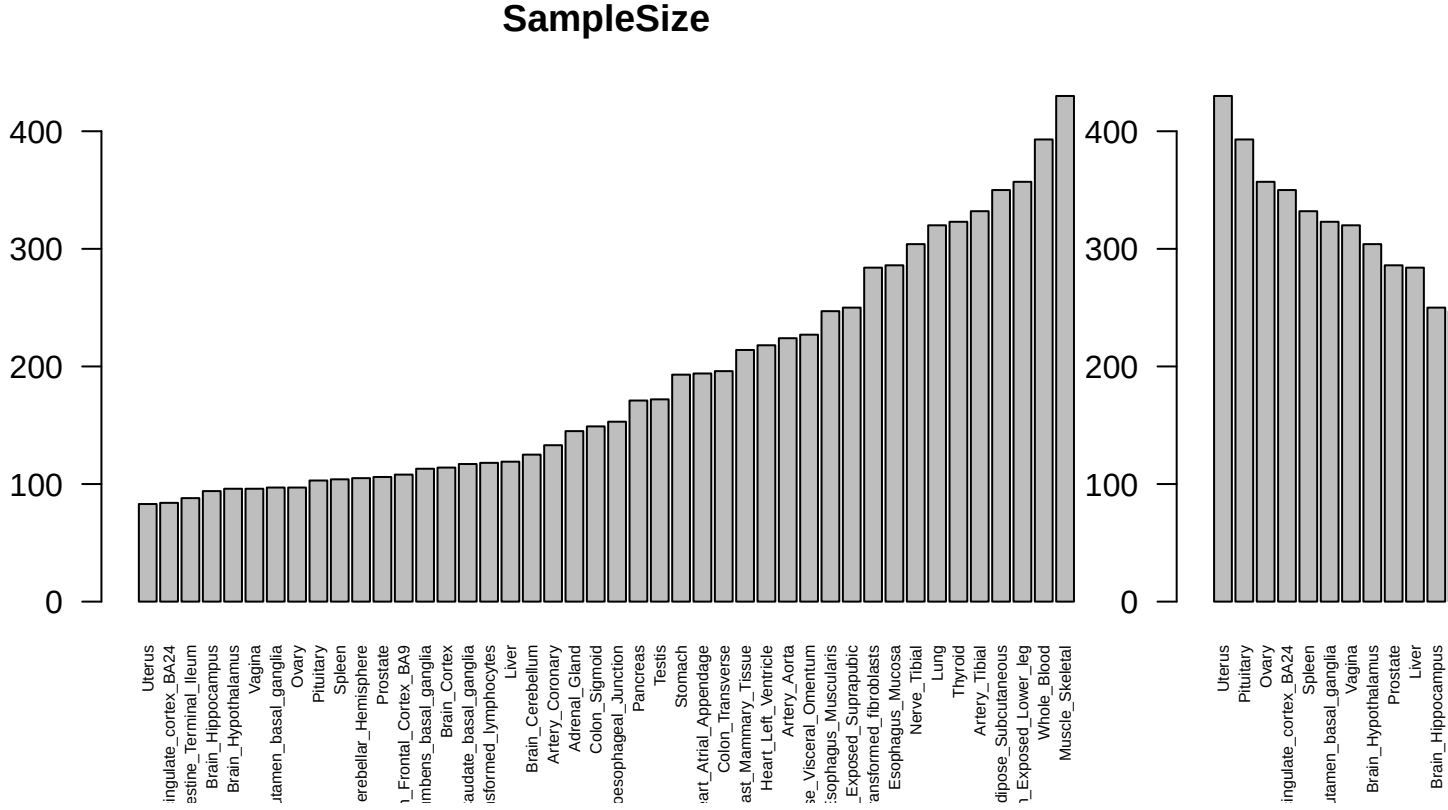
StandardError

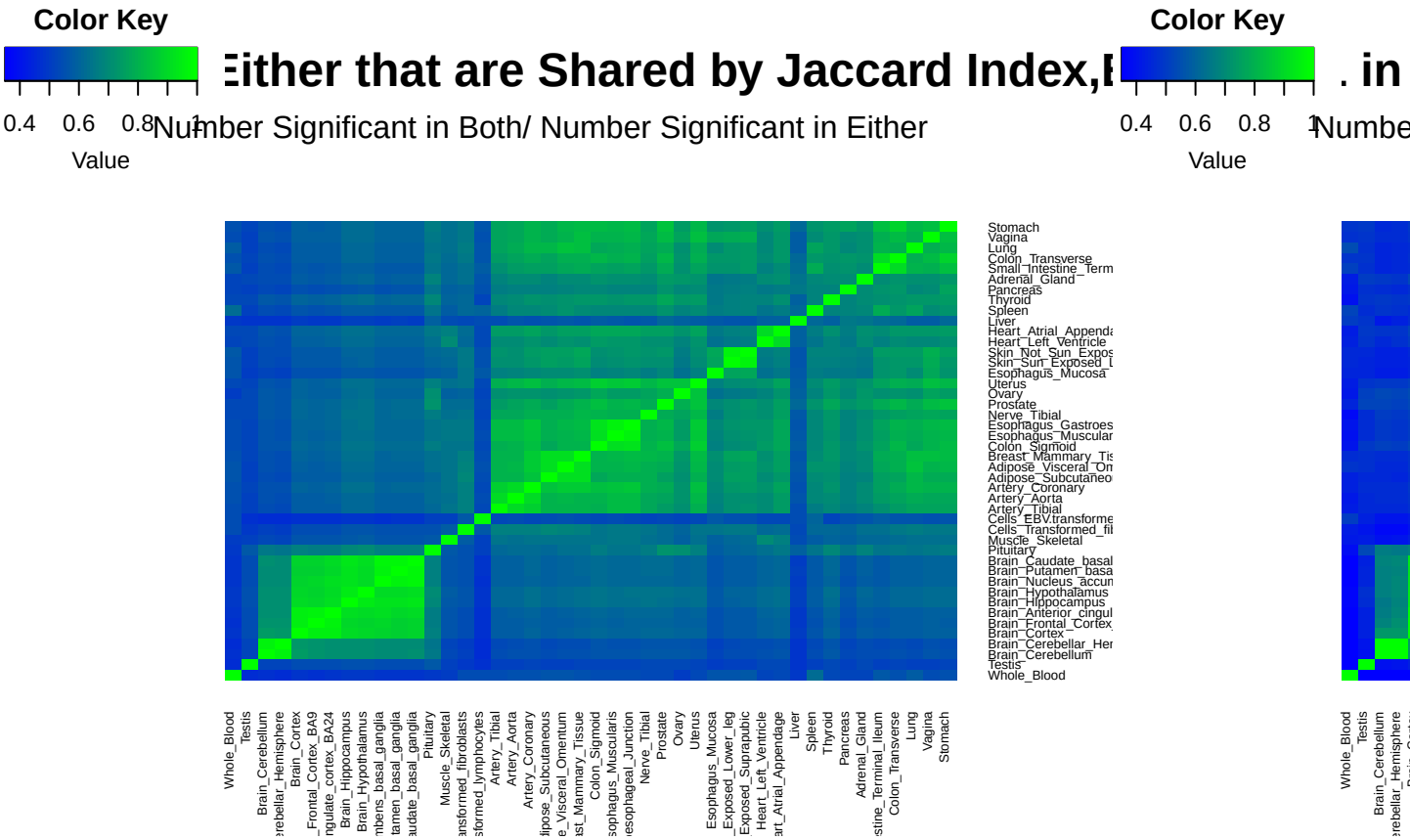
The aim of this document is to investigate the correlation of standard error and sample size, and to show how the presence of small sample sizes and large standard errors in biologically 'unique' tissues drives incompatibilities between the fold-size sharing heatmap and significance sharing heatmap.

