

# practical 3: Inbreeding and kinship

Jerome Goudet and Bruce Weir

2021-07-15

```
correction<-FALSE
```

```
library(gaston)
library(hierfstat)
library(JGTeach)
```

For this practical we will use both simulated pedigrees, simulated genetic data on these pedigrees and the small chunk of chromosome 22 from the 1000 genomes.

## Kinship and inbreeding from pedigrees

1. load a pedigree from a monogamous population using `read.table`. Explore the pedigree : what are the first, second and third column? How many founders in this pedigree? Find at least two families with sibships of four (just inspect the pedigree).

```
ped<-read.table("https://www2.unil.ch/popgen/teaching/SISGData/pedmono.txt",header=TRUE)
str(ped)
```

```
## 'data.frame':   195 obs. of  3 variables:
## $ ind : int  1 2 3 4 5 6 7 8 9 10 ...
## $ dam : int  NA NA NA NA NA NA NA NA NA NA ...
## $ sire: int  NA NA NA NA NA NA NA NA NA NA ...
```

```
dim(ped)
```

```
## [1] 195  3
```

```
# number of founders
# founders are individuals with missing sires and dams
```

```
sum(is.na(ped$sire) & is.na(ped$dam))
```

```
## [1] 20
```

```
# families: 28:32; 59:62
```

```
#first inds of gen 1:5 : c(21,47,77,117,159)
```

First column is individual id, second is dam and third is sire.

Here, dams and sires are arbitrary, as I simulated a hermaphroditic population.

20 founders (individuals with no known parents). to find the families, the pedigree is small, so this can be explore by hands

- Calculate the additive relationship matrix of individuals from this pedigree using `JGTeach::pedARM`. What is the pedigree relatedness among the sibship previously identified?

and of these individuals with their parents? Examine sibship 92:95. What is special about it?

```
ARM<-JGTeach::pedARM(ped$sire,ped$dam)
```

```
ARM[28:32,28:32]
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,] 1.0 0.5 0.5 0.5 0.5
## [2,] 0.5 1.0 0.5 0.5 0.5
## [3,] 0.5 0.5 1.0 0.5 0.5
## [4,] 0.5 0.5 0.5 1.0 0.5
## [5,] 0.5 0.5 0.5 0.5 1.0
```

```
ped[28:32,]
```

```
##      ind dam sire
## 28 28 14 10
## 29 29 14 10
## 30 30 14 10
## 31 31 14 10
## 32 32 14 10
```

```
ARM[28:32,c(ped[28,2],ped[28,3])]
```

```
##      [,1] [,2]
## [1,] 0.5 0.5
## [2,] 0.5 0.5
## [3,] 0.5 0.5
## [4,] 0.5 0.5
## [5,] 0.5 0.5
```

```
ARM[59:62,59:62]
```

```
##      [,1] [,2] [,3] [,4]
## [1,] 1.0 0.5 0.5 0.5
## [2,] 0.5 1.0 0.5 0.5
## [3,] 0.5 0.5 1.0 0.5
## [4,] 0.5 0.5 0.5 1.0
```

```
ARM[59:62,c(ped[59,2],ped[59,3])]
```

```
##      [,1] [,2]
## [1,] 0.5 0.5
## [2,] 0.5 0.5
## [3,] 0.5 0.5
## [4,] 0.5 0.5
```

```
#relatedness higher than 0.5 with their sibs
```

```
ARM[92:95,92:95]
```

```
##      [,1] [,2] [,3] [,4]
## [1,] 1.03125 0.56250 0.56250 0.56250
## [2,] 0.56250 1.03125 0.56250 0.56250
## [3,] 0.56250 0.56250 1.03125 0.56250
## [4,] 0.56250 0.56250 0.56250 1.03125
```

```
#and their parents
ARM[92:95,c(ped[92,2],ped[92,3])]
```

```
##      [,1]  [,2]
## [1,] 0.59375 0.53125
## [2,] 0.59375 0.53125
## [3,] 0.59375 0.53125
## [4,] 0.59375 0.53125
```

