STAT588/BIOL588: Genomic Data Science
Lecture 8: Genome-Wide Association Study (GWAS)

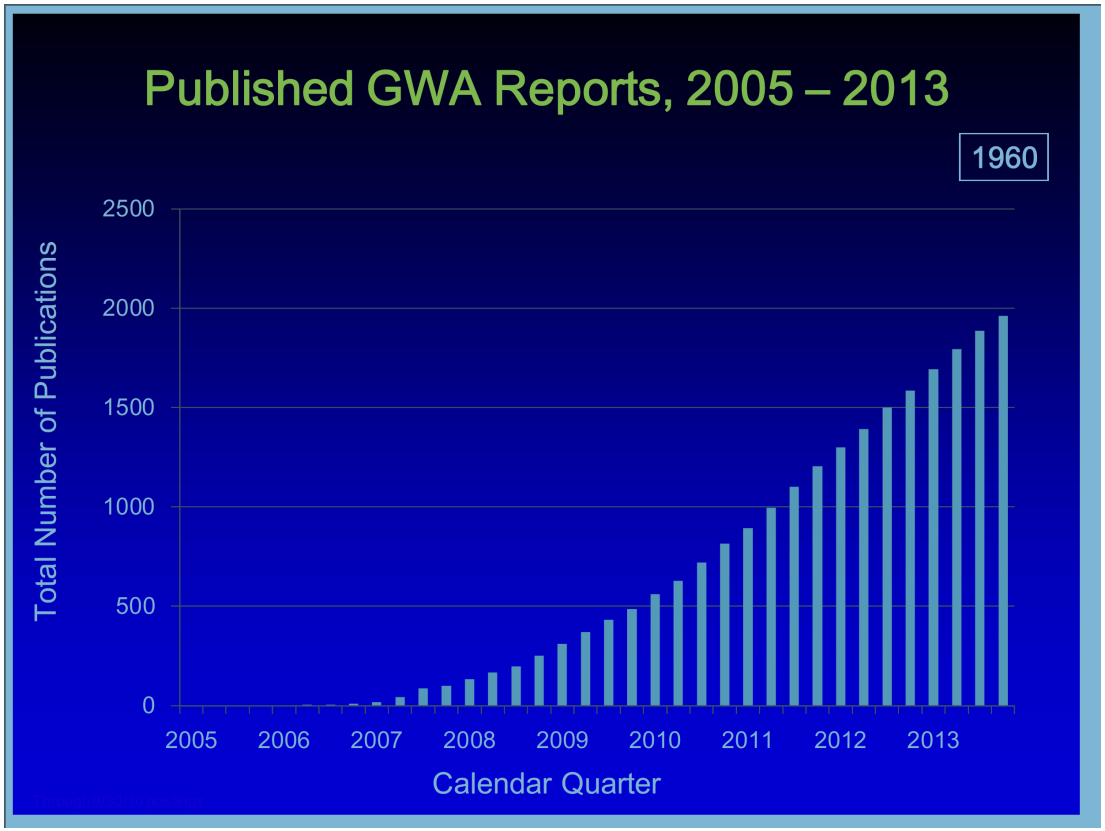
Dr. Yen-Yi Ho (hoyen@stat.sc.edu)

Objectives of Lecture 8

- Data Preparation
- Perform Genome-Wide Association Analysis
- Statistical Power
- Multiple Comparisons
  - Family-Wise Error Rate
  - Bonferroni Correction
  - False Discovery Rate
  - q Value
  - Benjamini Hochberg Adjustment

GWAS Study

Exploratory investigation of genotype-trait association that involves characterization of a large segment of DNA (usually more than 500-1000 Kb region).



GWAS Data Example

Genotypes of 54,977 SNP markers from 83 sheep.

```
> gwas.url<-"http://people.stat.sc.edu/hoyen/BIOL599/Data/SNPxSample.txt"
> gwasdata<-read.table(file=gwas.url, header=T, sep="\t", na.strings="9")
> gwasdata[1:5,1:5]
                             sample1 sample10 sample11 sample12
250506CS3900065000002_1238.1       1        2        1        1
250506CS3900140500001_312.1        2        1        2        1
250506CS3900176800001_906.1        2        1        2        1
250506CS3900211600001_1041.1       2        2        2        1
250506CS3900218700001_1294.1       2        2        1        1
                             sample13
250506CS3900065000002_1238.1        1
250506CS3900140500001_312.1         2
250506CS3900176800001_906.1         2
250506CS3900211600001_1041.1        0
250506CS3900218700001_1294.1        1
```