Look at the ordering of Sample Size and see how it is almost identical to that of standard error, though no sample sizes differ by more than about 4 fold.

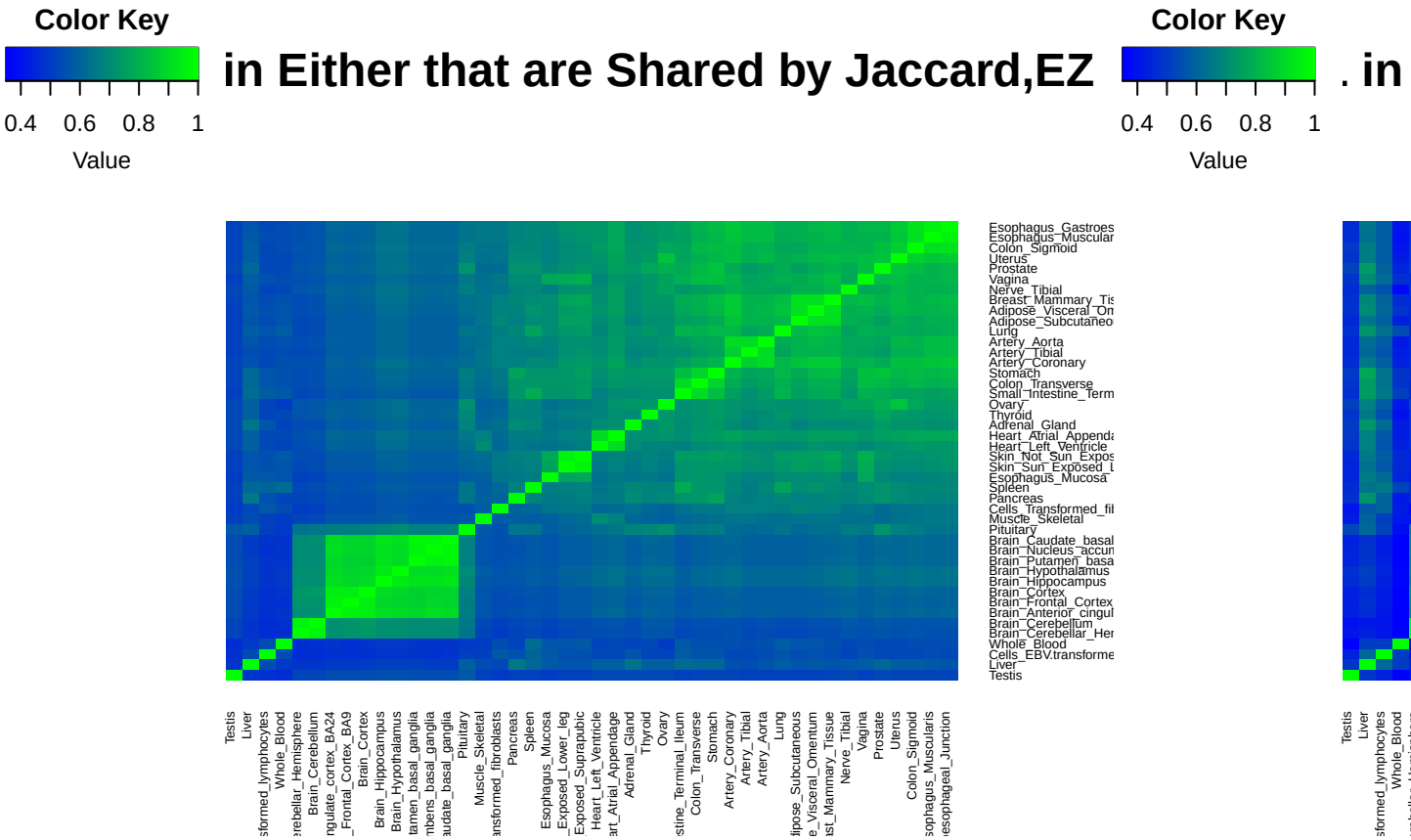
The correlation between the sqrt of the sample size and the median standard error is -0.9575283.



Now note that the non 'system-group' (i.e., Gender like uterus, ovary and vagina, or brain, which exhibit similar effects based on biological intuition) tissues which show the strongest discrepancy between inconsistent significance and sharing by effect size are exactly those with smallest sample size: the Liver and Pituitary (not really part of the brain). So my intutition was correct: the lack of consistency in significance (as reflected by sparsity in the Jaccard Index Plot) in the presence of consistency in effect size is due to the small sample size of 'biologically unique' tissue.

