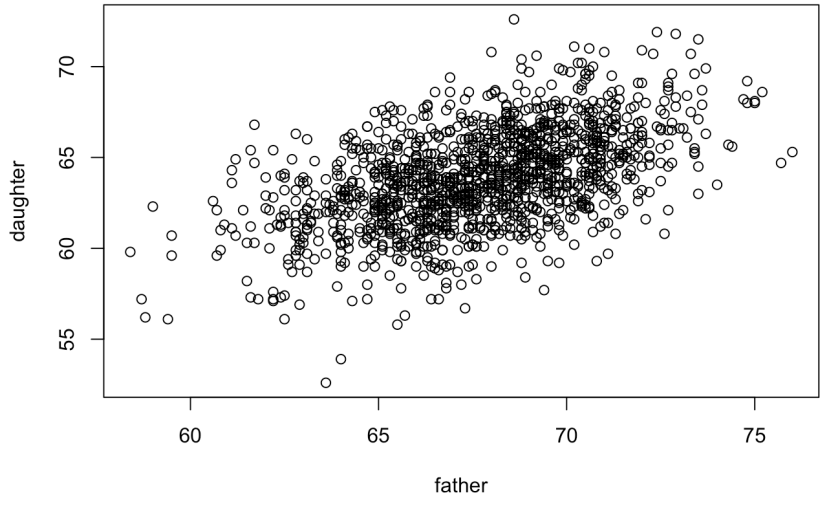
Statistics for Laboratory Scientists (140.615)

Principal components

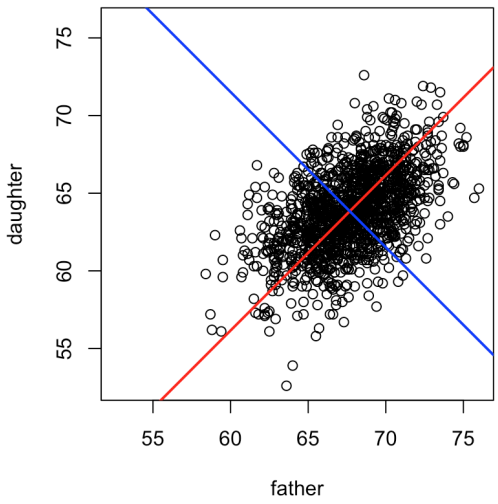
Example 1 - father/daughter data

```
pear <- read.csv("http://biostat.jhsph.edu/~iruczins/teaching/140.615/data/father_daughter.csv",
                 row.names=NULL,skip=3)
plot(pear)
```



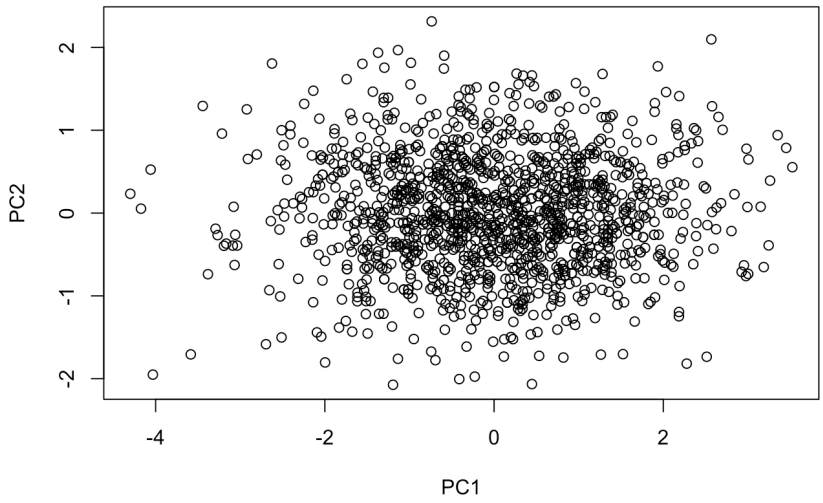
Calculate the principal components, and add them to the scatter plot.

```
pear.pca <- prcomp(pear,retx=TRUE,center=TRUE,scale.=TRUE)

int <- pear.pca$center
rot <- pear.pca$rotation

b1 <- rot[2,1]/rot[1,1]
a1 <- int[2]-b1*int[1]
b2 <- rot[2,2]/rot[1,2]
a2 <- int[2]-b2*int[1]

par(pty="s")
r <- range(pear)
plot(pear,xlim=r,ylim=r)
abline(a1,b1,col="red",lwd=2)
abline(a2,b2,col="blue",lwd=2)
```