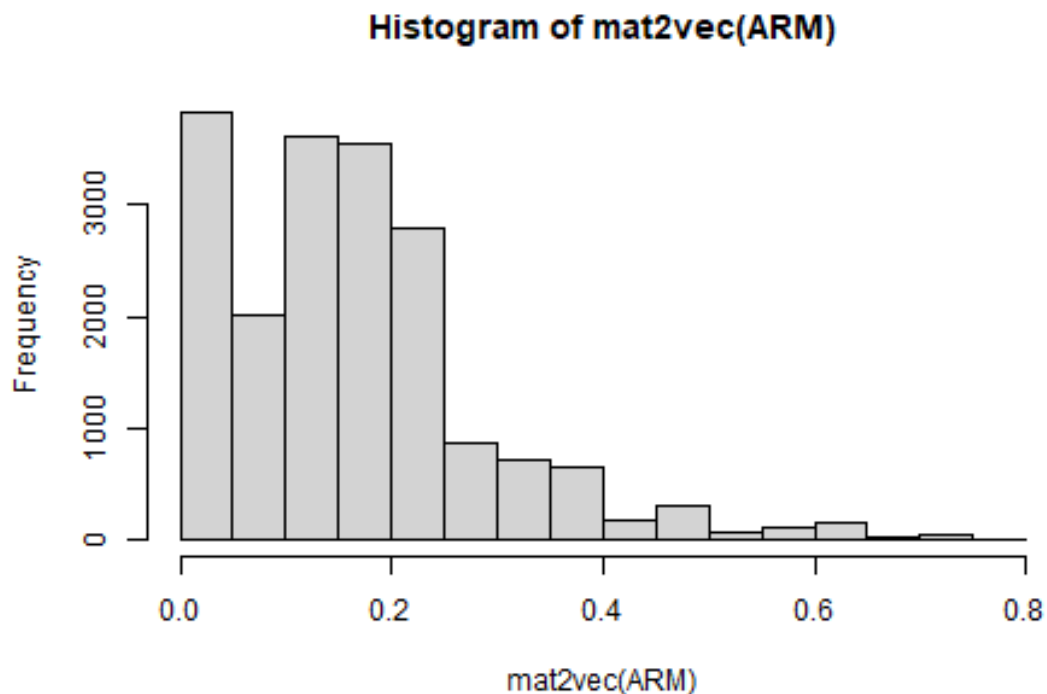
```
#their parents are related
ARM[c(ped[92,2],ped[92,3]),c(ped[92,2],ped[92,3])]
```

```
##      [,1]  [,2]
## [1,] 1.1250 0.0625
## [2,] 0.0625 1.0000
```

relatedness between sibs is 0.5 (as well as with their parents).  
ind 92-95 are more related than 0.5 because they share a common ancestor

- Plot a histogram of the relatedness using only off diagonal elements. The function `hierfstat::mat2vec` will be useful. Comments? Transform the relatedness matrix into a kinship and inbreeding matrix using `hierfstat::GRM2kinship`. Plot a heatmap of resulting matrix using `image` and interpret