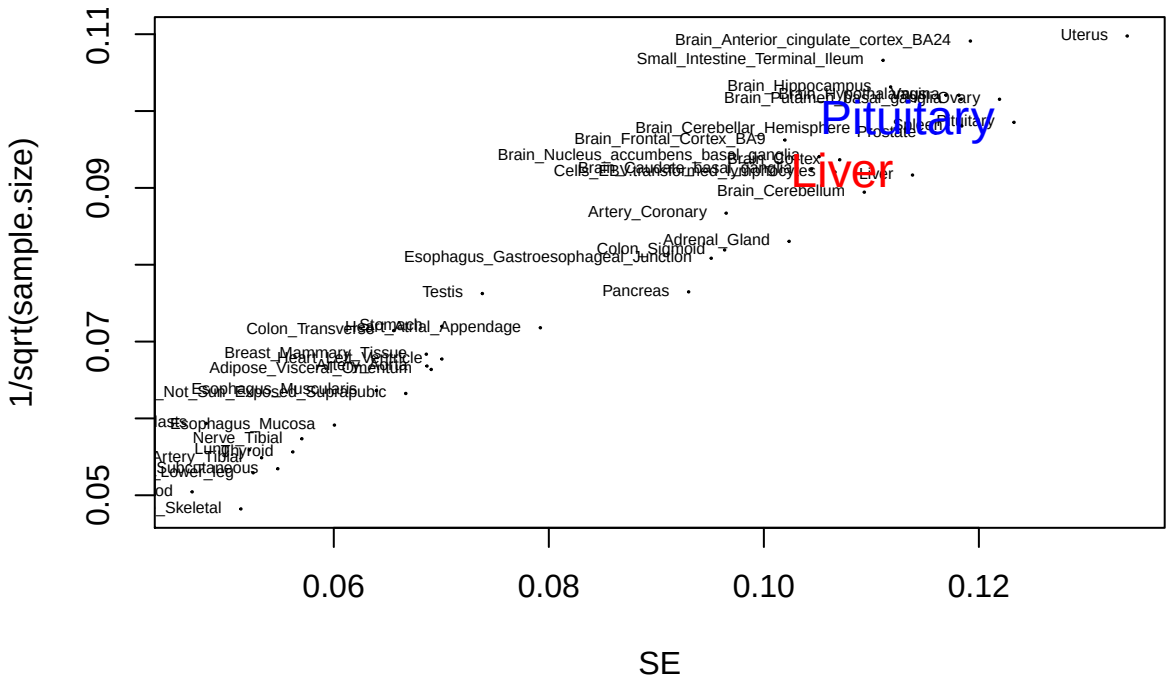


This difference persists with EE model as well (plots not shown for simplicity) which is somewhat reassuring from a methodological standpoint. The good news is, for the purpose of the paper, I think a little reordering of the tissues might fend off a referee :)



Show the idea of how 'loner' tissues which show an unusual relationship between Effect and Significance tend to have a small sample size and median. standard. error.

```
#size.no.brain=sqrt(size[-c(7:16)])
#median.se.no.brain=median.standard.error[-c(7:16)]
median.se=median.standard.error
plot(median.se,1/sqrt(size),cex=0.1,ylab="1/sqrt(sample.size)",xlab="SE")
text(median.se,1/sqrt(size),colnames(maxz),cex=0.5,pos=2)
text(median.se["Liver"],1/sqrt(size)["Liver"],"Liver",col="red",cex=1.5,pos=2)
text(median.se["Pituitary"],1/sqrt(size)["Pituitary"],"Pituitary",col="blue",cex=1.5,pos=2)
```



And this translates to marginal variances:

```
b.test=read.table("../Data/maxz.txt")
se.test=b.test/b.test
library('mvtnorm')
source("~/matrix_ash/R/main.R")
source("~/matrix_ash/R/mixEm.R")
j=14608
r=38
k=1073
cov=readRDS("../Data/covmatAug13withED.rds")
pis=readRDS("../Data/pisAug13withED.rds")$pihat

all.arrays=post.array.per.snp(j=j,covmat = cov,b.gp.hat = b.test,se.gp.hat = se.test)

b.mle=as.vector(t(b.test[j,]))##turn i into a R x 1 vector
V.gp.hat=diag(se.test[j,])^2
V.gp.hat.inv <- solve(V.gp.hat)

log.lik.snp=log.lik.func(b.mle,V.gp.hat,cov)
log.lik.minus.max=log.lik.snp-max(log.lik.snp)
#log.pi=log(pis)
#s=log.lik.minus.max+log.pi
exp.vec=exp(log.lik.minus.max)
post.weights=t(exp.vec*pis/sum(exp.vec*pis))

U.gp1kl <- (post.b.gpkl.cov(V.gp.hat.inv, cov[[k]]))
mu.gp1kl <- as.array(post.b.gpkl.mean(b.mle, V.gp.hat.inv, U.gp1kl))

plot(all.arrays$post.covs[k,],diag(U.gp1kl))
```


```
##make sure individual posterior covariances are not greater than likliehood variance
sum(all.arrays$post.covs>1)
```

```
## [1] 0
```

$Var(Y_r) = E(Y_r^2) - E(Y_r)^2$

So for $E(Y^2)$, we integrate over K, where Z is the random variable with $P(Z = k) = \tilde{\pi}_k$:

$E(Y_r^2) = E(E(Y_r^2|Z)$

$= \sum_k \tilde{\pi}_k[U_{kr} + \mu_{kr}^2]$

So $Var(Y_r) = \sum_k \tilde{\pi}_k[U_{kr} + \mu_{kr}^2]$ - $(\sum_k \tilde{\pi}_k \mu_{kr})^2$

Let's check our calculations for a given tissue:

```
marginal.var=read.table("../../Dropbox/Aug12/Aug13withEDmarginal.var.txt")[,-1]
#r=sample(seq(1:44),1)
r=38
pi.kl=post.weights


sum.test.part.one=post.weights*(all.arrays$post.covs[,r]+all.arrays$post.means[,r]^2)

##the majority (53% of wiehgt at componenet 1073)
####show example
diag(U.gp1kl)[r] ##less than 1
```

```
## [1] 0.963171
```

```
mu.gp1kl[r]^2
```

```
## [1] 23.45535
```

```
part.one.at.k=post.weights[k]*(all.arrays$post.covs[k,r]+all.arrays$post.means[k,r]^2)

mu.2.atk=(post.weights[k]*all.arrays$post.means[k,r])^2


sum.test.e.y.2.r=(post.weights*all.arrays$post.means[,r])

sum(sum.test.part.one)-sum(sum.test.e.y.2.r)^2
```

```
## [1] 5.83046
```

