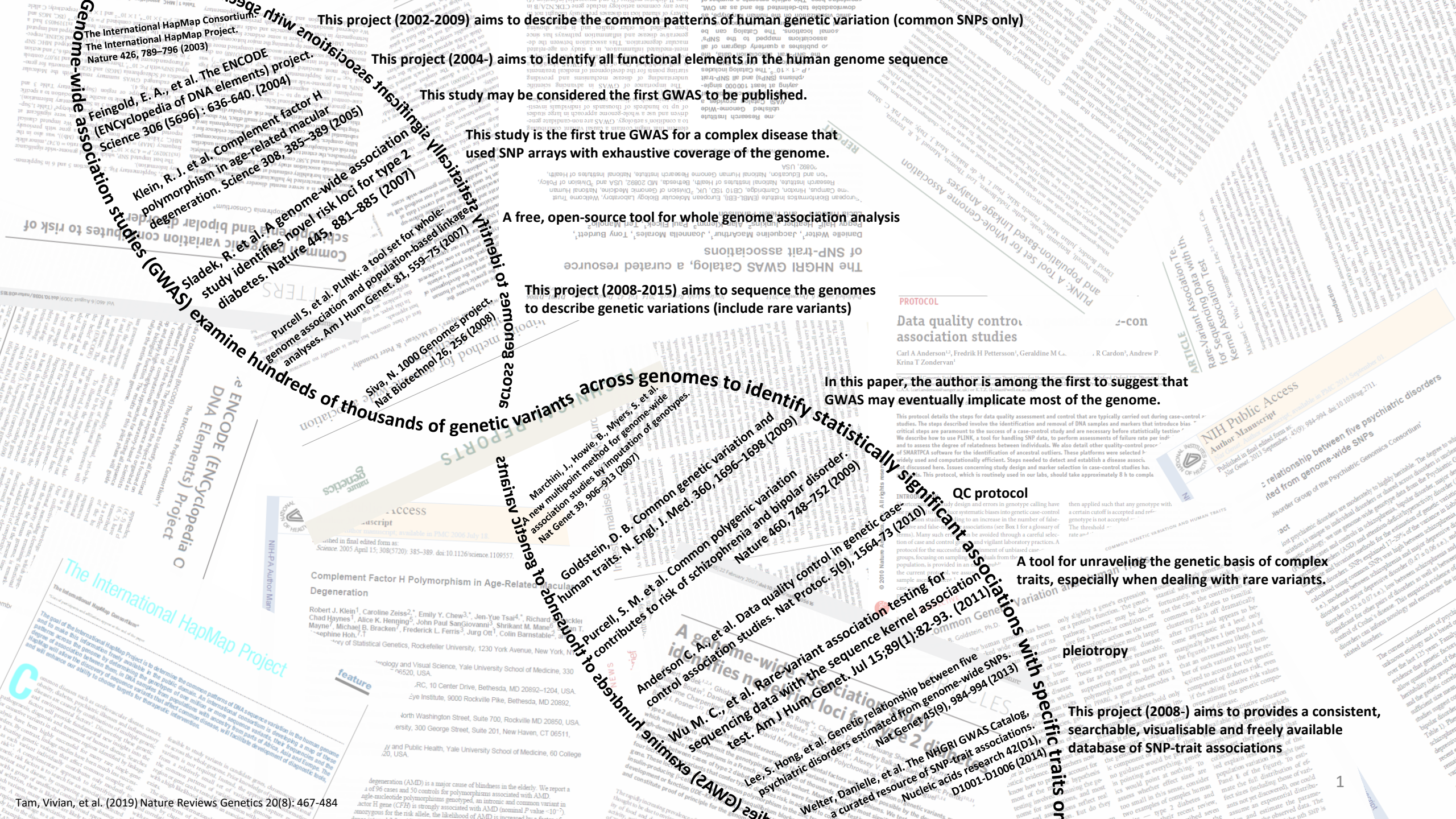
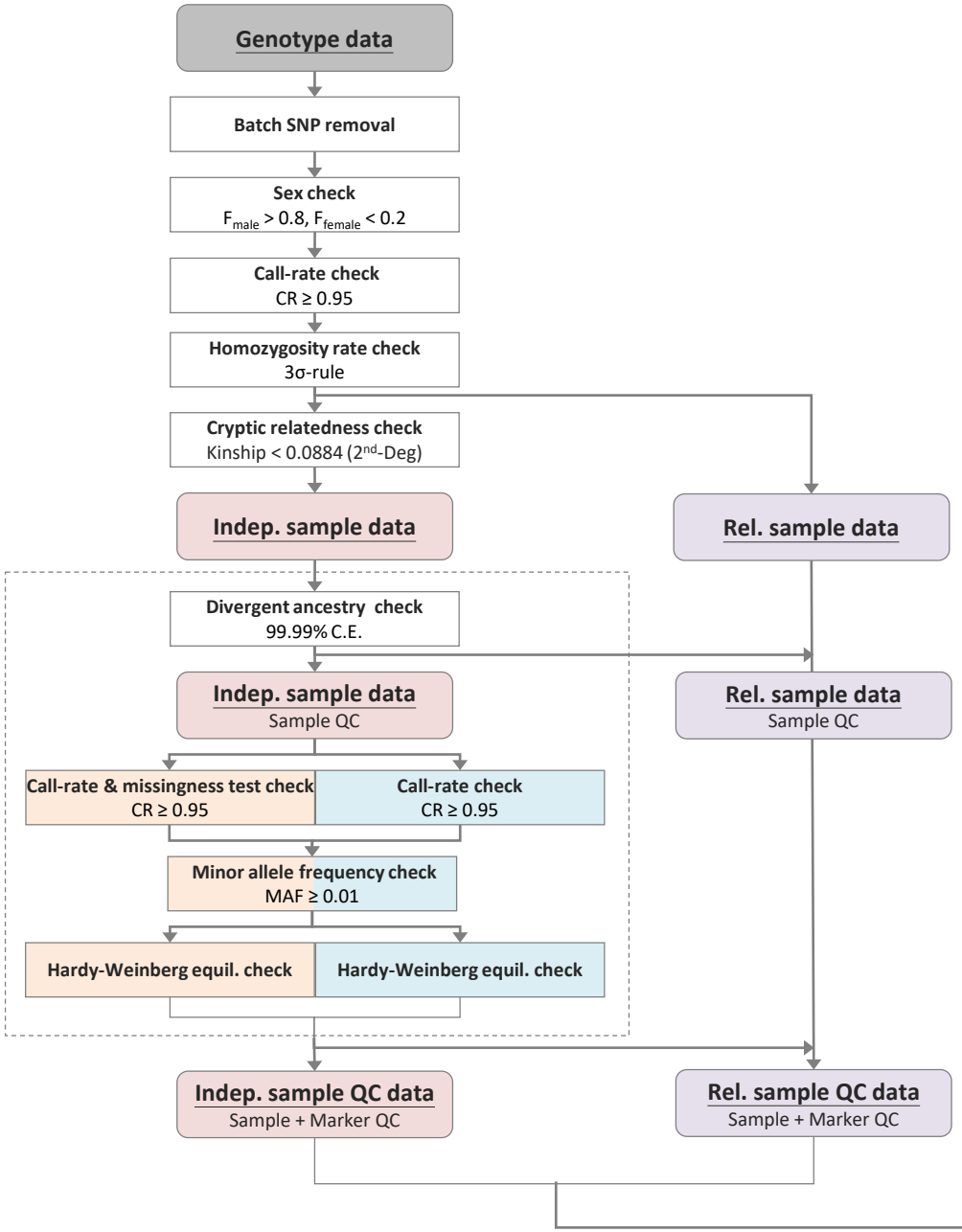
生物資訊與分析

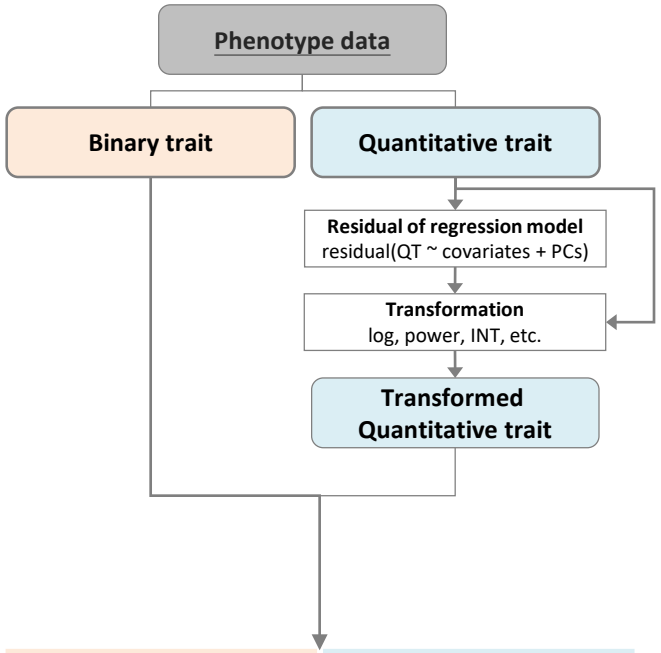
GWAS: From QC to trait associations

陳佳煒 (Jia-Wei Chen)

2024/04/02


This project (2002-2009) aims to describe the common patterns of human genetic variation (common SNPs only)
This project (2004-) aims to identify all functional elements in the human genome sequence
This study may be considered the first GWAS to be published.
This study is the first true GWAS for a complex disease that used SNP arrays with exhaustive coverage of the genome.
A free, open-source tool for whole genome association analysis
This project (2008-2015) aims to sequence the genomes to describe genetic variations (include rare variants)
In this paper, the author is among the first to suggest that GWAS may eventually implicate most of the genome.
QC protocol
A tool for unraveling the genetic basis of complex traits, especially when dealing with rare variants.
pleiotropy
This project (2008-) aims to provides a consistent, searchable, visualisable and freely available database of SNP-trait associations
Genome-wide association studies (GWAS) examine hundreds of thousands of genetic variants across genomes to identify statistically significant associations with specific traits or
The International HapMap Consortium. The International HapMap Project. Nature 426, 789–796 (2003)
Feingold, E. A., et al. The ENCODE (ENCyclopedia of DNA elements) project. Science 306 (5696) : 636-640. (2004)
Klein, R. J. et al. Complement factor H polymorphism in age-related macular degeneration. Science 308, 385–389 (2005)
Sladek, R. et al. A genome-wide association study identifies novel risk loci for type 2 diabetes. Nature 445, 881–885 (2007)
Purcell S, et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet. 81, 559–75 (2007)
Siva, N. 1000 Genomes project. Nat Biotechnol 26, 256 (2008)
Marchini, J., Howie, B., Myers, S. et al. A new multipoint method for genome-wide association studies by imputation of genotypes. Nat Genet 39, 906–913 (2007)
Goldstein, D. B. Common genetic variation and human traits. N. Engl. J. Med. 360, 1696–1698 (2009)
Purcell, S. M. et al. Common polygenic variation contributes to risk of schizophrenia and bipolar disorder. Nature 460, 748–752 (2009)
Anderson C. A., et al. Data quality control in genetic case-control association studies. Nat Protoc. 5(9), 1564-73 (2010)
Wu M. C., et al. Rare-variant association testing for sequencing data with the sequence kernel association test. Am J Hum Genet. Jul 15;89(1):82-93. (2011)
Lee, S. Hong, et al. Genetic relationship between five psychiatric disorders estimated from genome-wide SNPs. Nat Genet 45(9), 984-994 (2013)
Welter, Danielle, et al. The NHGRI GWAS Catalog, a curated resource of SNP-trait associations. Nucleic acids research 42(D1), D1001-D1006 (2014)
Tam, Vivian, et al. (2019) Nature Reviews Genetics 20(8): 467-484
1