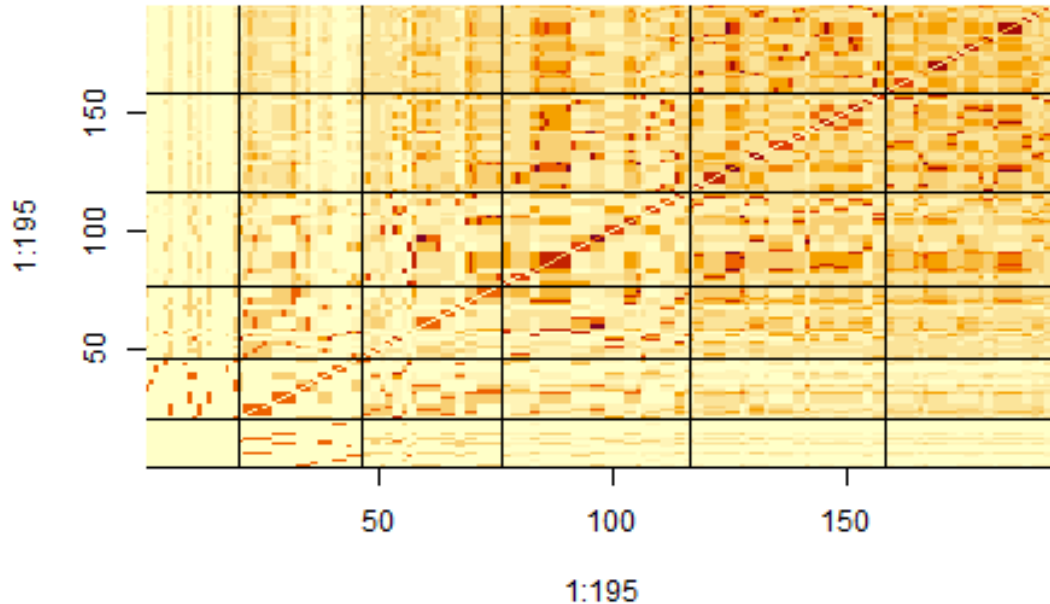
```
hist(mat2vec(ARM))
```



```
ped.kin<-hierfstat::grm2kinship(ARM)
image(1:195,1:195,ped.kin)
```

```
#add vertical and horizontal lines between generations
```

```
genlim<-c(21,47,77,117,159)
abline(h=genlim-.5,v=genlim-.5)
```



The histogram shows a wide spread of relatedness values, with pics at 0 and 0.2, and few very highly related individuals. This would be uncommon for a natural population, but here we have a 5 generations pedigree with a lot of relatedness. The heatmap shows this very well with relatedness increasing (color getting redder) as we move to the right and the top. We also see well the sibships blocs and their parents and grand parents showing as red square in off diagonal

## Kinship and inbreeding from markers, compared to pedigrees

2. We have generated genotypes dosage from these individuals. First download, then load this file using `readRDS`, and compute
  - Allele sharing kinship using `hierfstat::beta.dosage`
  - Standard GRM (average of ratios) as computed in e.g. GCTA using `gaston::GRM`;
  - Standard GRM (ratio of averages) using `JGTeach::Kc0`. Compare each to the expected values obtained from the pedigree and discuss.

```
genoped<-readRDS("geno_mono.RDS")
genoped.Kas<-hierfstat::beta.dosage(genoped,inb=FALSE,Mb=TRUE)
genoped.M<-with(genoped.Kas,betas*(1-MB)+MB)
genoped.Kc0<-JGTeach::Kc0(genoped.M,matching=TRUE)
bed.genoped<-gaston::as.bed.matrix(genoped)
genoped.std.GRM<-gaston::GRM(bed.genoped,autosome.only=FALSE)
```