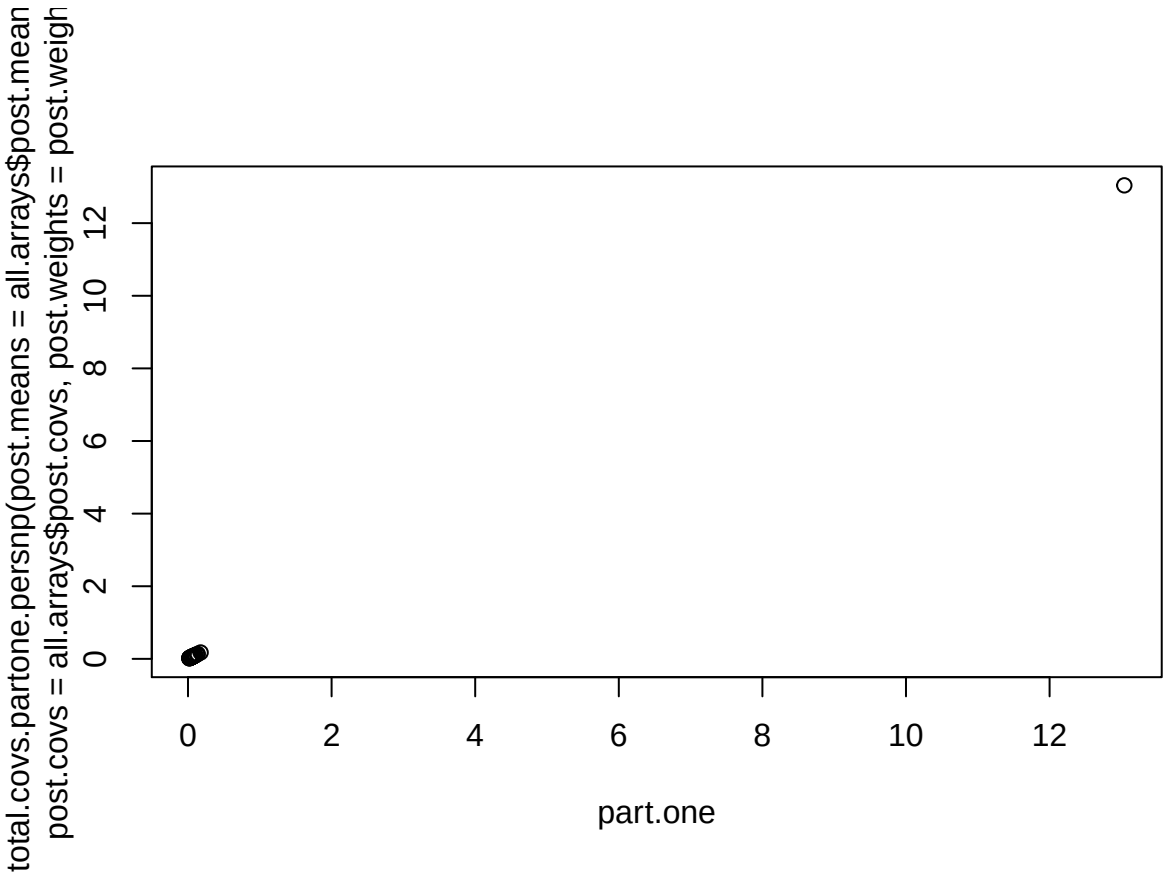
```
part.one.r=post.weights%*%(all.arrays$post.covs[,r]+all.arrays$post.means[,r]^2)
e.y.2.r=(post.weights%*%(all.arrays$post.means[,r]))^2
part.one.r-e.y.2.r
```

```
##          [,1]
## [1,] 5.83046
```

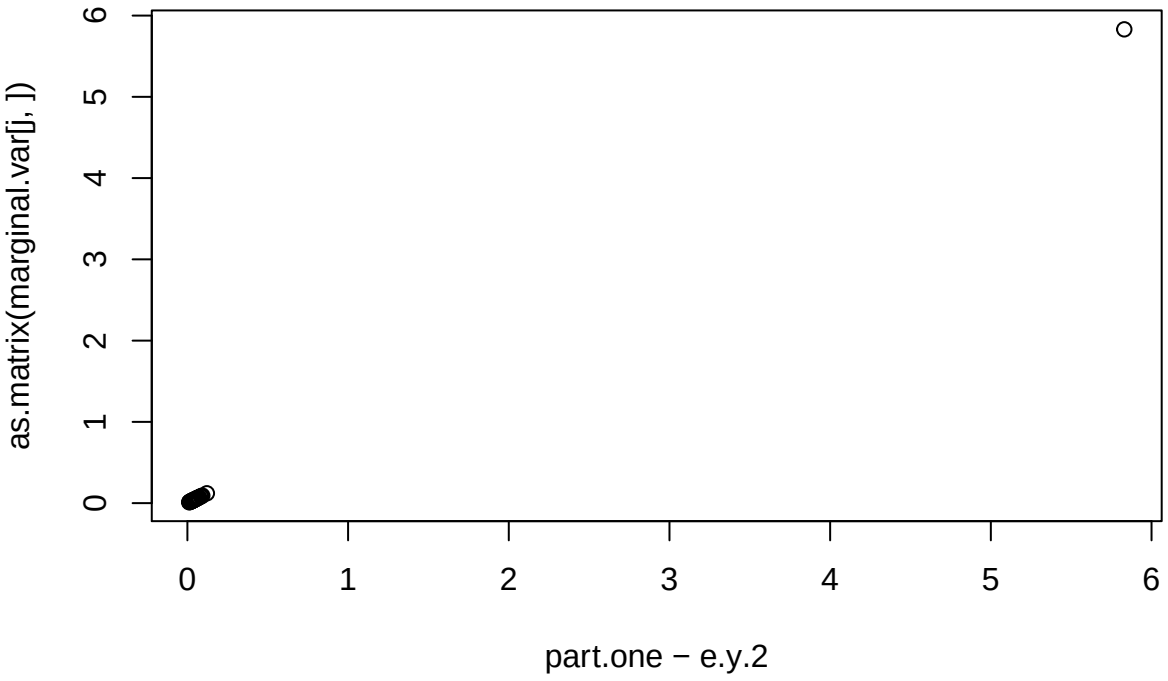
```
marginal.var[j,r]
```

```
## [1] 5.83046
```

```
##test for individula component
k=which.max(post.weights)


###show function correct for the first part acorss all tissues
part.one=post.weights%*%(all.arrays$post.covs+all.arrays$post.means^2)


plot(part.one,total.covs.partone.persnp(post.means = all.arrays$post.means,post.covs = all.arrays$post.
```




```
####and check across all check with marginal variance computation##
e.y.2=(post.weights%*%(all.arrays$post.means))^2

plot(part.one-e.y.2,as.matrix(marginal.var[j,]))
```



```
part.one.tim=function(post.weights,post.covs){
  post.weights%*%post.covs
}

part.two.tim=function(post.weights,post.means,tissue){
  r=tissue
  grand.mean=post.weights%*%post.means
  post.weights%*%((post.means[,r]-grand.mean[r])^2)
}


b.test=read.table("../Data/maxz.txt")
se.test=b.test/b.test

j=14608
r=38
k=1073

j=100
r=10

cov=readRDS("../Data/covmatAug13withED.rds")
pis=readRDS("../Data/pisAug13withED.rds")$pihat

all.arrays=post.array.per.snp(j=j,covmat = cov,b.gp.hat = b.test,se.gp.hat = se.test)

b.mle=as.vector(t(b.test[j,]))##turn i into a R x 1 vector
V.gp.hat=diag(se.test[j,])^2
V.gp.hat.inv <- solve(V.gp.hat)

log.lik.snp=log.lik.func(b.mle,V.gp.hat,cov)
log.lik.minus.max=log.lik.snp-max(log.lik.snp)
#log.pi=log(pis)
#s=log.lik.minus.max+log.pi
exp.vec=exp(log.lik.minus.max)
post.weights=t(exp.vec*pis/sum(exp.vec*pis))

part.one.tim(post.weights = post.weights,post.covs = all.arrays$post.covs)[r]
```

```
## [1] 0.493777
```

```
part.two.tim(post.weights = post.weights,post.means = all.arrays$post.means,tissue = r)
```

```
##              [,1]
## [1,] 7.22546e-05
```

```
part.one.tim(post.weights = post.weights,post.covs = all.arrays$post.covs)[r]+part.two.tim(post.weights
```