	Binary trait	Quantitative trait
Indep. sample	**Firth logistic regression** with covariates and population structure adjustments	**Linear regression** with/without covariates and population structure for transformed quantitative traits
Rel. sample	**Logistic mixed model** with covariates and population structure adjustments	**linear mixed model** with/without covariates and population structure for transformed quantitative traits

- **Fine mapping**
- **Functional annotation**
- **GWEIS**
- **Rare variant analysis**
- **Meta-analysis**
- **Heritability**
- **TWAS**
- **Colocalization**
- **EWAS**
- **PheWAS**
- **Gene analysis**
- **Gene-set/pathway analysis**
- **MR**
- **PRS**

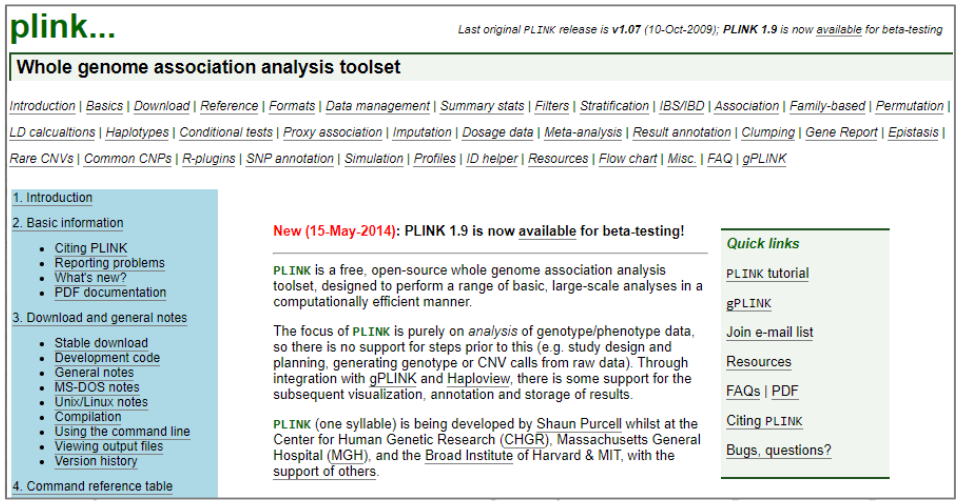
Software – PLINK

REPORT

PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses

Shaun Purcell, Benjamin Neale, Kathe Todd-Brown, Lori Thomas, Manuel A. R. Ferreira, David Bender, Julian Maller, Pamela Sklar, Paul I. W. de Bakker, Mark J. Daly, and Pak C. Sham

Whole-genome association studies (WGAS) bring new computational, as well as analytic, challenges to researchers. Many existing genetic-analysis tools are not designed to handle such large data sets in a convenient manner and do not necessarily exploit the new opportunities that whole-genome data bring. To address these issues, we developed PLINK, an open-source C/C++ WGAS tool set. With PLINK, large data sets comprising hundreds of thousands of markers genotyped for thousands of individuals can be rapidly manipulated and analyzed in their entirety. As well as providing tools to make the basic analytic steps computationally efficient, PLINK also supports some novel approaches to whole-genome data that take advantage of whole-genome coverage. We introduce PLINK and describe the five main domains of function: data management, summary statistics, population stratification, association analysis, and identity-by-descent estimation. In particular, we focus on the estimation and use of identity-by-state and identity-by-descent information in the context of population-based whole-genome studies. This information can be used to detect and correct for population stratification and to identify extended chromosomal segments that are shared identical by descent between very distantly related individuals. Analysis of the patterns of segmental sharing has the potential to map disease loci that contain multiple rare variants in a population-based linkage analysis.



plink...

*Last original PLINK release is **v1.07** (10-Oct-2009); **PLINK 1.9** is now available for beta-testing*

Whole genome association analysis toolset

Introduction | Basics | Download | Reference | Formats | Data management | Summary stats | Filters | Stratification | IBS/IBD | Association | Family-based | Permutation | LD calcualtions | Haplotypes | Conditional tests | Proxy association | Imputation | Dosage data | Meta-analysis | Result annotation | Clumping | Gene Report | Epistasis | Rare CNVs | Common CNPs | R-plugins | SNP annotation | Simulation | Profiles | ID helper | Resources | Flow chart | Misc. | FAQ | gPLINK

1. Introduction
2. Basic information
   - Citing PLINK
   - Reporting problems
   - What's new?
   - PDF documentation
3. Download and general notes
   - Stable download
   - Development code
   - General notes
   - MS-DOS notes
   - Unix/Linux notes
   - Compilation
   - Using the command line
   - Viewing output files
   - Version history
4. Command reference table

New (15-May-2014): **PLINK 1.9 is now available for beta-testing!**

PLINK is a free, open-source whole genome association analysis toolset, designed to perform a range of basic, large-scale analyses in a computationally efficient manner.

The focus of PLINK is purely on *analysis* of genotype/phenotype data, so there is no support for steps prior to this (e.g. study design and planning, generating genotype or CNV calls from raw data). Through integration with gPLINK and Haploview, there is some support for the subsequent visualization, annotation and storage of results.

PLINK (one syllable) is being developed by Shaun Purcell whilst at the Center for Human Genetic Research (CHGR), Massachusetts General Hospital (MGH), and the Broad Institute of Harvard & MIT, with the support of others.

Quick links

PLINK tutorial

gPLINK

Join e-mail list

Resources

FAQs | PDF

Citing PLINK

Bugs, questions?

from candidate-gene to unbiased whole-genome searches. The standard logic of the WGAS design implicitly assumes that common variants with modest effects on disease frequently exist and explain substantial proportions of variation (i.e., the common disease/common variant [CD/

signal from noise. To a large extent, this problem can be assuaged by moderate increases in sample size: basic power calculations show that maintaining the same power when performing an exponentially larger number of Bonferroni-corrected tests requires only a linear increase in sample

From the Center for Human Genetic Research, Massachusetts General Hospital, Boston (S.P.; B.N.; K.T.-B.; L.T.; M.A.R.F.; D.B.; J.M.; P.S.; P.I.W.d.B.; M.J.D.); Broad Institute of Harvard and Massachusetts Institute of Technology, Cambridge, MA (S.P.; B.N.; D.B.; J.M.; P.S.; P.I.W.d.B.; M.J.D.); Institute of Psychiatry, University of London, London (B.N.); and Genome Research Center, University of Hong Kong, Hong Kong (P.C.S.)

Received February 6, 2007; accepted for publication May 2, 2007; electronically published July 25, 2007.

Address for correspondence and reprints: Dr. Shaun Purcell, Center for Human Genetic Research, Massachusetts General Hospital, Room 6.254, CPZ-N, 185 Cambridge Street, Boston, MA, 02114. E-mail: shaun@pngu.mgh.harvard.edu

Am. J. Hum. Genet. 2007;81:559–575. 0002-9297/2007/8103-0013$15.00
DOI: 10.1086/519795

Chang *et al. GigaScience* (2015) 4:7
DOI 10.1186/s13742-015-0047-8

GIGA SCIENCE

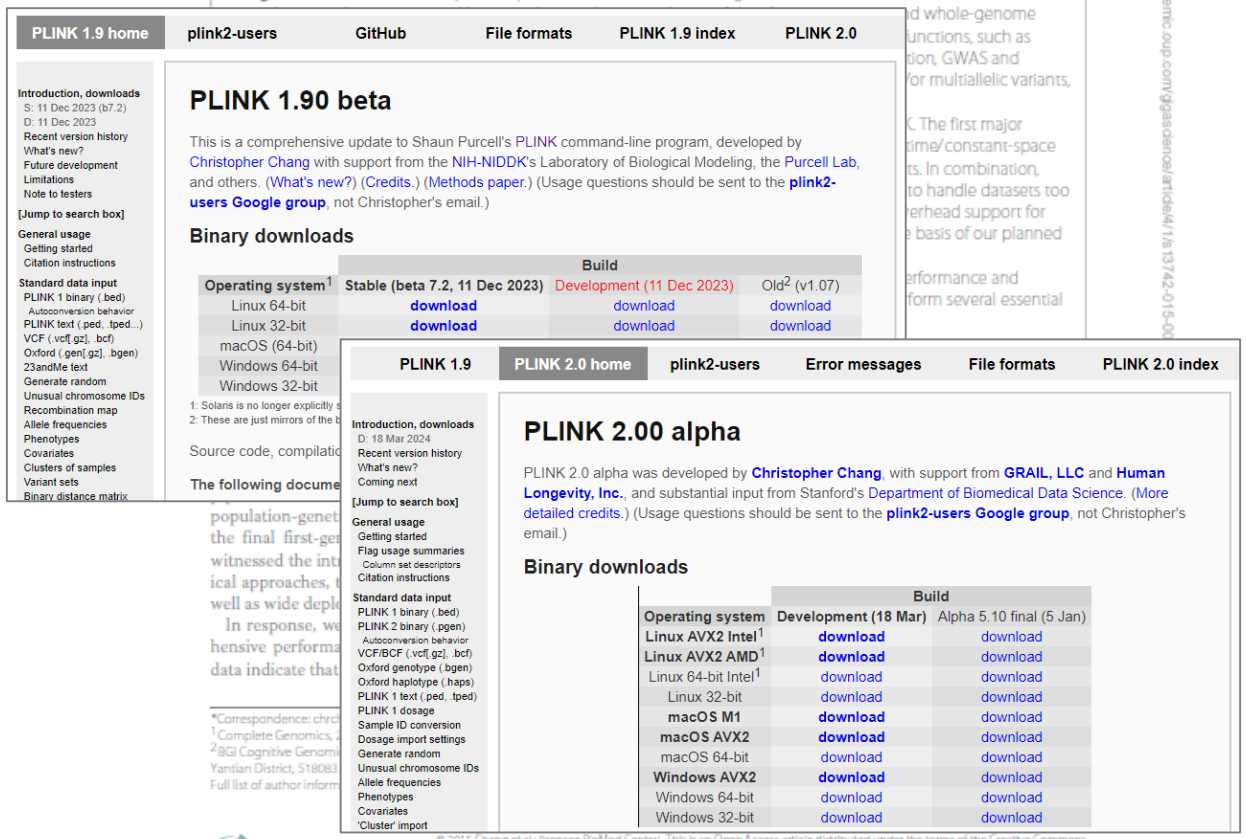
TECHNICAL NOTE Open Access

Second-generation PLINK: rising to the challenge of larger and richer datasets

Christopher C Chang[1,2*], Carson C Chow[3], Laurent CAM Tellier[2,4], Shashaank Vattikuti[3], Shaun M Purcell[5,6,7,8] and James J Lee[3,9]

Abstract

Background: PLINK 1 is a widely used open-source C/C++ toolset for genome-wide association studies (GWAS) and



PLINK 1.9 home | plink2-users | GitHub | File formats | PLINK 1.9 index | PLINK 2.0

Introduction, downloads
S: 11 Dec 2023 (b7.2)
D: 11 Dec 2023
Recent version history
What's new?
Future development
Limitations
Note to testers
[Jump to search box]
General usage
Getting started
Citation instructions
Standard data input
PLINK 1 binary (.bed)
Autoconversion behavior
PLINK text (.ped, .tped...)
VCF (.vcf[.gz], .bcf)
Oxford (.gen[.gz], .bgen)
23andMe text
Generate random
Unusual chromosome IDs
Recombination map
Allele frequencies
Phenotypes
Covariates
Clusters of samples
Variant sets
Binary distance matrix

PLINK 1.90 beta

This is a comprehensive update to Shaun Purcell's PLINK command-line program, developed by Christopher Chang with support from the NIH-NIDDK's Laboratory of Biological Modeling, the Purcell Lab, and others. (What's new?) (Credits.) (Methods paper.) (Usage questions should be sent to the **plink2-users Google group**, not Christopher's email.)