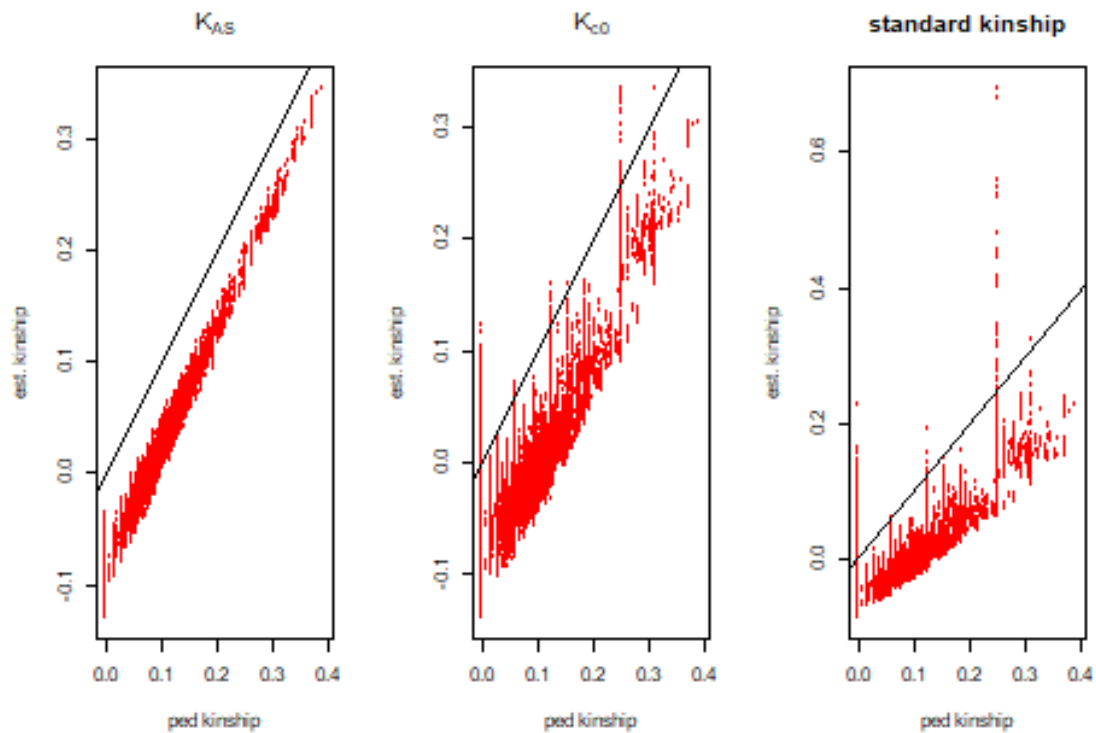
```

par(mfrow=c(1,3))
plot(mat2vec(ARM)/2,mat2vec(genoped.Kas$betas),
     cex=0.5,col="red",main=expression(K[AS]),
     xlab="ped kinship",ylab="est. kinship")
abline(c(0,1))
plot(mat2vec(ARM)/2,mat2vec(genoped.Kc0),
     cex=0.5,col="red",main=expression(K[c0]),
     xlab="ped kinship",ylab="est. kinship")
abline(c(0,1))

# gaston::GRM reports relatedness rather than kinship, hence the /2

plot(mat2vec(ARM)/2,mat2vec(genoped.std.GRM)/2,
     cex=0.5,col="red",main="standard kinship",
     xlab="ped kinship",ylab="est. kinship")
abline(c(0,1))

```



```

par(mfrow=c(1,1))

```

the three plots show a relation between markers and pedigree kinship, but of different quality. KAS shows a very tight, although downwardly biased relation. Much more scatter is seen for Kc0, and even more (and a different slope) for standard.

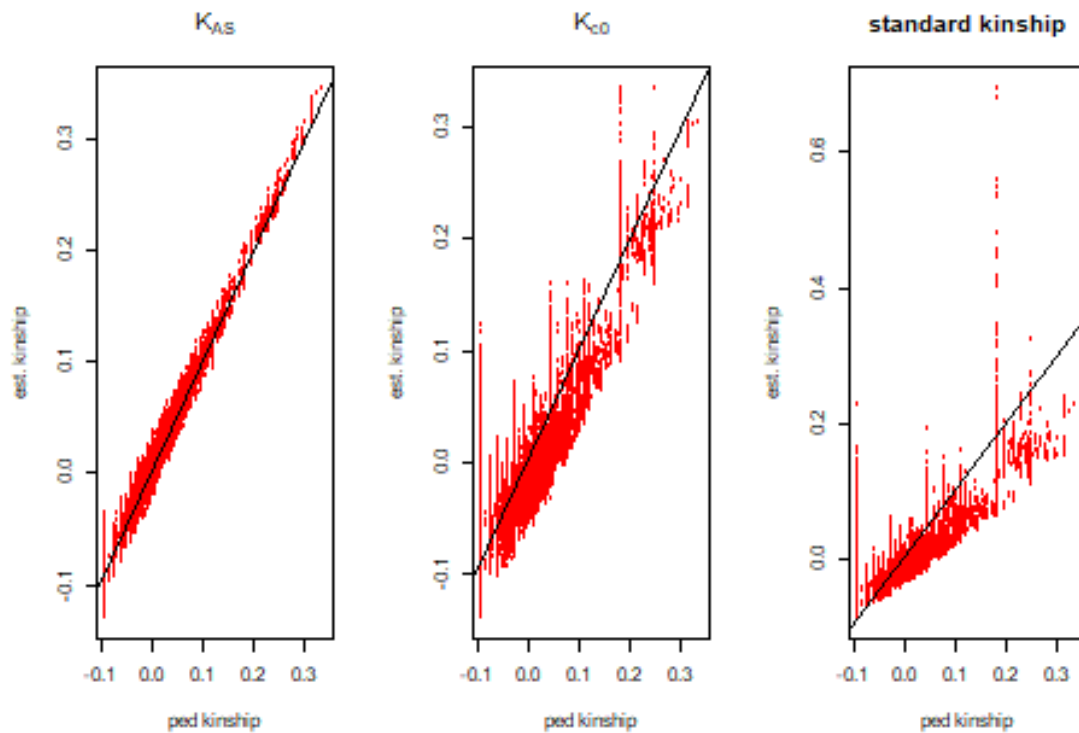
- Redo these comparisons after having “rescaled” the pedigree kinship such that mean expected kinship is 0 and discuss

```

#we divide by two because we want kinships rather than relatedness
mARM<-mean(mat2vec(ARM)/2)
ARMc<-(ARM/2-mARM)/(1-mARM)

par(mfrow=c(1,3))
plot(mat2vec(ARMc),mat2vec(genoped.Kas$betas),
     cex=0.5,col="red",main=expression(K[AS]),
     xlab="ped kinship",ylab="est. kinship")
abline(c(0,1))
plot(mat2vec(ARMc),mat2vec(genoped.Kc0),
     cex=0.5,col="red",main=expression(K[c0]),
     xlab="ped kinship",ylab="est. kinship")
abline(c(0,1))
plot(mat2vec(ARMc),mat2vec(genoped.std.GRM)/2,
     cex=0.5,col="red",main="standard kinship",
     xlab="ped kinship",ylab="est. kinship")
abline(c(0,1))