Recode genotype

```
> gwasdata<-t(gwasdata)
> gdata<-matrix(NA, nrow=nrow(gwasdata), ncol=ncol(gwasdata))
> for(i in 1:ncol(gwasdata)){
+  gdata[,i]<-ifelse(gwasdata[,i]==0, "AA", gdata[,i])
+  gdata[,i]<-ifelse(gwasdata[,i]==1, "AB", gdata[,i])
+  gdata[,i]<-ifelse(gwasdata[,i]==2, "BB", gdata[,i])
+ }
> colnames(gdata)<-colnames(gwasdata)
> dim(gdata)
[1]    83 54977
> gdata[1:3,1:3]
     250506CS3900065000002_1238.1 250506CS3900140500001_312.1
[1,] "AB"                         "BB"
[2,] "BB"                         "AB"
[3,] "AB"                         "BB"
     250506CS3900176800001_906.1
[1,] "BB"
[2,] "AB"
[3,] "BB"
```

Genotyping Errors

A genotyping error occurs when the genotype reported is not the underlying genotype. Often time in a gwas analysis, test for Hardy-Weinberg equilibrium (HWE) is conducted and markers that are not in HWE are removed. There are a number of problems with this approach.

- Deviation from HWE could be due to association between genotypes and trait of interest
- Departure from HWE could be due to population structure
- Multiple hypothesis testing

Despite of all the difficulties, in practice SNPs are frequently dropped from the analysis due to a failure of HWE.

Data Preparation

- Individuals with missing genotypes $> 80\%$ of markers
- Genetic markers with missing genotypes $> 80\%$
- HWE test p values < 0.05
- Minor allele frequency < 0.05

Individuals with missing genotypes

```
> missp<-function(genotype){
+  ans<-length(which(is.na(genotype)))/length(genotype)
+  return(ans)
+ }
> missp(gdata[1,])
[1] 0.1002601
```

The first individual has about 10% missing genotypes.

Individuals with missing genotypes

```
> missp<-function(genotype){
+  ans<-length(which(is.na(genotype)))/length(genotype)
+  return(ans)
+ }
> missp(gdata[1,])
[1] 0.1002601
```

The first individual has about 10% missing genotypes.

Exercise1: Read in the gwas data, remove all individuals with more than 80% missing genotypes.

Markers with missing genotypes

```
> missp<-function(genotype){
+  ans<-length(which(is.na(genotype)))/length(genotype)
+  return(ans)
+ }
> missp(gdata[,10])
[1] 1
```

The 10th marker has about 100% missing genotypes.

Markers with missing genotypes

```
> missp<-function(genotype){
+  ans<-length(which(is.na(genotype)))/length(genotype)
+  return(ans)
+ }
> missp(gdata[,10])
[1] 1
```

The 10th marker has about 100% missing genotypes.

Exercise2: Remove all markers with more than 80% missing genotypes.

HWE

```
> library("HardyWeinberg")
> hwe<-rep(NA, length=ncol(gdata2))
> for(i in 1:ncol(gdata2)){
+  g<-factor(gdata2[,i], levels=c("AA", "AB", "BB"))
+  tab<-table(g)
+  hwe[i]<-HWChisq(tab, verbose=F)[[2]][1]
+  #print(i)
+ }
> rmg2<-which(hwe<0.05)
> length(rmg2)
[1] 3799
> gdata3<-gdata2[,-rmg2]
```

Minor Allele Frequency (MAF)

```
> getMAF<-function(genotype){
+  tmp<-unlist(strsplit(as.character(genotype), split=""))
+  tabf<-table(tmp)/sum(table(tmp))
+  imin<-which.min(tabf)
+  freq<-tabf[imin]
+  names(freq)<-NULL
+  ans<-list(names(tabf[imin]), freq)
+  return(ans)
+ }
> getMAF(gdata3[,1])
[[1]]
[1] "A"
[[2]]
[1] 0.253012
```

Minor Allele Frequency (MAF)

```
> getMAF<-function(genotype){
+  tmp<-unlist(strsplit(as.character(genotype), split=""))
+  tabf<-table(tmp)/sum(table(tmp))
+  imin<-which.min(tabf)
+  freq<-tabf[imin]
+  names(freq)<-NULL
+  ans<-list(names(tabf[imin]), freq)
+  return(ans)
+ }
> getMAF(gdata3[,1])
[[1]]
[1] "A"
[[2]]
[1] 0.253012
```

Exercise3: Record minor allele and remove markers with MAF $< 5\%$.