Binary downloads

	Build		
Operating system[1]	Stable (beta 7.2, 11 Dec 2023)	Development (11 Dec 2023)	Old[2] (v1.07)
Linux 64-bit	download	download	download
Linux 32-bit	download	download	download
macOS (64-bit)			
Windows 64-bit			
Windows 32-bit			

1: Solaris is no longer explicitly
2: These are just mirrors of the b

Source code, compilatio

The following docume

PLINK 1.9 | PLINK 2.0 home | plink2-users | Error messages | File formats | PLINK 2.0 index

Introduction, downloads
D: 18 Mar 2024
Recent version history
What's new?
Coming next
[Jump to search box]
General usage
Getting started
Flag usage summaries
Column set descriptors
Citation instructions
Standard data input
PLINK 1 binary (.bed)
PLINK 2 binary (.pgen)
Autoconversion behavior
VCF/BCF (.vcf[.gz], .bcf)
Oxford genotype (.bgen)
Oxford haplotype (.haps)
PLINK 1 text (.ped, .tped)
PLINK 1 dosage
Sample ID conversion
Dosage import settings
Generate random
Unusual chromosome IDs
Allele frequencies
Phenotypes
Covariates
'Cluster' import

PLINK 2.00 alpha

PLINK 2.0 alpha was developed by **Christopher Chang**, with support from **GRAIL, LLC** and **Human Longevity, Inc.**, and substantial input from Stanford's Department of Biomedical Data Science. (More detailed credits.) (Usage questions should be sent to the **plink2-users Google group**, not Christopher's email.)

Binary downloads

	Build	
Operating system	Development (18 Mar)	Alpha 5.10 final (5 Jan)
Linux AVX2 Intel[1]	**download**	download
Linux AVX2 AMD[1]	**download**	download
Linux 64-bit Intel[1]	download	download
Linux 32-bit	download	download
macOS M1	**download**	download
macOS AVX2	**download**	download
macOS 64-bit	download	download
Windows AVX2	**download**	download
Windows 64-bit	download	download
Windows 32-bit	download	download

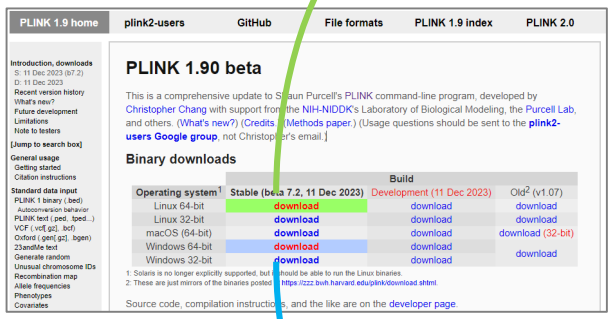
population-genet
the final first-ge
witnessed the intr
ical approaches,
well as wide deple
In response, we
hensive performa
data indicate that

*Correspondence: chrc
[1]Complete Genomics,
[2]BGI Cognitive Genomi
Yantian District, 518083
Full list of author inform

Software – PLINK download & initialization

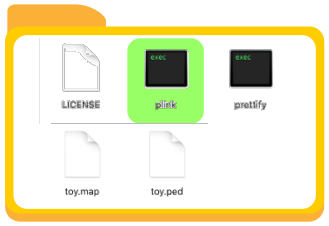
Download → **Decomposition** → **Open CLI Command-Line Interface** → **Execution**

plink_linux_x86_64_20231211.zip

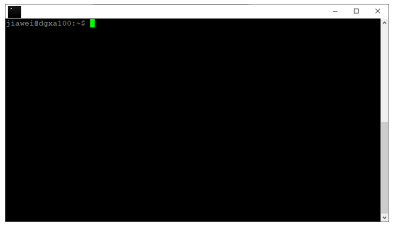


plink_win64_20231211.zip

plink_linux_x86_64_20231211

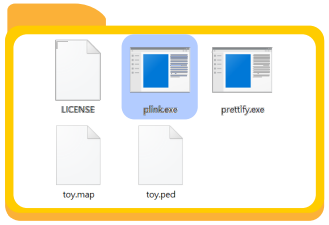


open **Terminal**

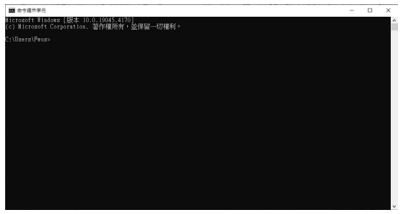


```
$ cp plink_linux_x86_64_20231211/plink usr/local/bin
$ plink
```

plink_win64_20231211



open **Command Prompt (cmd.exe)**:
Windows + X ⇨ Command Prompt (C)
命令提示字元 (C)



```
> cd plink_win64_20231211
> plink
```

Data format – *.ped & *.map

*.ped

Family ID	Individual ID	Paternal ID	Maternal ID	Sex	Phenotype	marker1		marker2		marker3		
						allele1	allele2	allele1	allele2	allele1	allele2	
FAM001	ind1	0	0	1	2	G	G	A	A	C	G	…
FAM001	ind2	0	0	1	2	G	G	A	G	G	G	
FAM001	ind3	0	0	2	1	G	T	A	G	C	G	
⋮												

For case/control study, we let

❶ Family IDs are all the same or equal to Individual IDs

❷ Paternal IDs and Maternal IDs are 0s

Sex: 0 = unknown, 1 = male, 2 = female

Phenotype: -9/0 = missing, 1 = unaffected, 2 = affected

❶ **--1** if using 0/1 to represent for unaffected/affected

❷ **--missing-phenotype -99** to reset the representation for missing phenotypes

Genotype: 0 = missing, 1 = A, 2 = C, 3 = G, 4 = T, non-zero integers or any characters

❶ all markers are ***biallelic***

❷ two alleles of missing genotype are 0s

❸ **--missing-genotype N** to reset the representation for missing genotypes

*.map

Chromosome	Marker ID	Genetic distance	Physical position
1	marker1	0	565433
1	marker2	0	752566
1	marker3	0	753541
⋮			

Chromosome: 1-22, 23 = X, 24 = Y, 25 = XY(PAR), 26 = MT

Physical position: base pair (bp)

❶ **--allow-extra-chr** allow for unrecognized chromosome codes

❷ be careful about the genome build

ID: any characters, take Affymetrix Axiom array for example,

❶ Probe Set ID (Ax-***): unique id for probe sequence

❷ Affy SNP ID (Affy-***): unique id for CHR, POS, REF, and ALT

❸ dbSNP RS ID (rs***): unique id for a genome build

Genetic distance: centimorgan (cM)

Data format – bfiles

Map file (*.map)

Physical position	565433	752566	753541	…
Genetic distance	0	0	0	
Marker ID	marker1	marker2	marker3	
Chr	1	1	1	

Extended map file (*.bim)

Major allele	G	A	G	
Minor allele	T	G	C	
Physical position	565433	752566	753541	
Genetic distance	0	0	0	
Marker ID	marker1	marker2	marker3	
Chr	1	1	1	…

--make-bed

Family ID	Individual ID	Paternal ID	Maternal ID	Sex	Phenotype	marker1		marker2		marker3		
FAM001	ind1	0	0	1	2	G	G	A	A	C	G	…
FAM001	ind2	0	0	1	2	G	G	A	G	G	G	
FAM001	ind3	0	0	2	1	G	T	A	G	C	G	
⋮						⋮						

Pedigree file (*.ped)

Family ID	Individual ID	Paternal ID	Maternal ID	Sex	Phenotype
FAM001	ind1	0	0	1	2
FAM001	ind2	0	0	1	2
FAM001	ind3	0	0	2	1
⋮					

Family information file (*.fam)

marker1	marker2	marker3	
11	11	10	…
11	10	01	
10	10	11	
⋮			

Binary-coded genotype file (*.bed)

2-bit genotype codes

00	Homozygous for first allele in .bim file
01	Missing genotype
10	Heterozygous
11	Homozygous for second allele in .bim file

Getting started

$ plink **--bfile** [input_filepath] **--[flags]** [options] **--out** [output_filepath]

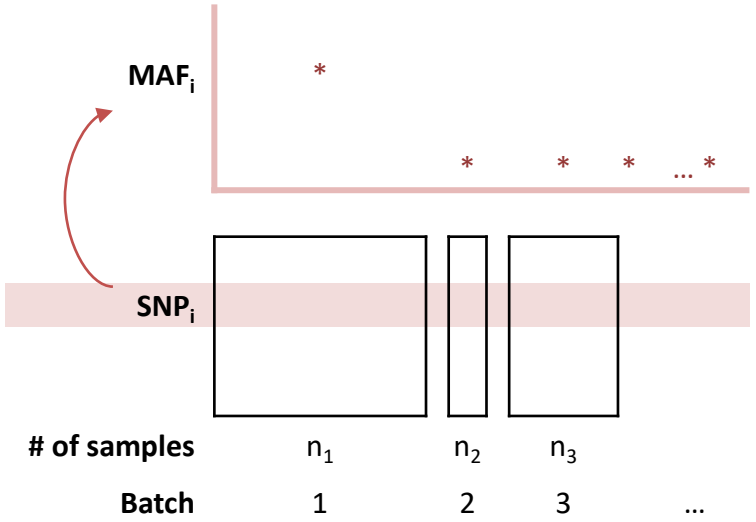
```
$ plink --bfile path/of/your/file --chr 1-22 --make-bed --out dat_auto
```

extract the autosomes (chr 1 to 22) and generate a new file named dat_auto

Sample QC – Batch SNP removal

Why batch SNP removal?

- May cause suspicious association results



How to do?