```



```
par(mfrow=c(1,1))
```

The rescaling is just made through the realisation that the average kinship for KAS is 0, so if we bring pedigree kinship to a mean of 0, we see a perfect fit. More about this can be found in Goudet, Kay Weir 2018 mol ecol

For a discussion of these issues, see Weir and Goudet, 2017 Genetics, Fig. 5 and followings

## kinship and inbreeding in the 1000 genomes

3. We will now compare these three estimators of genomic kinship using the subset of Chromosome 22 we used in previous practical. Start by computing the three estimators :

```
ch22<-read.VCF("chr22_Mb0_20.recode.vcf.gz")
```

```
## ped stats and snps stats have been set.
## 'p' has been set.
## 'mu' and 'sigma' have been set.
```

```
ch22.M<-readRDS("matching.ch22.RDS")
Mb<-mean(mat2vec(ch22.M))
```

```
ch22.Kas<-(ch22.M-Mb)/(1-Mb)
ch22.Kc0<-JGTeach::Kc0(ch22.M,matching=TRUE)
ch22.std.GRM<-gaston::GRM(ch22)
```

- Produce a heat map for each (this may take some times, you might want to save the results to an external file for future inspection, hence the commented call to `png`), after having removed the main diagonal (which contains the self kinships), and arranging samples according to `super_population` first, and then population.

```
#extract self kinship and convert it to inbreeding
```

```
ch22.Kas.inb<-diag(ch22.Kas)*2-1/2
diag(ch22.Kas)<-NA
ch22.Kc0.inb<-diag(ch22.Kc0)*2-1/2
diag(ch22.Kc0)<-NA
ch22.std.Inb<-diag(ch22.std.GRM)-1
diag(ch22.std.GRM)<-NA
```

