

Resampling Methods

Applied to GWAS data (one genotype)

Data

```
genos = as.matrix(read.table("./genos.txt"))
phenos = as.matrix(read.table("./phenos.txt"))
#make a binary group label from genotype 1500
table(genos[,1500])
group = ifelse(genos[,1500]==0,0,1)
table(group)
```

Linear model

```
mod = lm(phenos~group)
summary(mod)
# means for g1 = 0 and g1 = 1 groups
mean(phenos[group==0])
mean(phenos[group==1])
m0 = mod$coef[1]
m1 = sum(mod$coef)
m0
m1
```

Student's T test

```
students=t.test(phenos~group,var.equal=TRUE)
students
m0
m1
```

Save the t-statistic and p-value for later

```
stat=students$statistic
pvalue=students$p.value
```

Permutation test

```
sample(group)
table(sample(group)) #repeat a few times
mean(pheno[sample(group)==0])
P=1000
tperm=NULL
for(p in 1:P){
  tperm[p]=t.test(phenos~sample(group),var.equal=TRUE)$statistic
}
```

Stat is positive, so first look at upper end of tperm distribution (1-sided test)

```
mean(tperm>stat)
```

Now do the 2-sided test, which is what t.test does by default

```
permp=mean(abs(tperm)>stat)
permp
pvalue
```

Bootstrap test

```
table(group[sample(1:length(group),replace=TRUE)]) #repeat a few times
B=1000
tboot=NULL
for(b in 1:B){
  bstrap=sample(1:length(group),replace=TRUE)
  tboot[b]=t.test(phenos[bstrap]~group[bstrap])$statistic
}
```

2-sided test

```
bootp=mean(abs(tboot)>stat)
```

Wait - this is a huge p-value tboot is not a null distribution yet!

```
tbootnull=tboot-mean(tboot)
bootp=mean(abs(tbootnull)>stat)
bootp
```