- Compare AFs of SNPs to ones in public database (e.g., 1000 genomes project) to identify the SNPs with weird AFs
- Instead of excluding the SNP, marking the genotype calls of samples from problematic batches as missing for this specific SNP

Sample QC – Batch SNP removal

```
$ plink --bfile path/of/your/file \
  --zero-cluster batchSNPs.zero  \
  --within dat.clst \
  --make-bed --out dat_wg
```

dat.clust

FID	IID	Batch
FAM001	ind1	1
FAM001	ind2	1
FAM001	ind3	2
⋮		

The batch information can be requested from typing center

batchSNPs.zero

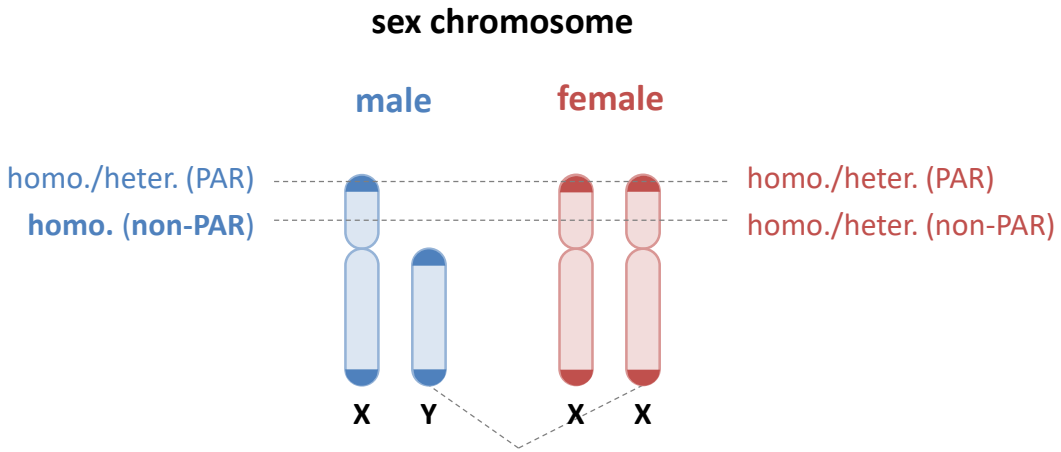
SNP	Batch
marker1	1
marker1	3
marker3	3
⋮	

It may be a challenge for non-programmers to find out batch SNPs

Sample QC – Sex check

Why sex check?

- May have chromosome anomaly or structural variation
- May be a covariate in the subsequent analysis



PAR (Pseudoautosomal region)
homologous sequences of nucleotides on the X and Y chromosomes

How to do?

- Using either **homozygosity rate** or **inbreeding coefficient** of X chromosome to check for the gender

	Male	Female
Homozygosity rate	≥ 0.9	< 0.9
Inbreeding coefficient (F)	> 0.8	< 0.2

- If EHR data is accessed, directly comparing EHR gender to genetic gender

Sample QC – Sex check

```
$ plink --bfile dat_wg --check-sex --out dat_wg
$ grep "PROBLEM" dat_wg.sexcheck > rmInd_sex.txt
$ plink --bfile dat_auto --remove rmInd_sex.txt --make-bed --out dat_auto_1
```

dat_wg.**sexcheck**

FID	IID	PEDSEX	SNPSEX	STATUS	F
FAM001	ind1	1	1	OK	0.9588
FAM001	ind2	1	1	OK	0.9616
FAM001	ind3	1	1	OK	0.9588
⋮					
FAM001	ind17	2	1	PROBLEM	0.9539
⋮					
FAM001	ind28	1	2	PROBLEM	-0.05586
⋮					

→

rmInd_sex.txt

FAM001	ind17
FAM001	ind28

Remove samples with inconsistent genders (**STATUS = PROBLEM**)

FID	IID	PEDSEX	SNPSEX	STATUS	F
		1	1	OK	> 0.8
		2	2	OK	< 0.2
		1	2	PROBLEM	< 0.2
		2	1	PROBLEM	> 0.8
		1	0	PROBLEM	[0.2 , 0.8]
		2	0	PROBLEM	[0.2 , 0.8]

Sample QC – Genotype call-rate & heterozygosity rate check

Why genotype call-rate check?

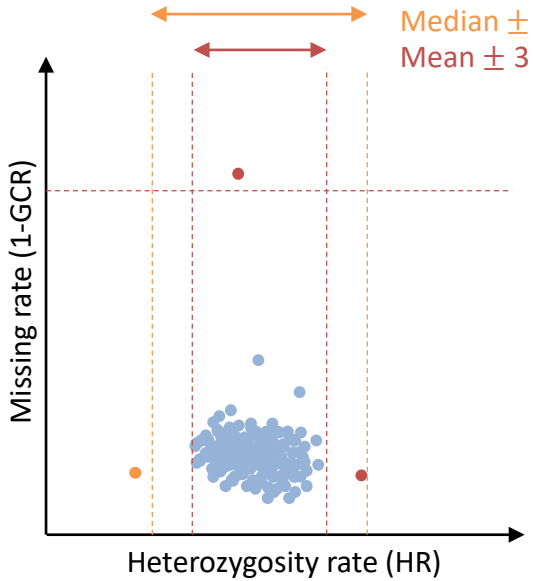
- Low DNA quality or concentration often have below-average call rates & genotype accuracy

Why heterozygosity rate check?

- An excessive or reduced proportion of heterozygote genotypes, which may be indicative of DNA sample contamination or inbreeding, respectively

How to do?

- Calculate genotype call-rate (GCR) and heterozygosity rate (HR) for **autosomes**,



N = # of markers

N_i = # of nonmissing genotypes for individual i

O_i = # of homozygous genotypes for individual i

$GCR_i = \frac{N_i}{N}$

$HR_i = \frac{N_i - O_i}{N_i}$

Sample QC – Genotype call-rate & heterozygosity rate check

```
$ plink --bfile dat_auto_1 --missing --het --out dat_auto_1
$ Plink –bfile dat_auto_1 --remove rmInd_missing_het.txt --make-bed --out dat_auto_2
```

dat_auto_1.**imiss**

FID	IID	MISS_PHENO	N_MISS	N_GENO	F_MISS
FAM001	ind1	Y			0.001578
FAM001	ind2	Y			0.00181
FAM001	ind3	Y			0.001375
⋮					

Remove samples with CR = 1 − F_MISS smaller than a threshold, say 0.95

rmInd_missing_het.txt

FAM001	ind?	CR
FAM001	ind?	HET

dat_auto_1.**het**

FID	IID	O(HOM)	E(HOM)	N(NM)	F
FAM001	ind1	426964		624027	
FAM001	ind2	428153		623882	
FAM001	ind3	425156		624154	
⋮					

Remove samples with HR = $\frac{O(HOM)}{N(NM)}$ out of mean(HR) $\pm$ 3 sd(HR)

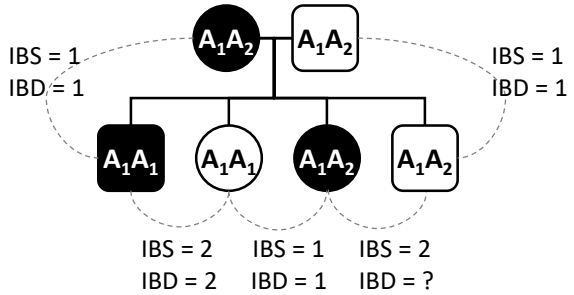
Sample QC – Cryptic relatedness check

Why cryptic relatedness check?

- An association study will be interfered by the relatedness, e.g., violation of assumption for linear/logistic models, bias AF estimation, etc.
- A family-based study should take the relatedness into account

How to do?

- Usually, using **independent** SNPs (pair correlation $r^2 < 0.2$) to calculate the relatedness of individuals by either **proportion IBD** or **kinship coefficient** to check for the relatedness



	Proportion IBD	Kinship coeff.
duplicate/MZ twin	1	> 0.354
1st degree	0.5	[0.177, 0.354]
2nd degree	0.25	[0.0884, 0.177]
halfway of 2nd & 3rd degrees	> 0.1875	
3rd degree	0.125	[0.0442, 0.0884]

Sample QC – Cryptic relatedness check

```
$ plink --bfile dat_auto_2 --indep-pairwise 50 5 0.2 --out dat_auto_2
$ plink --bfile dat_auto_2 --extract dat_auto_2.prune.in --king-cutoff 0.0442 --out dat_auto_2
$ plink --bfile dat_auto_2 --remove dat_auto_2.king.cutoff.out.id --make-bed --out dat_auto_3
```

or

```
$ plink --bfile dat_auto_2 --indep-pairwise 50 5 0.2 --out dat_auto_2
$ plink --bfile dat_auto_2 --extract dat_auto_2.prune.in --genome --out dat_auto_2
$ plink --bfile dat_auto_2 --remove rmInd_relate.txt --make-bed --out dat_auto_3
```

dat_auto_2.**genome**

FID1	IID1	FID2	IID2	RT	EZ	Z0	Z1	Z2	PI_HAT	PHE	DST	PPC	RATIO
									0				
									0				
									0.0135				
									0.589				
									0.5052				
⋮													

rmInd_relate.txt

FAM001	ind?
FAM001	ind?

Remove samples with lower CR in a pair of samples if **PI_HAT** > 0.1875

It may be a challenge for non-programmers to find the optimal sample set

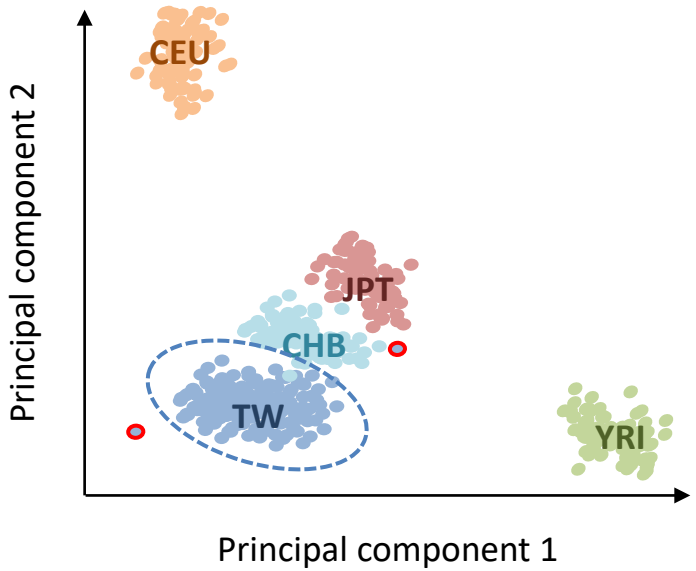
Sample QC – Divergent ancestry check

Why divergent ancestry check?

- Confounding to case/control groups, i.e. identified markers may not associated with disease but differently distributed in ancestries (Hamer, D. , & Sirota, L. (2000))

How to do?