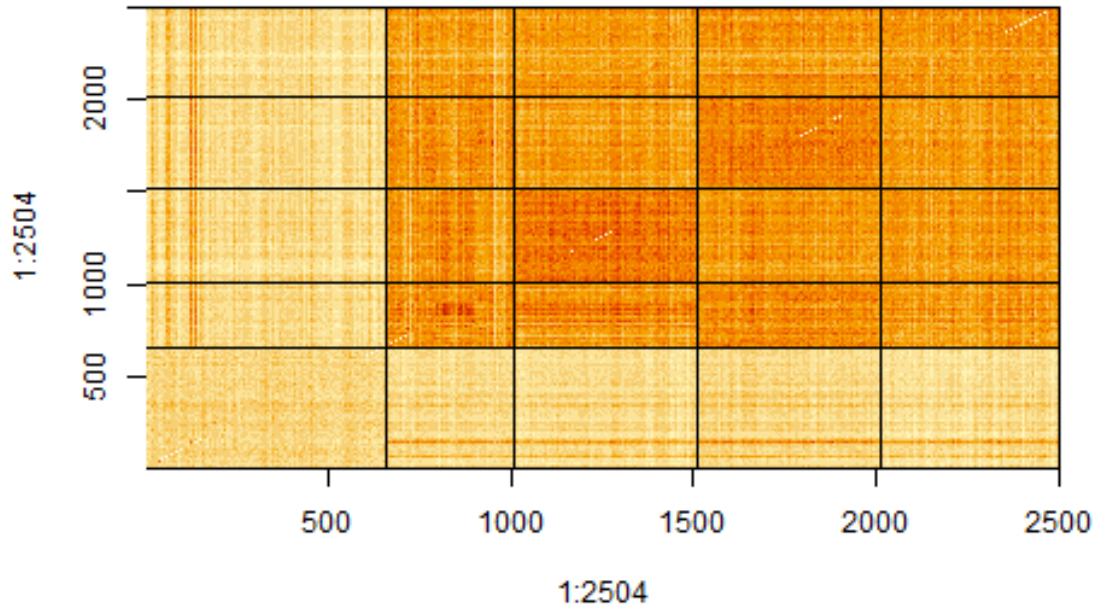
```
samp.desc<-read.table("ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/release/20130502/integrated_call
```

```
#to order samples
```

```
ospp<-with(samp.desc,order(super_pop,pop))
lsp<-cumsum(table(samp.desc$super_pop))
```

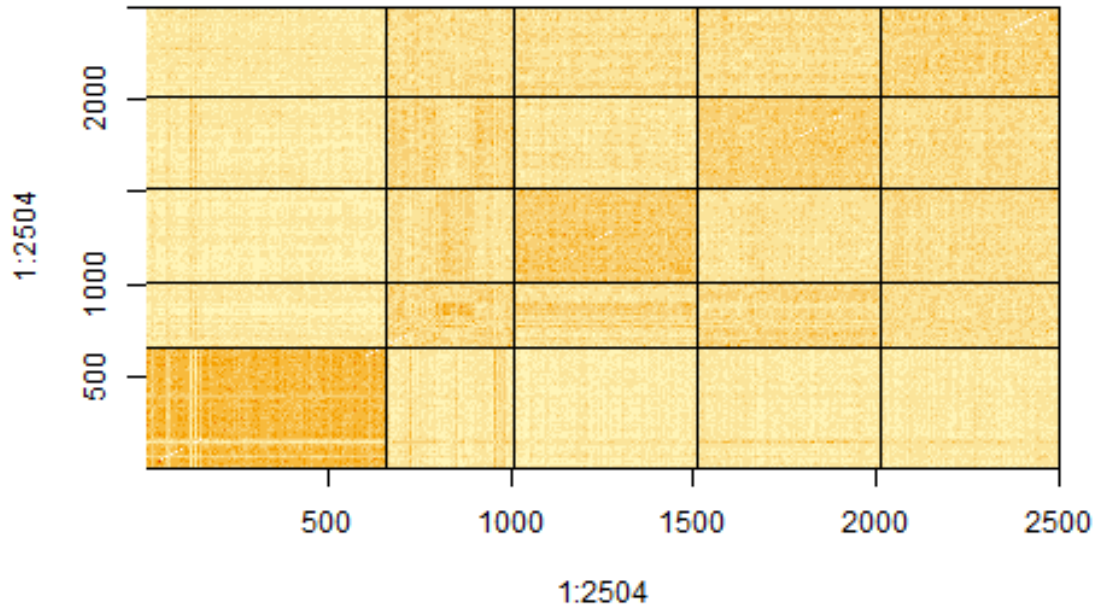
```
#png("1kg_ch22chunck_Kas.png")
```

```
image(1:2504,1:2504,ch22.Kas[ospp,ospp])
abline(h=lsp,v=lsp)
```