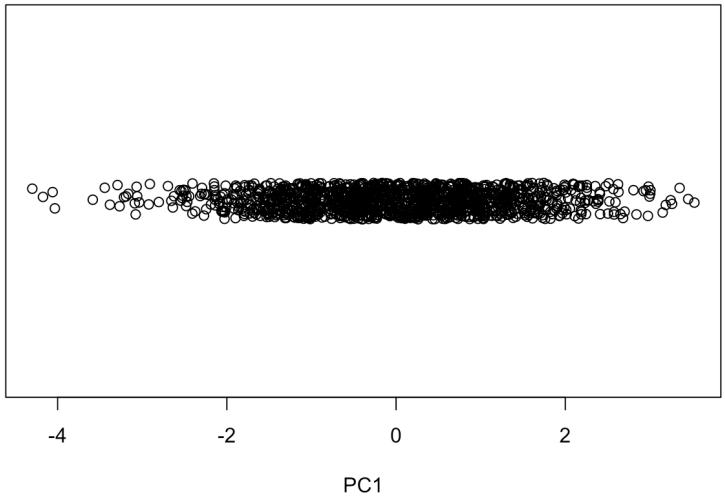


Plot the “principal components” (really: the rotated data).

```
par(pty="m")
plot(pear.pca$x)
```




```
# First principal component only
plot(pear.pca$x[,1],runif(nrow(pear.pca$x),-0.1,0.1),ylim=c(-1,1),xlab="PC1",ylab="",yaxt="n")
```



Some technical information.

```
summary(pear.pca)
```

```
## Importance of components:
##                           PC1    PC2
## Standard deviation     1.2318 0.6947
## Proportion of Variance 0.7587 0.2413
## Cumulative Proportion  0.7587 1.0000
```

```
pear.pca$rotation
```

```
##                PC1        PC2
## father   0.7071068 -0.7071068
## daughter 0.7071068  0.7071068
```

```
pear.pca$sdev
```

```
## [1] 1.2318248 0.6946997
```

```
pear.pca$center
```

```
##   father daughter
## 67.67871 63.83823
```

Example 2 - MVN in 10 dimensions

A function to simulate multivariate normal data.

```
myrmvn <- function(mu,sigma,hm=1,...){
  n=length(mu)
  if(sum((dim(sigma)-rep(n,2))^2)!=0)
    stop("Check the dimensions of mu and sigma!")
  if(det(sigma)==0) stop("The covariance matrix is singular!")
  a=t(chol(sigma))
  z=matrix(rnorm(n*hm),nrow=n)
  y=t(a%*%z+mu)
  return(y)
}
```

Simulate data from a multivariate normal distribution.