- Merge study genotypes to HapMap3 or 1000 genomes data
- Exclude ambiguous SNPs (A/T or C/G polymorphic)
- Prune highly correlated SNPs
- PCA
- Exclude individuals out of 99.9% confidence band of data



Sample QC – Divergent ancestry check

Download 1000 genomes phase 3 data https://www.cog-genomics.org/plink/2.0/resources#phase3_1kg

Download HapMap 3 data ftp://ftp.ncbi.nlm.nih.gov/hapmap/genotypes/2010-05_phaseIII/plink_format/

```
$ awk '{print $1,$4,$4,$2}' dat_auto.bim > lst_var.txt
$ plink2 --zst-decompress all_hg38.pgen.zst > all_hg38.pgen
$ plink2 --pfile all_hg38 vzs \
    --allow-extra-chr \
    --extract bed1 lst_var.txt \          extract SNPs in your array
    --snps-only \
    --max-alleles 2 \
    --set-all-var-ids @:# \               represent SNP id as chr:pos for the further merge
    --rm-dup exclude-all \
    --make-bed --out all_hg38_extract
$ sed 's/#IID/IID/g;s/Population/population/g' all_hg38.psam > relationships_w_pops.txt
```

Sample QC – Divergent ancestry check

```
$ plink2 --bfile dat_auto_3 --set-all-var-ids @:# --rm-dup exclude-all --make-bed --out dat_auto_3_
$ plink --bfile dat_auto_3_ --bmerge all_hg38_extract -make-bed --out tmp
```

```
Error: ??? variants with 3+ alleles present.
* If you believe this is due to strand inconsistency, try --flip with tmp-merge.missnp.
 (Warning: if this seems to work, strand errors involving SNPs with A/T or C/G alleles probably remain in your data.
  If LD between nearby SNPs is high, --flip-scan should detect them.)
* If you are dealing with genuine multiallelic variants, we recommend exporting
  that subset of the data to VCF (via e.g. '--recode vcf'), merging with another tool/script,
  and then importing the result; PLINK is not yet suited to handling them.
```

```
$ plink --bfile all_hg38_extract --flip tmp-merge.missnp --make-bed --out all_hg38_extract_flip
$ plink --bfile dat_auto_3_1 --bmerge all_hg38_extract_flip --make-bed --out dat_auto_3_1
```

or

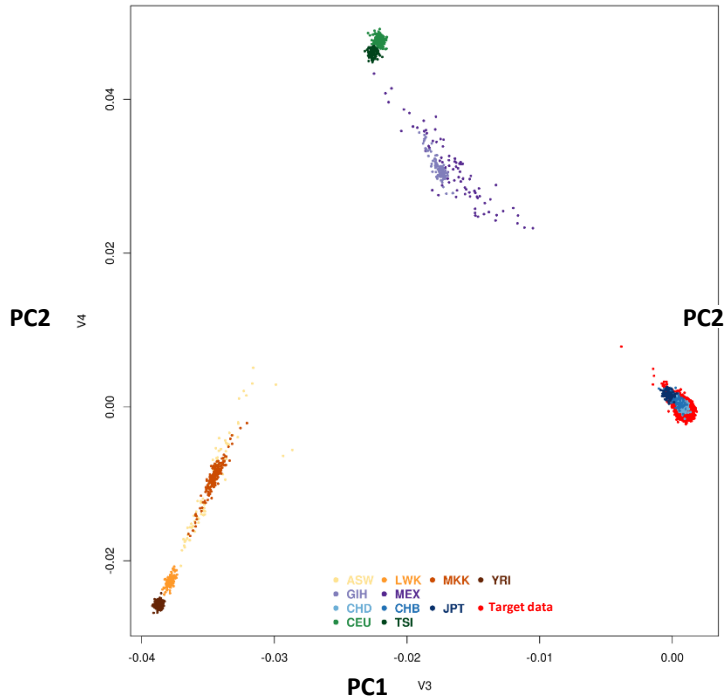
```
$ plink --bfile all_hg38_extract --exclude tmp-merge.missnp --make-bed --out all_hg38_extract_ex
$ plink --bfile dat_auto_3_1 --bmerge all_hg38_extract_ex --make-bed --out dat_auto_3_1
```

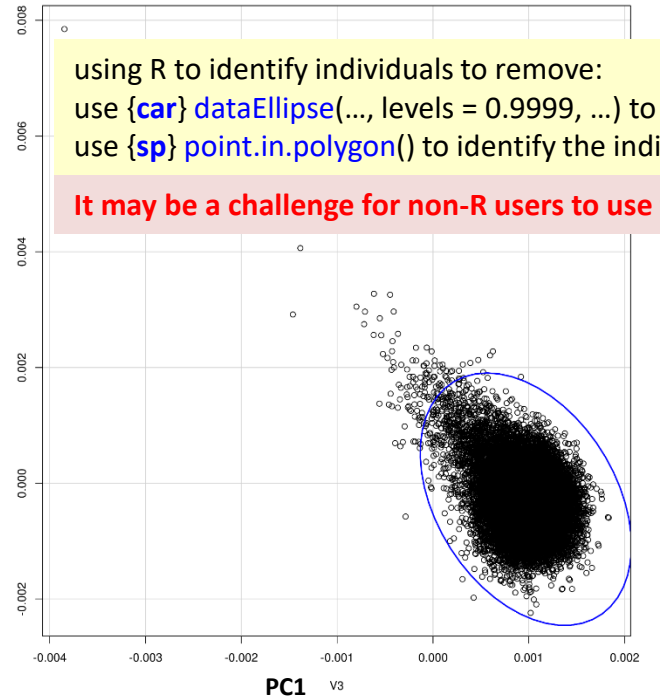
Sample QC – Divergent ancestry check

```
$ plink2 --bfile dat_auto_3_1 --threads 16 --geno 0.05 --maf 0.01 --pca approx --out dat_auto_3_1
$ plink --bfile dat_auto_3 --remove rmInd_divAncestry.txt --make-bed --out dat_auto_4
```

dat_auto_3_1.**eigenvec**

FID	IID	PC1	PC2	…	PC10





using R to identify individuals to remove:
use {**car**} dataEllipse(…, levels = 0.9999, …) to plot the ellipse bands
use {**sp**} point.in.polygon() to identify the individuals out of the bands

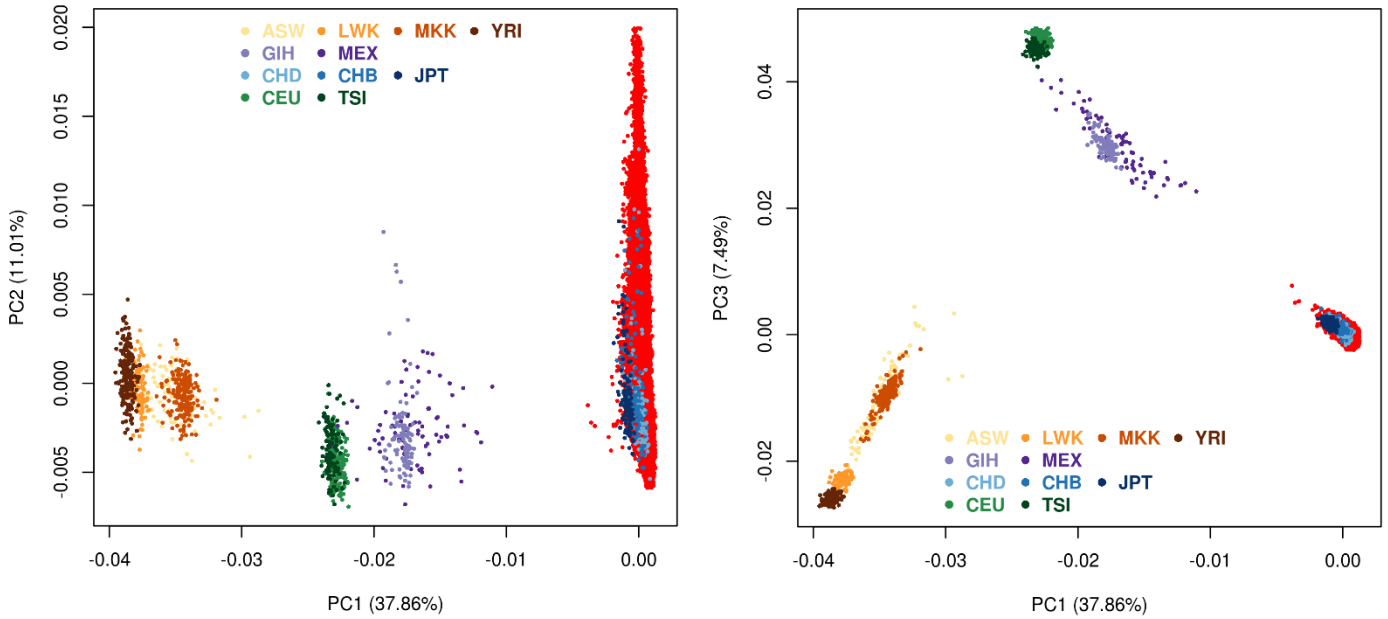
It may be a challenge for non-R users to use these functions

Sample QC – Divergent ancestry check

```
$ plink2 --bfile dat_auto_3_1 --threads 16 --geno 0.05 --maf 0.01 --pca approx --out dat_auto_3_1
$ plink --bfile dat_auto_3 --remove rmInd_divAncestry.txt --make-bed --out dat_auto_4
```

dat_auto_3_1.**eigenvec**

FID	IID	PC1	PC2	PC3	…	PC10



When sample size of target data is much larger than that of ancestry data, the first PCs may be dominant by the variations of target data

Sample QC – Divergent ancestry check (projection)

Download 1000 genomes phase 3 data https://www.cog-genomics.org/plink/2.0/resources#phase3_1kg

Download HapMap 3 data ftp://ftp.ncbi.nlm.nih.gov/hapmap/genotypes/2010-05_phaseIII/plink_format/

```
$ awk '{print $1,$4,$4,$2}' dat_auto.bim > lst_var.txt
$ plink2 --zst-decompress all_hg38.pgen.zst > all_hg38.pgen
$ plink2 --pfile all_hg38 vzs --allow-extra-chr --extract bed1 lst_var.txt --snps-only --max-alleles 2 --set-all-var-ids @:# --rm-dup exclude-all --make-bed --out all_hg38_extract
$ sed 's/#IID/IID/g;s/Population/population/g' all_hg38.psam > relationships_w_pops.txt


$ plink2 --bfile all_hg38_extract \
    --maf 0.01 \
    --freq counts \
    --pca biallelic-var-wts \
    --out all_hg38_extract
```

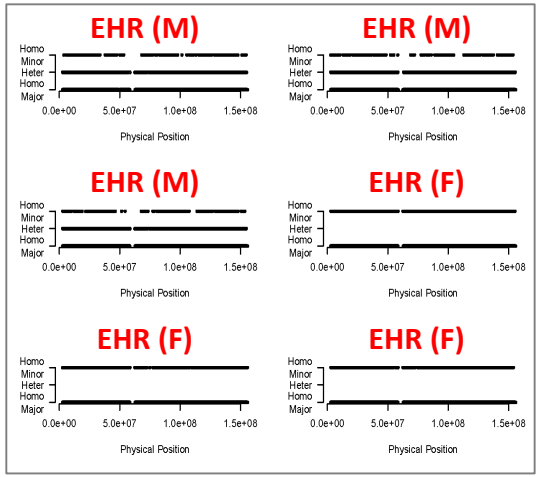
Using this method helps avoid strong influence of the target data on the first PCs

Sample QC – Divergent ancestry check (projection)