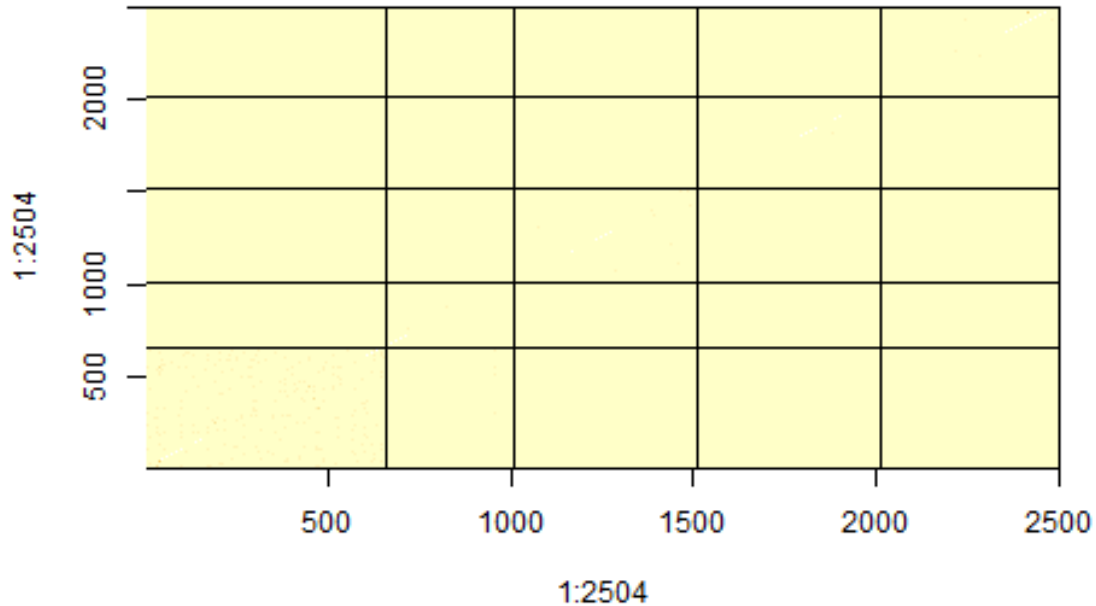


```
#dev.off()  
#png("1kg_ch22chunk_Kc0.png")  
image(1:2504,1:2504,ch22.Kc0[ospp,ospp])  
abline(h=lsp,v=lsp)
```





```
#dev.off()  
#png("1kg_ch22chunck_Kstdk.png")  
image(1:2504,1:2504,ch22.std.GRM[ospp,ospp]/2)  
abline(h=lsp,v=lsp)
```



```
#dev.off()
```

- Discuss the resulting heatmaps, in the light of what you know about human population demographic history.

We are now using a population with no pedigree, so no predicted values.

The three heatmaps look very different.

Populations are arranged by continent, in the order Africa, America, East Asia, Europe and South Asia from left to right.

Along the main diagonal are kinship within continents.

For KAS, we see two blocs, Africa on one side with very light color (low values of kinship), and the rest of the world on the other side, with more red, meaning higher kinship.

EAS shows highest kinship values, and AMR the most heterogeneity.

Between continent, within the non African block, we see high values too, compared to AFR-AFR or AFR-rest of the world

For KCO, the picture is VERY different: pairs within africa are more related than pairs from anywhere else in the world, both within and between continents. This is surprising, as kinship based on markers should reflect how similar the genomes are, and we've seen that african genomes are the most heterozygous, and therefore the most likely to differ at any position

Std is even worse, showing no difference among continents and being driven by specific pairs with high values, not showing up here.

Standard is VERY sensitive to rare alleles,  
if we were to filter on higher maf,  
the picture will look like kc0

What we are seeing is not specific to this chunk of chr 22,  
roughly the same picture emerges from the whole genome.

Overall then, for kinship, both for pedigree and for  
markers estimates, we should use KAS