GWAS analysis: One-SNP-at-a-time

```
> singlesnpM<-function(genotype, y){
+  ina<-which(is.na(genotype))
+  if (length(ina)>0){
+         g<-factor(genotype[-ina])
+         weight<-y[-ina]
+  }else{
+         g<-factor(genotype)
+         weight<-y
+  }
+  tab<-table(g)
+  if(length(tab) < 2){
+         ans<-NA
+  }else{
+          fit<-lm(weight ~ g )
+          fit0<-lm( weight ~ 1)
+          modelc<-anova(fit0, fit)
+          ans<-modelc[[6]][2]
+  }
+  return(ans)
+ }
> weight<-rnorm(nrow(gdata4), mean=50, sd=10)
> singlesnpM(gdata4[,1], y=weight)
[1] 0.0730777
```

GWAS analysis: One-SNP-at-a-time

```
> singlesnpM<-function(genotype, y){
+  ina<-which(is.na(genotype))
+  if (length(ina)>0){
+         g<-factor(genotype[-ina])
+         weight<-y[-ina]
+  }else{
+         g<-factor(genotype)
+         weight<-y
+  }
+  tab<-table(g)
+  if(length(tab) < 2){
+         ans<-NA
+  }else{
+          fit<-lm(weight ~ g )
+          fit0<-lm( weight ~ 1)
+          modelc<-anova(fit0, fit)
+          ans<-modelc[[6]][2]
+  }
+  return(ans)
+ }
> weight<-rnorm(nrow(gdata4), mean=50, sd=10)
> singlesnpM(gdata4[,1], y=weight)
[1] 0.0730777
```

Exercise 4: Perform GWAS analysis for all SNP markers.

Manhattan plot: get SNP map position

```
> map.url<-"http://people.stat.sc.edu/hoyen/BIOL599/Data/SNPmap.txt"
> map<-read.table(file=map.url, header=T, sep="\t", stringsAsFactor=F)
> map[1:5,]
                          name chromosome  position
1 250506CS3900065000002_1238.1         15   5327353
2  250506CS3900140500001_312.1         23  27428869
3  250506CS3900176800001_906.1          7  89002990
4 250506CS3900211600001_1041.1         16  44955568
5 250506CS3900218700001_1294.1          2 157820235
```

Exercise 5: Find genetic position for all the markers in the gwas analysis.

Manhattan plot: prepare data

```
> data2[1:5,]
  CHR        BP         P                                SNP
1  15   5327353 0.0730777 250506CS3900065000002_1238.1
2  23  27428869 0.9058141  250506CS3900140500001_312.1
3   7  89002990 0.5963234  250506CS3900176800001_906.1
4  16  44955568 0.2456556 250506CS3900211600001_1041.1
5   2 157820235 0.9174387 250506CS3900218700001_1294.1
```

CHR is the chromosome; BP is the basepair position of the marker; P is the pvalues; SNP is the name of the SNP.