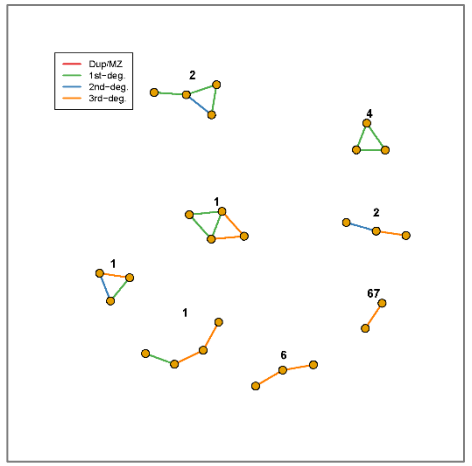
```
$ plink2 --bfile all_hg38_extract \
  --maf 0.01 \
  --freq counts \
  --pca biallelic-var-wts \
  --out all_hg38_extract

$ plink2 --bfile dat_auto_3_1 \
  --read-freq all_hg38_extract.acounts \
  --score all_hg38_extract.eigenvec.var 2 4 header read variance-standardize no-mean-imputation \
  --score-col-nums 5-14
  --out dat_auto_3_1
```

Sample QC – Figures

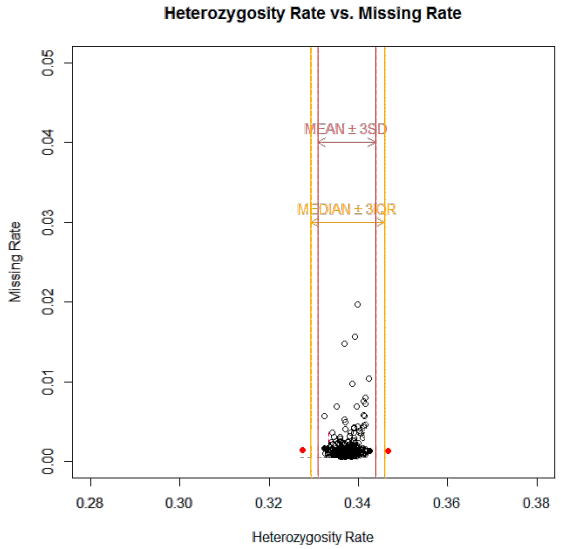
Sex check



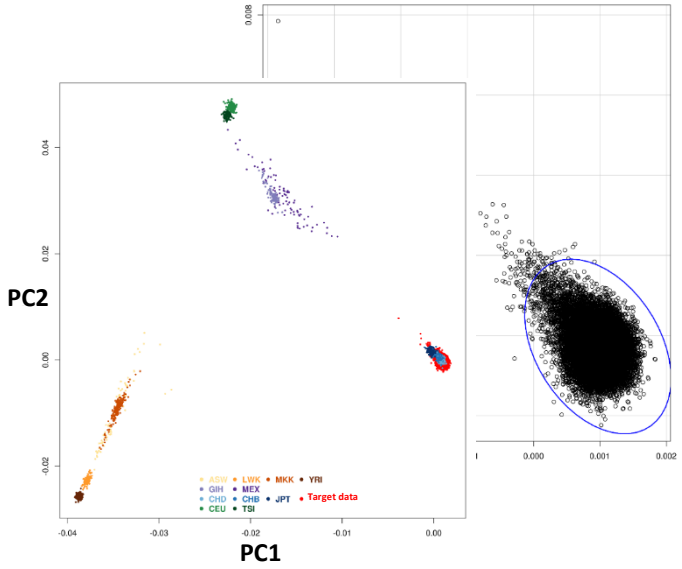
Cryptic relatedness check



Genotype call-rate & heterozygosity rate check



Divergent ancestry check



Marker QC – Genotype call-rate check

- Exclude markers with lower call rate, say 95% (sometimes, 99%)

```
$ plink --bfile dat_auto_4 --geno 0.05 --make-bed --out dat_auto_5
```

- For case/control study, we can further
  exclude markers with a large difference of call rates between cases & controls

```
$ plink --bfile dat_auto_4 --test-missing --out dat_auto_5
```

  exclude markers with a call rate less than 95% in either cases or controls

Marker QC – Minor allele frequency check

- Exclude markers with lower MAF, say 0.01 (sometimes, 0.05)

```
$ plink --bfile dat_auto_5 --maf 0.01 --make-bed --out dat_auto_6
```

Individual-based AF

Genotype call	AF_A	AF_B	MAF
AA	1	0	0
AB	0.5	0.5	0.5
BB	0	1	0

Population-based AF

		Allele 2	
		A	**B**
Allele 1	**A**	n_{AA}	n_{AB}
	B	n_{BA}	n_{BB}

N

$$AF_A = \frac{n_{AA}\times 2 + n_{AB}\times 1 + n_{BA}\times 1 + n_{BB}\times 0}{2N}$$

$$AF_B = \frac{n_{AA}\times 0 + n_{AB}\times 1 + n_{BA}\times 1 + n_{BB}\times 2}{2N}$$

$$MAF = \min(\, AF_A, AF_B \,)$$

We may also interested in what the **minor/major allele** is

Marker QC – Hardy-Weinberg equilibrium check

- Exclude markers with a p-value of Hardy-Weinberg equilibrium (HWE) test less than a threshold (e.g., Bonferroni's level)

```
$ plink --bfile dat_auto_6 --hwe 5e-08 --make-bed --out dat_auto_qc
```

  - In case/control study, the HWE test is applied for control individuals only
  - In quantitative-trait study, the HWE test is applied for all individuals

		Allele 2		
		A	**B**	
Allele 1	**A**	$p_{AA} = p_A^2$	$p_{AB} = p_A p_B$	p_A
	B	$p_{BA} = p_B p_A$	$p_{BB} = p_B^2$	p_B
		p_A	p_B	1

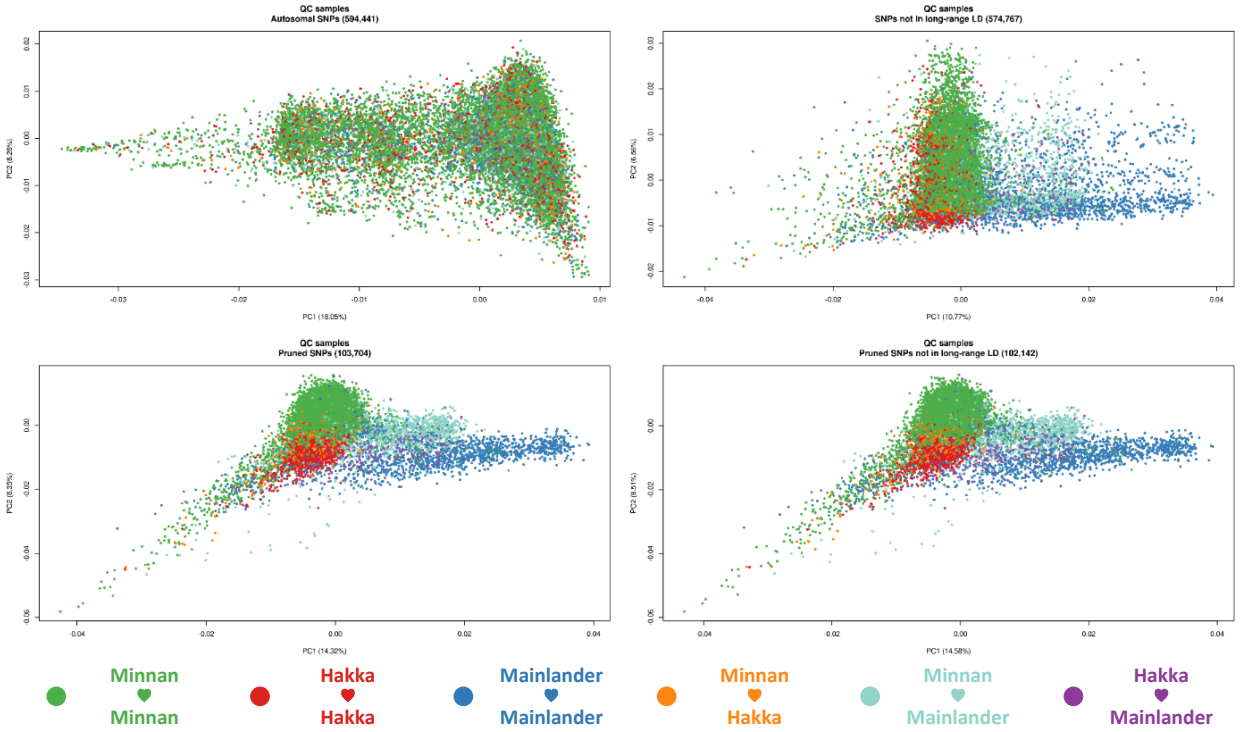
Hardy-Weinberg principle

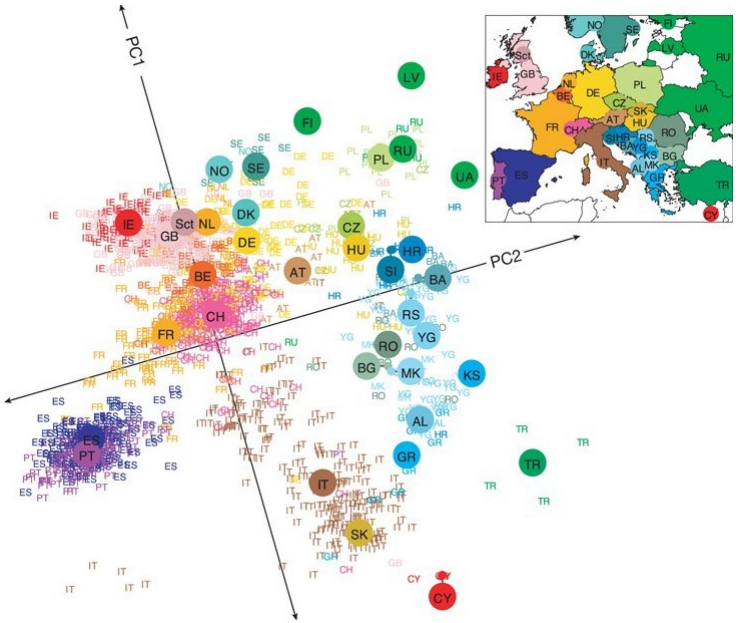
The genetic variation (allele/genotype frequencies) in a population will remain **constant** from one generation to the next in the **absence of disturbing factors** (e.g., selection, mutation and migration)

Association test – Subpopulation structure

Why subpopulation structure?