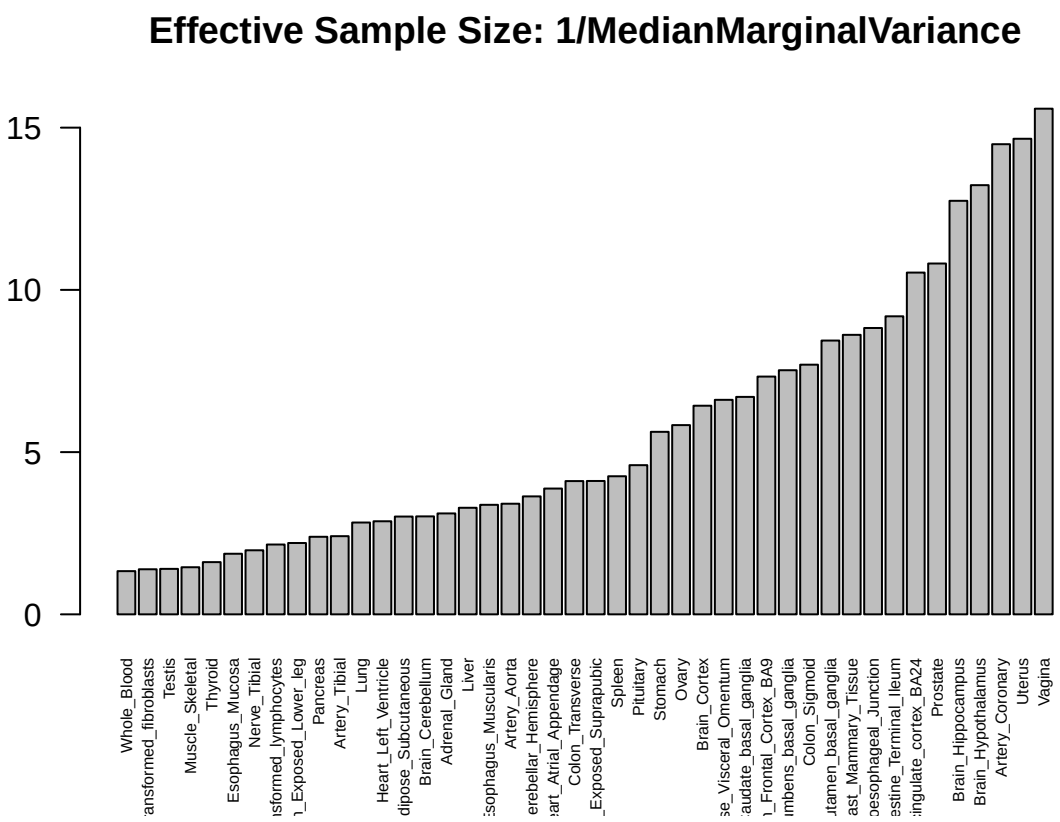
```
##           [,1]
## [1,] 0.4938492
```

```
total.covs.partone.persnp(all.arrays$post.means,all.arrays$post.covs,post.weights)[r]-((post.weights%*%
```

```
## [1] 0.4938492
```

Now we look at the median poserior variances:

```
median.mar.var=apply(marginal.var,2,median)
barplot(sort(1/median.mar.var,decreasing=F),cex.names=0.5,main="Effective Sample Size: 1/MedianMarginal
```

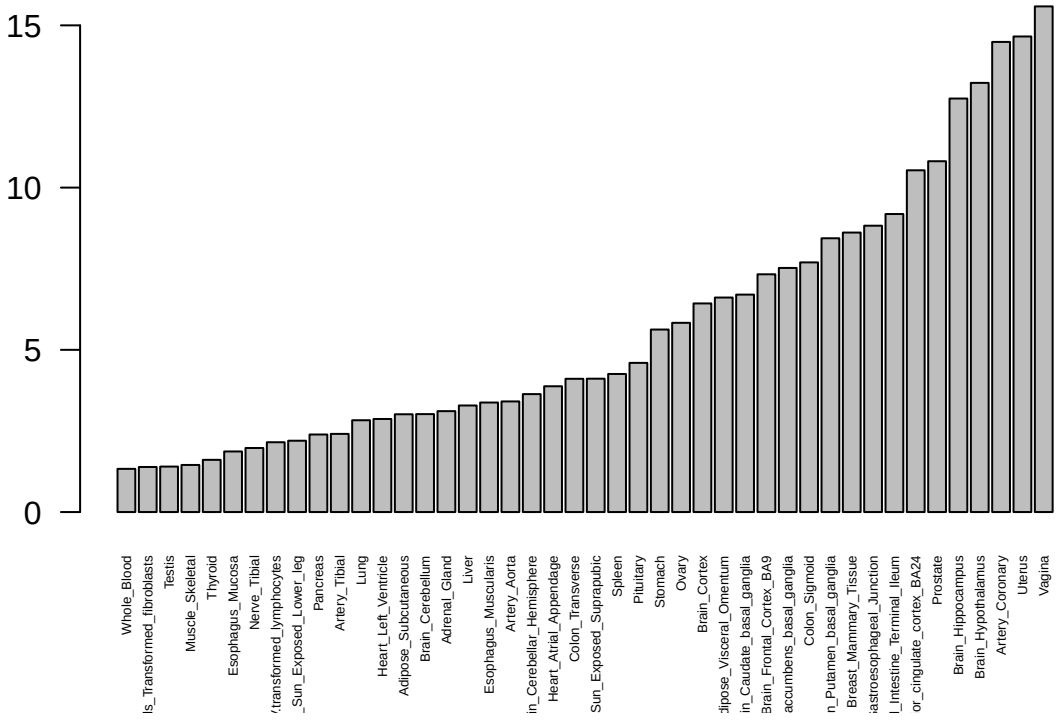


Now let's plot effective sample size. Recall:

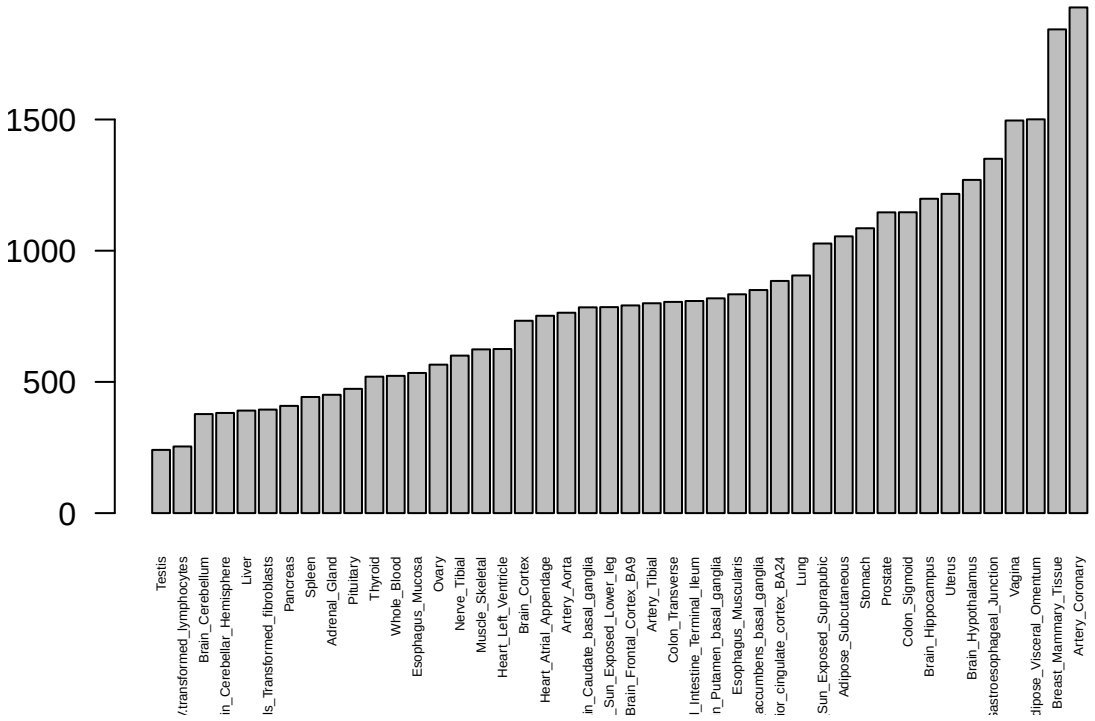
$$n_{jeff} = \frac{s_j^2}{\tilde{s}_j^2}$$

```
original.var=as.matrix(standard.error.from.z)^2
#original.var=(standard.error.from.z/standard.error.from.z)^2
size=as.matrix(exp.sort)
post.var=as.matrix(marginal.var)*standard.error.from.z^2
njeffective=size*original.var/post.var

gtex.colors=read.table('../Data/GTExColors.txt', sep = '\t', comment.char = '')[-missing.tissues,2]
missing.tissues=c(7,8,19,20,24,25,31,34,37)

##ask why some of the origianl variancres are smaller than posterior variances, even though calculations
###15093  44
increase=njeffective/size
```

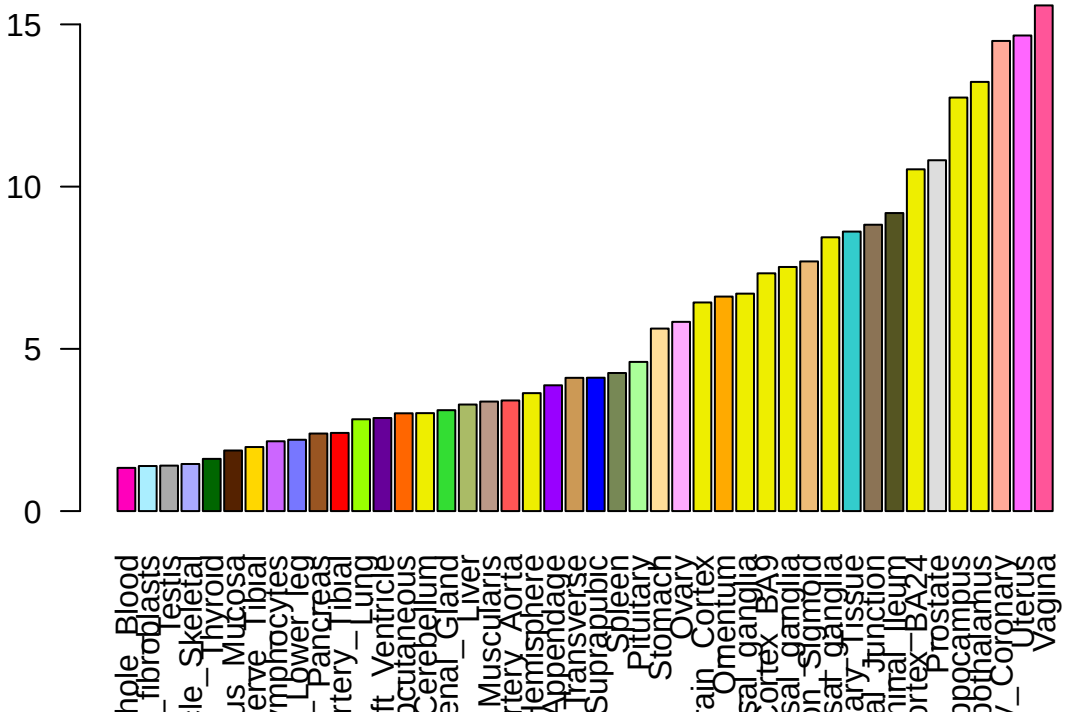
```
barplot(sort(apply(increase,2,median),decreasing=F),las=2,cex.names=0.4)
title("Median(Nj_effective/Nj_original)")
```