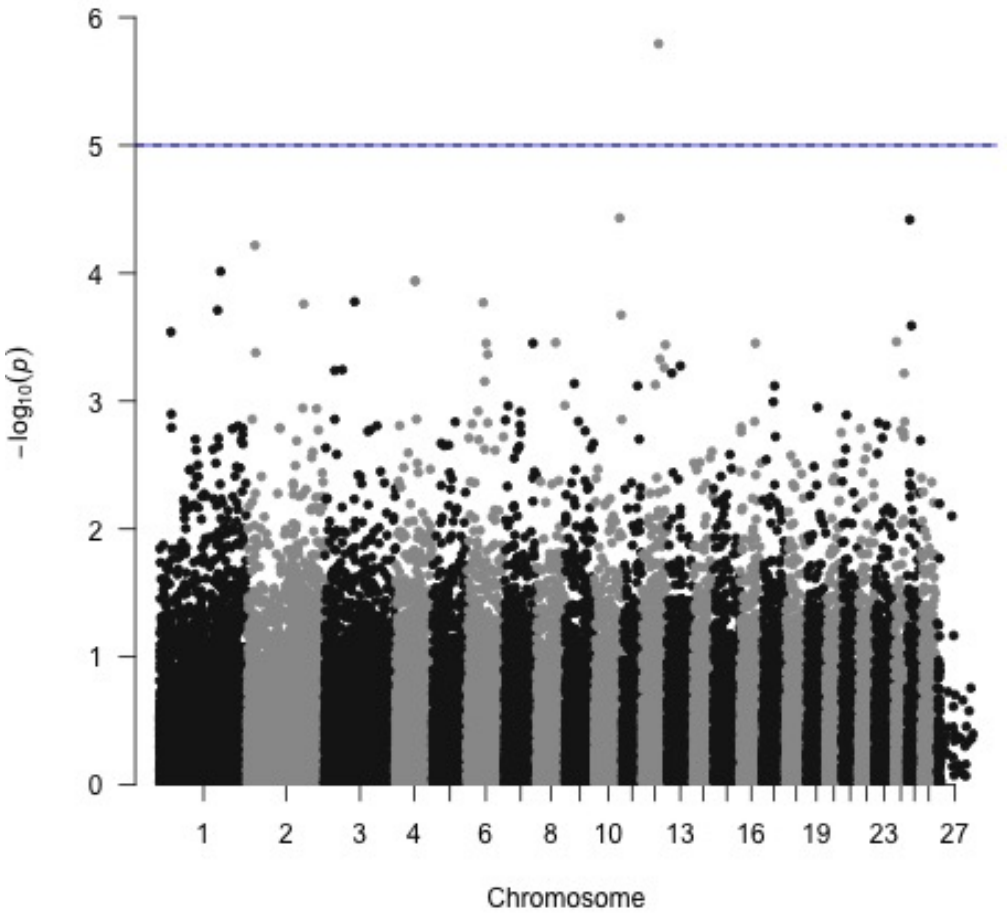
Rename Chromosome X to 27

Exercise 6: Get data2.

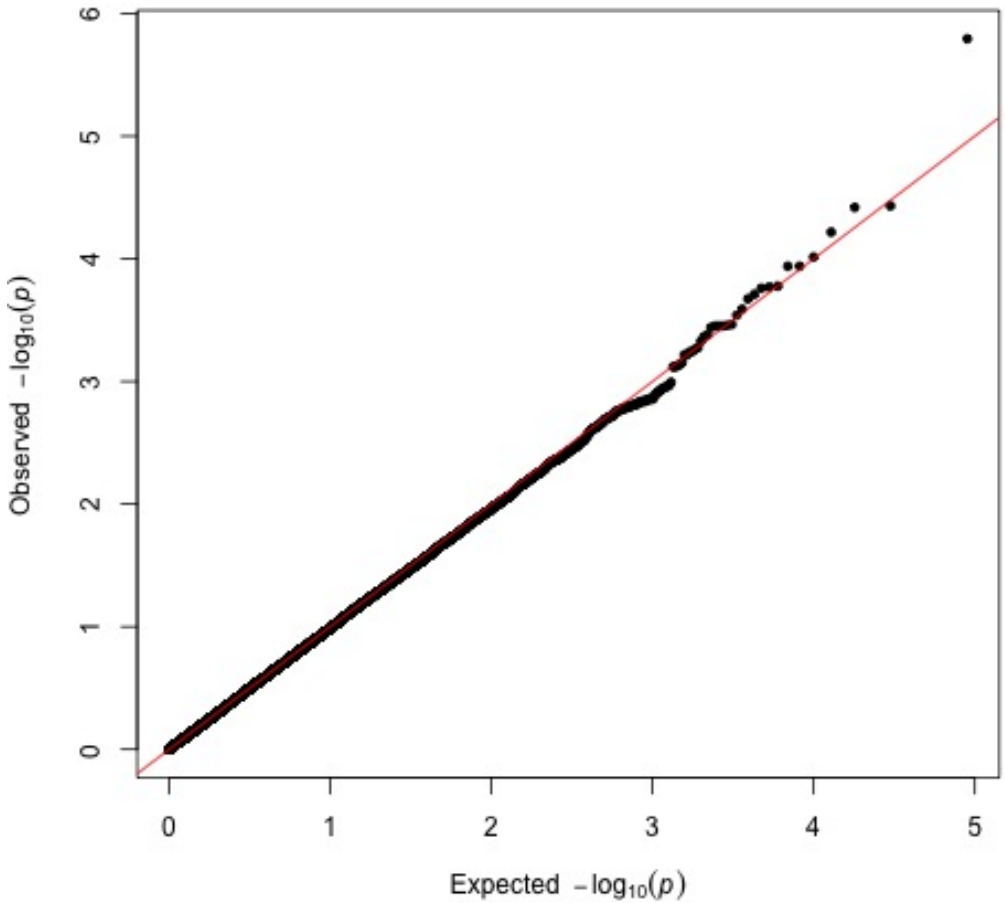
Manhattan plot

```
> library(qqman)
> library(RColorBrewer)
> manhattan(data2)
> abline(h=5, lty=2)
> abline(h=7.4, lty=2)
```

Manhattan plot



QQ plot



Top Table

	SNP	CHR	Minor Allele	MAF	p value
1	s26639.1	12.00	A	0.42	0.00
2	OAR10_85096817.1	10.00	B	0.16	0.00
3	s56341.1	25.00	B	0.22	0.00
4	OAR2_25847803.1	2.00	B	0.46	0.00
5	s08428.1	1.00	B	0.41	0.00
6	OAR4_66612313.1	4.00	A	0.5	0.00
7	OAR4_66622193.1	4.00	A	0.5	0.00
8	s52754.1	3.00	A	0.30	0.00
9	OAR6_55801684.1	6.00	A	0.35	0.00
10	OAR2_192100658.1	2.00	A	0.46	0.00