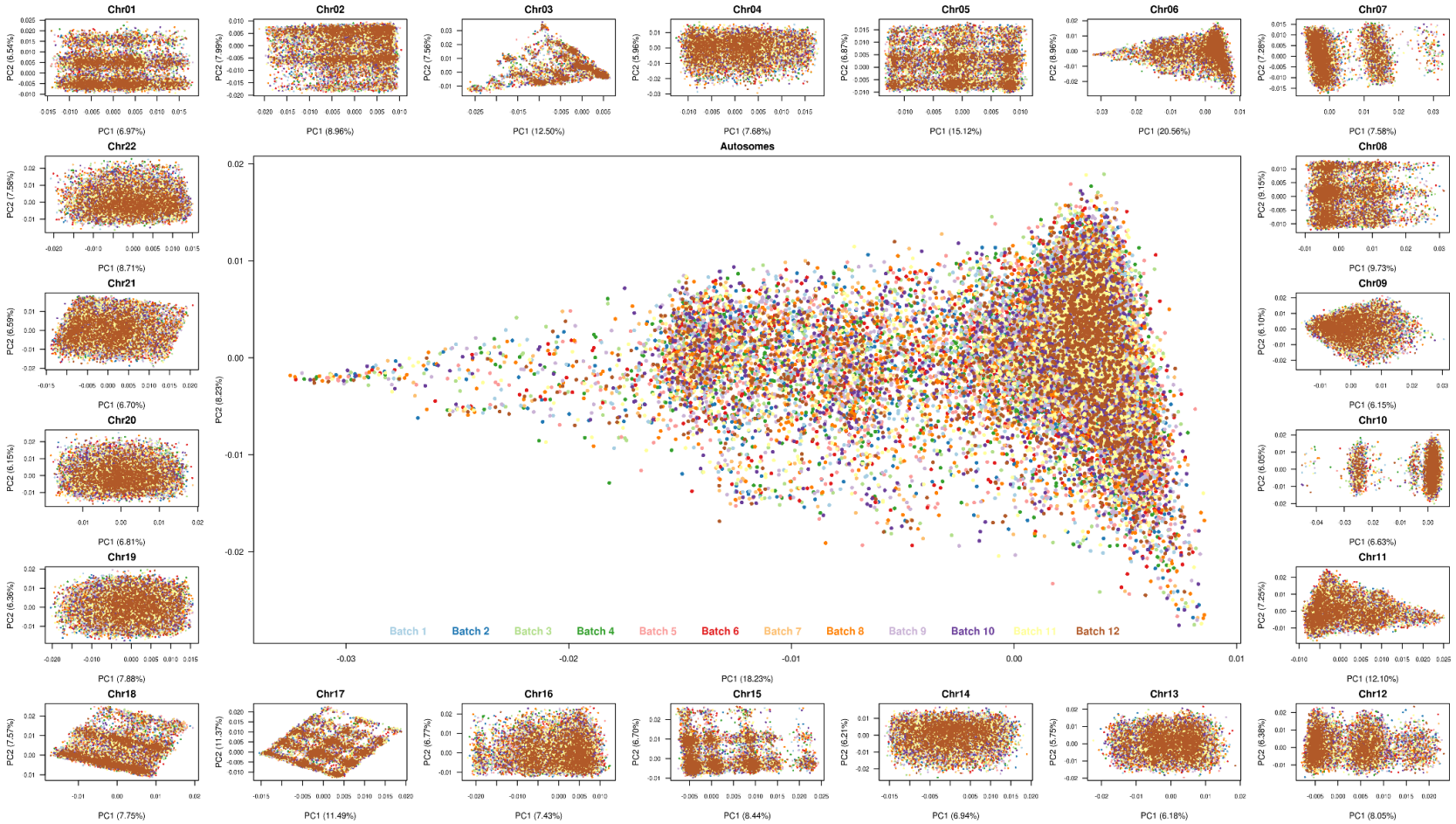
- Confounding to case/control groups, i.e. identified markers may not associated with disease but differently distributed in ancestries

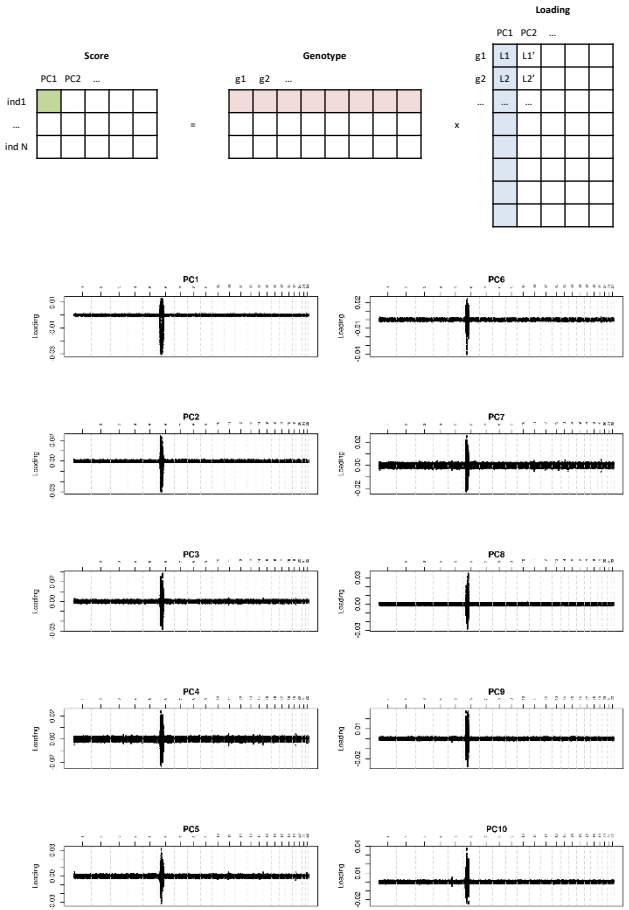




Novembre, J., Johnson, T., Bryc, K. *et al* (2008)

Association test – Subpopulation structure (score & loading)





Association test – Subpopulation structure

```
$ plink2 --bfile dat_auto_qc --pca --out dat_auto_qc
```

- pruning by counts of SNPs

```
$ plink --bfile dat_auto_qc --indep-pairwise 5000 500 0.2 --out dat_auto_qc_cnt
$ plink2 –bfile dat_auto_qc –extract dat_auto_qc_cnt.prune.in --pca biallelic-var-wts --out dat_auto_qc_pruningCnt
```

- pruning by distance

```
$ plink --bfile dat_auto_qc --indep-pairwise 5000kb 1 0.2 --out dat_auto_qc_kb
$ plink2 –bfile dat_auto_qc –extract dat_auto_qc_kb.prune.in --pca biallelic-var-wts --out dat_auto_qc_pruningKb
```

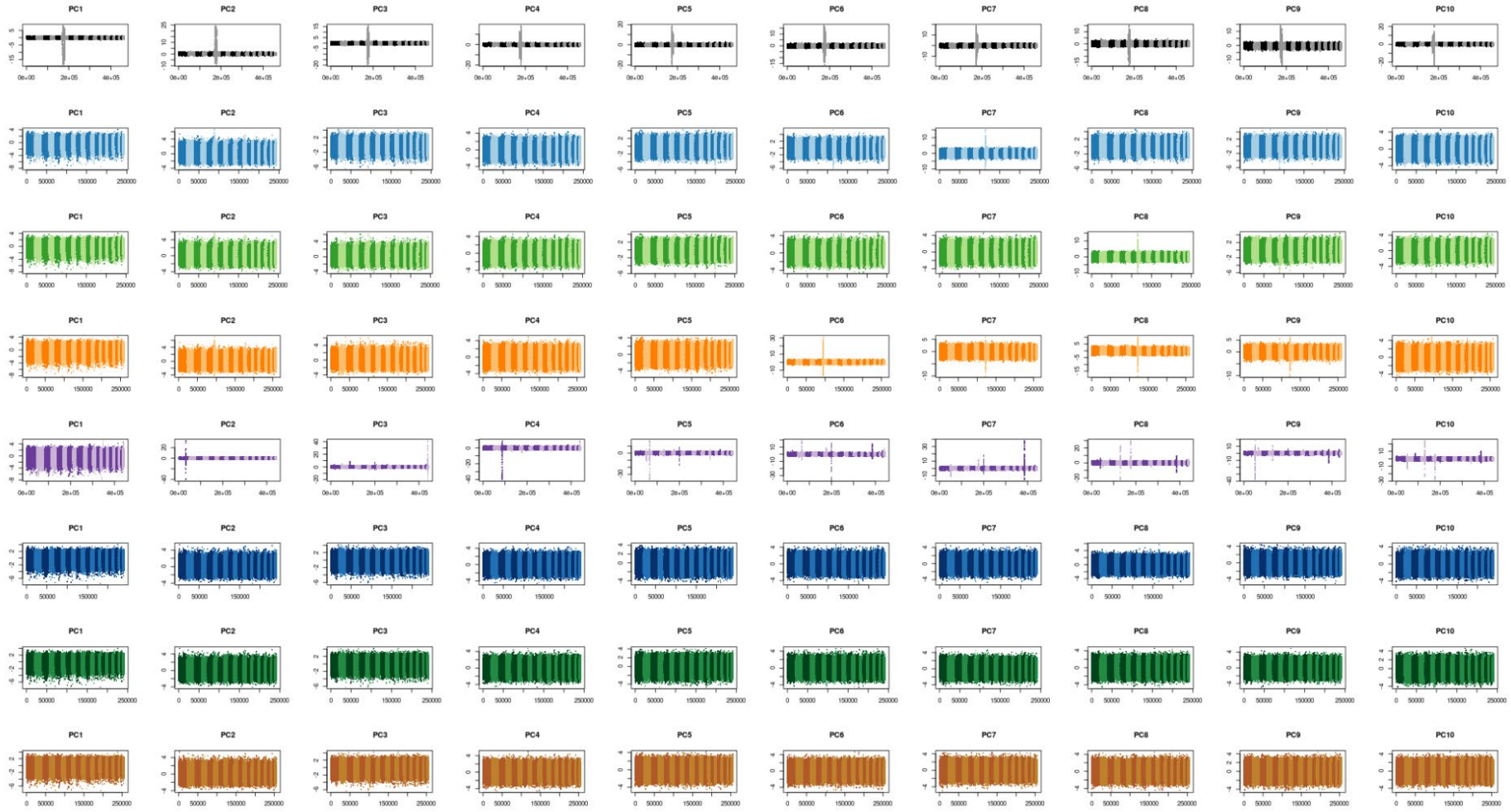
- clumping according to MAF

```
$ plink2 --bfile dat_auto_qc --freq --out dat_auto_qc
$ awk 'NR==1{$(NF+1)="pseudoMAF"} NR>1{$(NF+1)=($5>0.5?$5:(1-$5))}1' dat_auto_qc.afreq > dat_auto_qc.afreq_
$ plink --bfile dat_auto_qc --clump dat_auto_qc.afreq_ --clump-snp-field ID --clump-field pseudoMAF --out dat_auto_qc
$ awk '{print $3}' dat_auto_qc.clumped > dat_auto_qc.clumped.in
$ plink2 --bfile dat_auto_qc --extract dat_auto_qc.clumped.in --pca biallelic-var-wts --out dat_auto_qc_clumping
```

- remove long-range LD (LRLD) region

```
$ plink2 --bfile dat_auto_qc --exclude range LRLD_GRCh38.txt --pca biallelic-var-wts --out dat_auto_qc_rmLRLD
```

Association test – Subpopulation structure (loading)



Association test

Disease

Strict disease group definitions are necessary, but sample size may still be the ultimate challenge in GWAS

Firth logistic regression

It can deal with the issue of rare events or complete separation in traditional logistic regression

Normal assumption

Inverse normal transformation (INT):

$$\mathrm{INT}(x) = \Phi^{-1}\left(\frac{rank(x)-c}{n-2c+1}\right)\begin{cases} c = \frac{3}{8}\ (Blom, 1958) \\ c = \frac{1}{3}\ (Tukey, 1962) \\ c = \frac{1}{2}\ (Bliss, 1967) \end{cases}$$

Is it applied before or after covariate adjustment?

