johnsdata

Here I analyse john's data:

```
##   dir  ind bact time extr rin  mol index master.mix
## 1   1 M372 none    4    1 9.8 37.9     2          1
## 2   2 M375 Smeg   48    1 9.3 28.5     4          1
## 3   3 M373   GC   18    1 9.0 48.0     5          1
## 4   4 M374  BCG   48    1 8.8 24.8     6          1
## 5   5 M375 none    4    1 8.8 26.6     7          1
## 6   6 M372 Smeg   48    1 9.5 31.4    12          1
```

Now, let's quantile normalize these counts:

```
voomdat=apply(df_ordered,2,function(x){qqnorm(x,plot.it = F)$x})
```

And transpose it:

```
rep.row<-function(x,n){
  matrix(rep(x,each=n),nrow=n)
}
samplesbygenes=t(voomdat)
mean_voom_features <- apply(samplesbygenes, 2, mean);
voom_class_adj <- samplesbygenes - rep.row(mean_voom_features, dim(samplesbygenes)[1])##subtract the mea

###check to make sure they match


ID=with(anno_ordered,interaction(time,bact),drop=T)
m=model.matrix(~as.factor(ID)-1)
##there are no samples for 48.Staph
m=m[,-24]

rownames(samplesbygenes)[100]
```

```
## [1] "M375.Staph.18"
```

```
which(m[100,]==1)
```

```
## as.factor(ID)18.Staph
##                    23
```

Now fit with univariate ash for each feature:

```
beta=matrix(NA,nrow = dim(voom_class_adj)[2],ncol=dim(m)[2])##make this the J genes by k subgroup betas
se=matrix(NA,nrow = dim(voom_class_adj)[2],ncol=dim(m)[2])
t=matrix(NA,nrow = dim(voom_class_adj)[2],ncol=dim(m)[2])
colnames(se)=colnames(beta)=colnames(t)=as.matrix(levels(ID)[-24])
for(k in 1:ncol(m)){
```

```
  fit=lm((voom_class_adj)~(m[,k]-1))
  a=matrix(unlist(coef(summary(fit))),byrow = T,nrow=ncol(voom_class_adj))
  beta[,k]=a[,1]
  se[,k]=a[,2]
  t[,k]=a[,3]
}


write.table(beta,"betafit.txt",col.names = T)
write.table(se,"sefit.txt",col.names = T)
write.table(t,"tfit.txt",col.names = T)
```

```
t=read.table("tfit.txt")
b=read.table("betafit.txt")
se=read.table("sefit.txt")

s.j=se/se

lf.ash=matrix(NA,ncol=ncol(t),nrow=nrow(t))

for(i in 1:ncol(t)){
  lf.ash[,i]=ash(betahat = t[,i],sebetahat = s.j[,1],mixcompdist = "normal")$lfsr
}


thresh=0.05

index=which(rowSums(lf.ash<thresh)>0)

write.table(t[index,],"maxt.txt",col.names = F ,row.names = F)


system('/Users/sarahurbut/miniconda3/bin/sfa -gen maxt.txt -g 12538 -k 5 -n 26 i -o john')

A="john"

factor.mat=as.matrix(read.table("john_F.out"))
lambda.mat=as.matrix(read.table("john_lambda.out"))

library('mash')
cov=compute.covmat(b.gp.hat = t,sebetahat = s.j,Q =5, t.stat=t[index,],lambda.mat=lambda.mat,P=3,A=A, fa

omega=compute.covmat(b.gp.hat = t,sebetahat = s.j,Q =5, t.stat=t[index,],lambda.mat=lambda.mat,P=3,A=A,

library('mvtnorm')
compute.hm.train.log.lik.pen(train.b = t,se.train = s.j,covmat = cov,A=A,pen=1)


pis=readRDS(paste0("pis",A,".rds"))$pihat
b.test=t
se.test=s.j
weightedquants=lapply(seq(1:nrow(b.test)),function(j){total.quant.per.snp(j,cov,b.gp.hat=b.test,se.gp.ha
```

Now let's plot the most interesting covariance matrix:

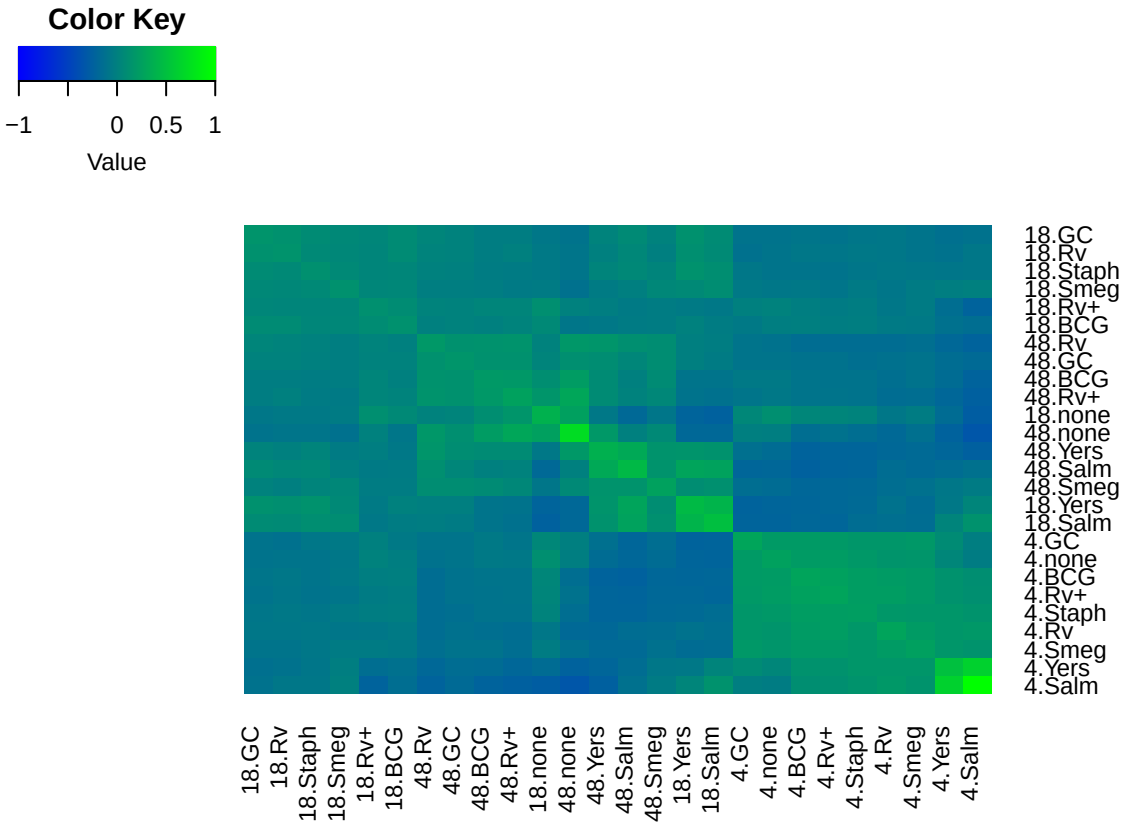
```
t=read.table("tfit.txt")
lf.ash=read.table('lfash.txt')
cov=readRDS("covmatjohn.rds")
library('gplots')
```

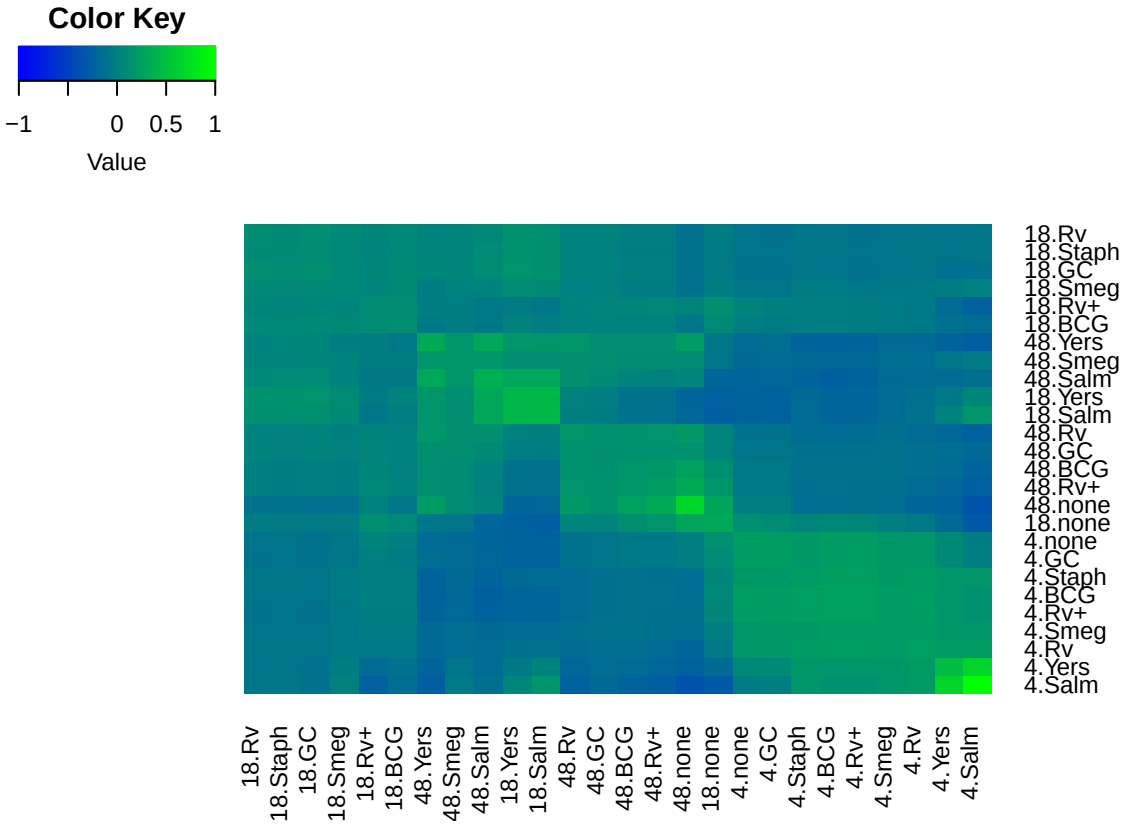
```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library('colorRamps')

for(k in c(2,3,9)){
  x=cov[[k]]/max(diag(cov[[k]]))
colnames(x)=rownames(x)=as.matrix(levels(ID)[-24])[,1]
  heatmap.2(x,revC = T,col = blue2green(256),dendrogram="none",density="none",trace="none")
}
```






```
lfsr=read.table("johnlfsr.txt")[,-1]
postt=read.table("johnposterior.means.txt")[,-1]
mean(lfsr<0.05)
```

```
## [1] 0.6475276
```

```
mean(lf.ash<0.05)
```

```
## [1] 0.2230636
```

```
##and this percent of values are shrunk###
mean(abs(postt)<abs(t))
```

```
## [1] 0.6314576
```