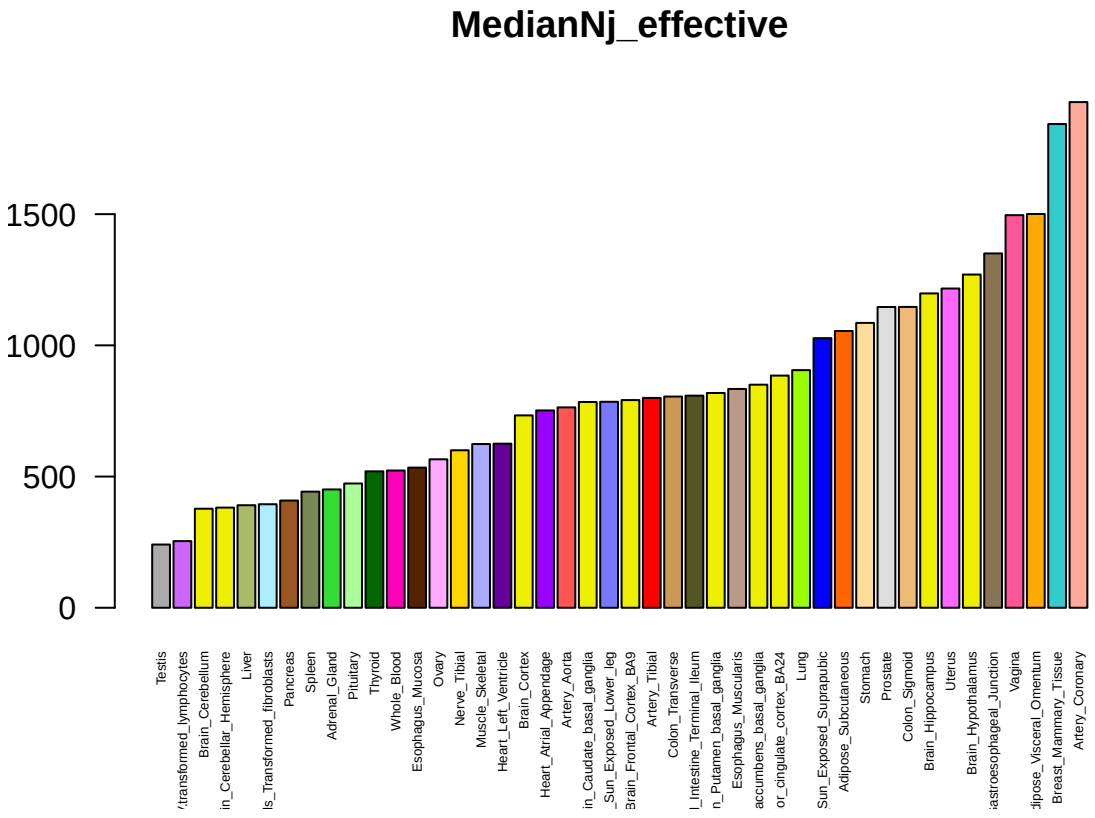

```
barplot(sort(apply(njeffective,2,median),decreasing=F),cex.names=0.4,las=2)
title("MedianNj_effective")
```




```
barplot(sort(apply(increase,2,median),decreasing=F),las=2,col=as.character(gtex.colors[order(apply(incr
title("Median(Nj_effective/Nj_original)")
```

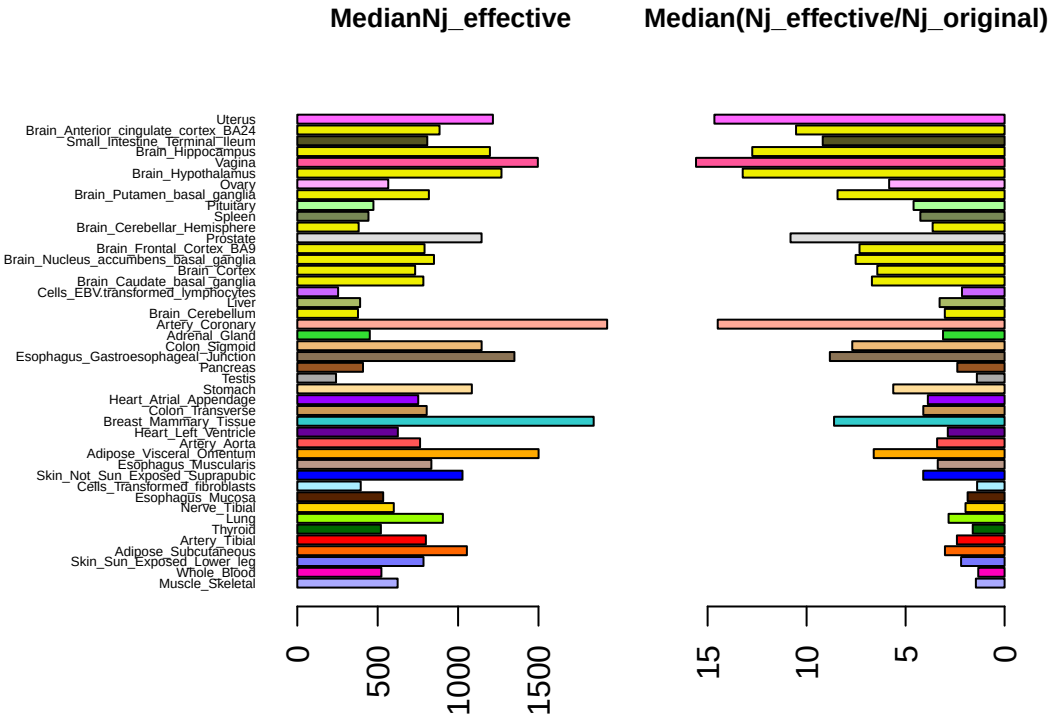


```
barplot(sort(apply(njeffective,2,median),decreasing=F),cex.names=0.4,las=2,col=as.character(gtex.colors
title("MedianNj_effective")
```