Relatedness

In QC, although we removed individuals having equal to or higher than 2nd/3rd -degree cryptic relations with others, we still can run LMM

	Binary trait	Quantitative trait
Indep. sample	**Firth logistic regression** with covariates and population structure adjustments	**Linear regression** with/without covariates and population structure for transformed quantitative traits
Rel. sample	**Logistic mixed model** with covariates and population structure adjustments	**linear mixed model** with/without covariates and population structure for transformed quantitative traits

Association test – Independent sample + binary trait

```
$ plink2 --bfile dat_auto_qc --pheno pheCov.txt --pheno-name bt_1 --1 \
  --covar pheCov.txt --covar-name age,sex,pc1-pc10 --covar-variance-standardize \
  --glm hide-covar cols=chrom,pos,ref,alt,a1freq,a1freqcc,gcountcc,nobs,orbeta,se,tz,p,firth,err
  --out dat_auto_qc
```

pheCov.txt

		phenotypes				covariates		subpopulation structure		
FID	IID	bt_1	bt_2	qt_1	qt_2	age	sex	pc1	pc2	…
FAM001	ind1									
FAM001	Ind2									
FAM001	ind3									
⋮										

data_auto_qc.bt_1.**glm.logistic.hybrid**

#CHROM	POS	ID	REF	ALT	A1	CASE_NON_A1_CT	CASE_HET_A1_CT	CASE_HOM_A1_CT	CTRL_NON_A1_CT	CTRL_HET_A1_CT	CTRL_HOM_A1_CT	A1_FREQ	A1_CASE_FREQ	A1_CTRL_FREQ	FIRTH?	OBS_CT	OR	LOG(OR)_SE	Z_STAT	P	ERRCODE

Association test – Independent sample + quantitative trait

```
$ plink2 --bfile dat_auto_qc --pheno pheCov.txt --pheno-name qt_1 \
  --covar pheCov.txt --covar-name age,sex,pc1-pc10 --covar-variance-standardize \
  --glm hide-covar cols=chrom,pos,ref,alt,a1freq,a1freqcc,gcountcc,nobs,orbeta,se,tz,p,firth,err
  --out dat_auto_qc
```

pheCov.txt

		phenotypes				covariates		subpopulation structure		
FID	IID	bt_1	bt_2	qt_1	qt_2	age	sex	pc1	pc2	…
FAM001	ind1									
FAM001	Ind2									
FAM001	ind3									
⋮										

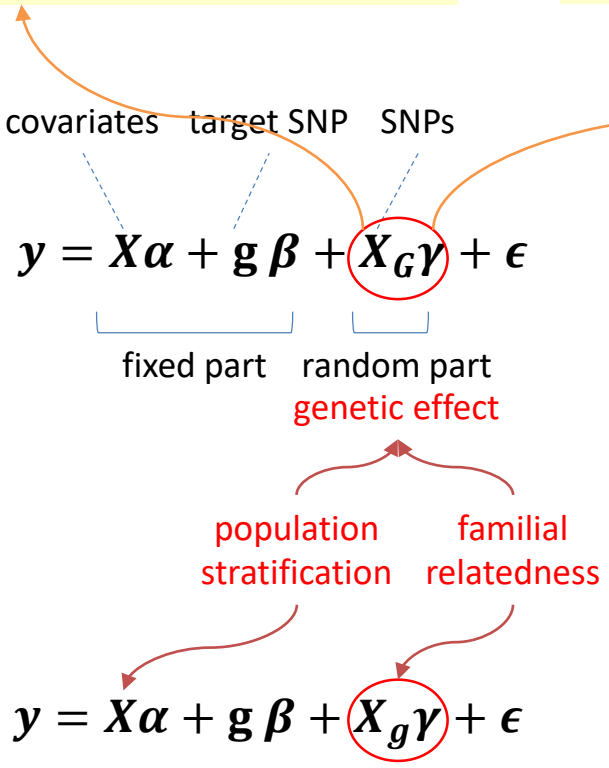
data_auto_qc.qt_1.glm.linear

#CHROM	POS	ID	REF	ALT	A1	A1_FREQ	OBS_CT	BETA	SE	T_STAT	P	ERRCODE

Association test – Relative sample

Proximal contamination
Inclusion of the target SNP or nearby SNPs (proximal SNPs) in the GRM interfere with the analysis of the target SNP causes both as a fixed effect tested for association and as a random effect as part of the GRM

Infinitesimal model (polygenic model)
a phenotype is influenced by an infinitely large number of genes, each of which makes an infinitely small (infinitesimal) contribution to the phenotype



Genetic effect attribute to all SNPs (X_G = all SNPs) that is computationally expensive, so typically use a subset of informative SNPs

Use a subset of SNPs to construct GRM can lead to insufficient correction for population stratification

Association test – Relative sample

	BOLT-LMM	SAIGE	regenie
Paper	Loh P.R., et al. (2015) Nat Genet.	Zhou W., et al. (2018) Nat Genet.	Mbatchou, J., et al. (2021) Nat Genet.
System	Linux, Windows	Linux, Windows	Linux
Phenotype type	Quantitative	Binary and quantitative	Binary and quantitative
Step 1	Estimate variance parameters Calculate BOLT-LMM-inf	Fit null model: $\boldsymbol{y} = \boldsymbol{X\alpha} + \boldsymbol{X_G\gamma} + \boldsymbol{\epsilon}$ Estimate variance ratio	Fit null model: $\boldsymbol{y} = \boldsymbol{X\alpha} + \boldsymbol{X_G\gamma} + \boldsymbol{\epsilon}$ Ridge regression, blockwise, CV scheme
Step 2	Estimate Gaussian mixture-model parameters Calculate BOLT-LMM (Gaussian mixture-model)	For binary traits, SPA accounts for case/control imbalance Firth bias-reduced logistic regression	For binary traits, SPA accounts for case/control imbalance Firth logistic regression
Model assumption	Infinitesimal Non-infinitesimal (Gaussian mixture-model)	Infinitesimal	Infinitesimal
Avoid proximal contamination	LOCO (auto)	LOCO (optional)	LOCO (auto)
Requirement	Genetic map (download) LD score (~~download~~)		
Note	* > 5,000 samples * balanced binary traits	* imbalance binary traits	* imbalance binary traits

Association test – Relative sample + quantitative trait

```
$ bolt --bfile=data_auto_qc \
    --modelSnps= \      only the random effects in the mixed model are restricted to provided set of SNPs
    --phenoFile=pheCov.txt --phenoCol=qt_1 \
    --covarFile=pheCov.txt --covarCol=sex --qCovarCol=age --qCovarCol=pc{1:10} \
    --LDscoresFile= \   from LDSC
    --geneticMapFile=genetic_map_hg38_withX.txt
    --numThreads=28 --lmm \
    --statsFile=data_auto_qc_bolt_qt_1.stats.txt \
    --verboseStats
```

data_auto_qc_bolt_qt_1.stats.txt

SNP	CHR	BP	GENPOS	ALLELE1	ALLELE0	A1FREQ	F_MISS	CHISQ_ LIINREG	P_ LINREG	BETA	SE	CHISQ_ BOLT_LMM_INF	P_ BOLT_LMM_INF	CHISQ_ BOLT_LMM	P_ BOLT_LMM

Association test – Relative sample + binary trait

- **SAIGE – step1 (run in R)**

```
> fitNULLGLMM(
+ plinkFile="dat_auto_qc",
+ traitType="binary",
+ phenoFile="pheCov.txt",
+ phenoCol="bt_1",
+ covarColList=c("sex","age",paste0("pc",1:10)),
+ qCovarCol="sex",
+ sampleIDColinphenoFile="IID",
+ LOCO=FALSE,
+ IsOverwriteVarianceRatioFile=TRUE,
+ useSparseGRMtoFitNULL=TRUE,
+ outputPrefix="dat_auto_qc_step1",
+ nThreads=72)
```

- **SAIGE – step2 (run in R)**

```
> SPAGMMATtest(
+ bedFile="dat_auto_qc.bed",
+ bimFile="dat_auto_qc.bim",
+ famFile="dat_auto_qc.fam",
+ LOCO=FALSE,
+ is_Firth_beta=TRUE,
+ pCutoffforFirth=0.05,
+ is_output_moreDetails=TRUE,
+ GMMATmodelFile="dat_auto_qc_step1.rda",
+ varianceRatioFile="dat_auto_qc_step1.varianceRatio.txt",
+ SAIGEOutputFile="dat_auto_qc_step2.txt")
```

The two files are generated from step1

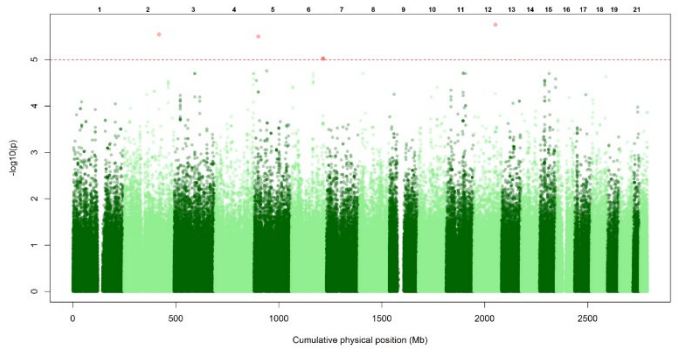
dat_auto_qc_step2.txt

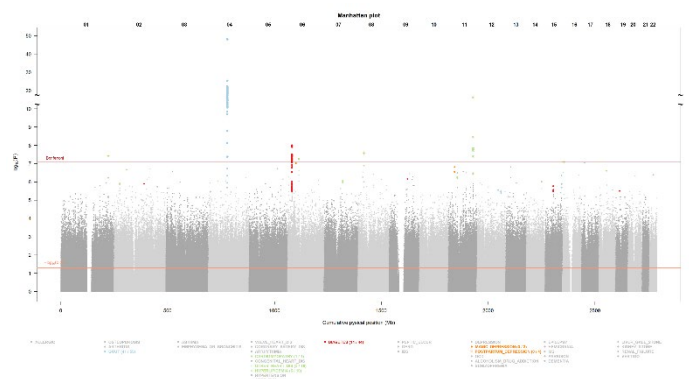
CHR	POS	MarkerID	Allele1	Allele2	AC_Allele2	MissingRate	BETA	SE	Tstat	Var	p.Value	p.value.NA	Is.SPA	AF_case	AF_ctrl	N_case	N_ctrl	N_case_hom	N_case_het	N_ctrl_hom	N_ctrl_het

Association test – Figures

Manhattan plot

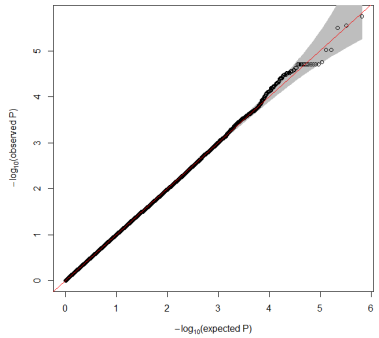
A Manhattan plot is a graphical representation commonly used in GWAS. It helps visualize the statistical significance of genetic variants (usually single nucleotide polymorphisms or SNPs) across the entire genome.





QQ-plot

A Q-Q plot assesses whether observed p-values from a GWAS follow the expected distribution (usually a uniform distribution under the null hypothesis). It helps identify potential population stratification, batch effects, or other issues affecting p-values.



Miami plot

The Miami plot is a less common but visually striking plot used in GWAS.

Thanks for your attention!!
<(_ _)>