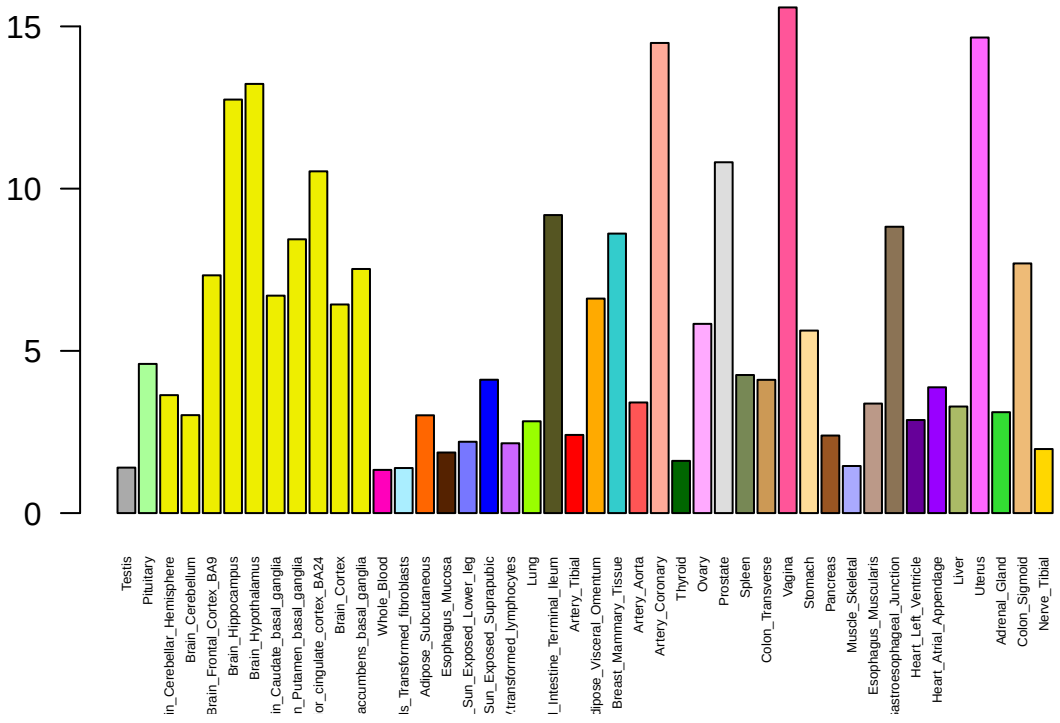
Let's plot again with order by original sample size:

```
par(mfrow=c(1,2))
samplesize=apply(size,2,function(x){unique(x)})
sampleorder=order(samplesize,decreasing = T)
median.nj.effective=apply(njeffective,2,median)
median.nj.increase=apply(increase,2,median)

par(mar=c(5.1,8,4.1,0.1))
barplot(median.nj.effective[sampleorder],cex.names=0.4,las=2,col=as.character(gtex.colors[sampleorder])
title("MedianNj_effective",cex.main=0.8)
par(mar=c(5.1,2,4.1,6))
barplot(median.nj.increase[sampleorder],cex.names=0.4,las=2,col=as.character(gtex.colors[sampleorder]),
title("Median(Nj_effective/Nj_original)",cex.main=0.8)
```

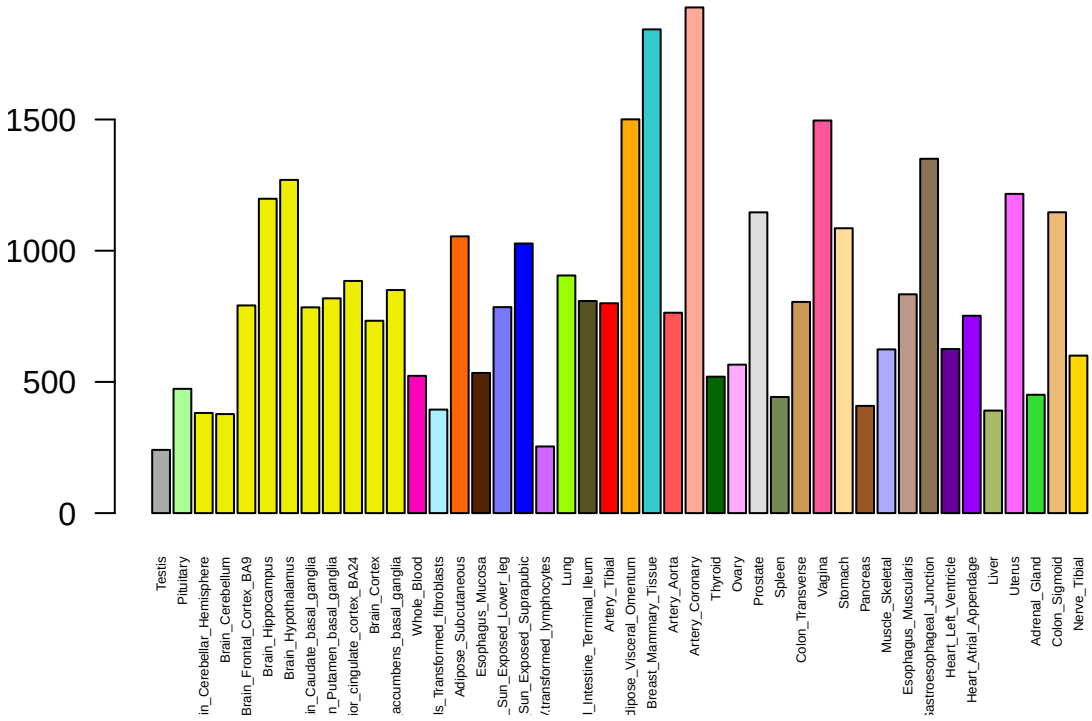


Now, let's plot with shuffled indices:

```
##now reorder by tissue colors##

uk3labels=read.table("../Analysis/uk3rowindices.txt")[,1]
gtex.colors.shuffle=gtex.colors[uk3labels]


##shuffleincrease

increaseshuffle=increase[,uk3labels]
barplot(apply(increaseshuffle,2,median),las=2,cex.names=0.4,col=as.character(gtex.colors.shuffle))
title("Median(Nj_effective/Nj_original)")
```




```
njeffectiveshuffle=njeffective[,uk3labels]

barplot(apply(njeffectiveshuffle,2,median),cex.names=0.4,las=2,col=as.character(gtex.colors.shuffle))
title("MedianNj_effective")
```



Show that my results V(B|D)=E(E(B^2|D,K))-E(E((B|D,k) are the same as V(B|D)=E(V(B|D,K))+V(E(B|D,k))

```
j=10000
b.mle=as.vector(t(b.test[j,]))##turn i into a R x 1 vector
se.test=b.test/b.test

V.gp.hat=diag(se.test[j,])^2
V.gp.hat.inv <- solve(V.gp.hat)
all.arrays=post.array.per.snp(j=j,covmat = cov,b.gp.hat = b.test,se.gp.hat = se.test)

log.lik.snp=log.lik.func(b.mle,V.gp.hat,cov)
log.lik.minus.max=log.lik.snp-max(log.lik.snp)
#log.pi=log(pis)
#s=log.lik.minus.max+log.pi
exp.vec=exp(log.lik.minus.max)
post.weights=t(exp.vec*pis/sum(exp.vec*pis))


v1=post.weights%*%all.arrays$post.covs+post.weights%*%(t(apply((all.arrays$post.means),1,'-',(post.weigh

v2=post.weights%*%(all.arrays$post.covs+all.arrays$post.means^2)-(post.weights%*%all.arrays$post.means)
```