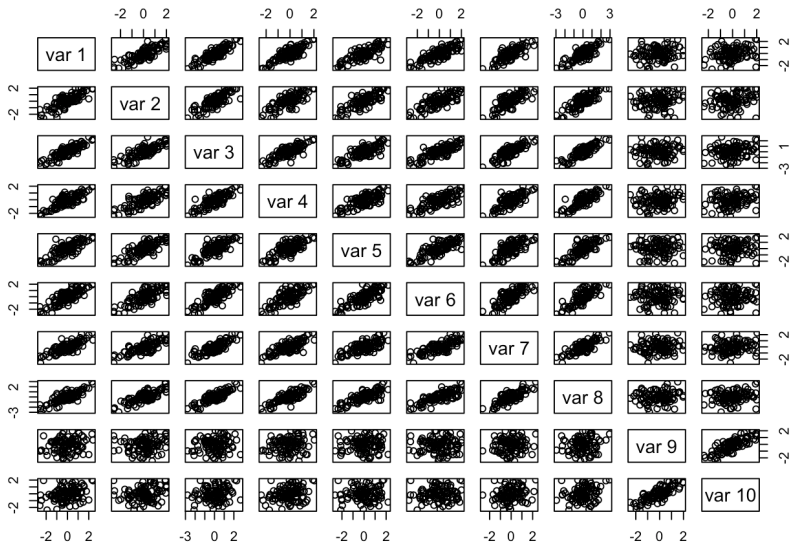
```
mu <- rep(0,10)
sigma <- diag(0.2,10)+0.8
sigma[1:8,9:10] <- 0.2
sigma[9:10,1:8] <- 0.2
mu
```

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

```
sigma
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]  1.0  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.2   0.2
##  [2,]  0.8  1.0  0.8  0.8  0.8  0.8  0.8  0.8  0.2   0.2
##  [3,]  0.8  0.8  1.0  0.8  0.8  0.8  0.8  0.8  0.2   0.2
##  [4,]  0.8  0.8  0.8  1.0  0.8  0.8  0.8  0.8  0.2   0.2
##  [5,]  0.8  0.8  0.8  0.8  1.0  0.8  0.8  0.8  0.2   0.2
##  [6,]  0.8  0.8  0.8  0.8  0.8  1.0  0.8  0.8  0.2   0.2
##  [7,]  0.8  0.8  0.8  0.8  0.8  0.8  1.0  0.8  0.2   0.2
##  [8,]  0.8  0.8  0.8  0.8  0.8  0.8  0.8  1.0  0.2   0.2
##  [9,]  0.2  0.2  0.2  0.2  0.2  0.2  0.2  0.2  1.0   0.8
## [10,]  0.2  0.2  0.2  0.2  0.2  0.2  0.2  0.2  0.8   1.0
```

```
set.seed(1)
n <- 100
dat <- myrmvn(mu,sigma,hm=n)

pairs(dat)
```

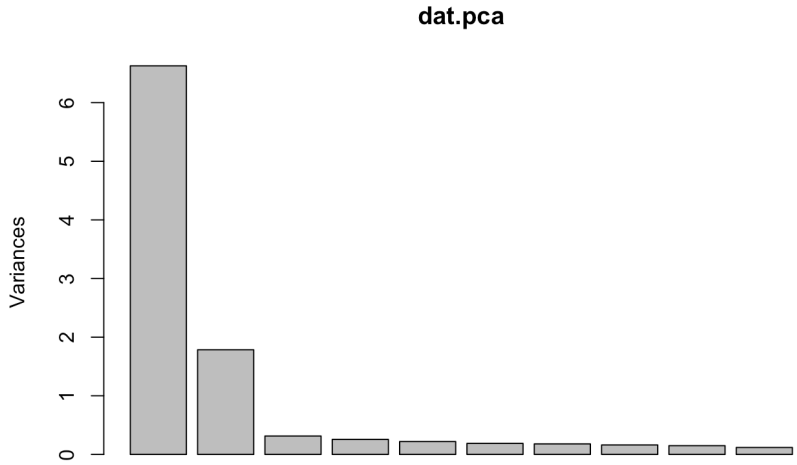


PCA

```
dat.pca <- prcomp(dat,retx=TRUE,center=TRUE,scale.=TRUE)
summary(dat.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5    PC6     PC7
## Standard deviation     2.5745 1.3361 0.5604 0.50644 0.46858 0.4336 0.42375
## Proportion of Variance 0.6628 0.1785 0.0314 0.02565 0.02196 0.0188 0.01796
## Cumulative Proportion  0.6628 0.8413 0.8727 0.89834 0.92030 0.9391 0.95705
##                            PC8     PC9    PC10
## Standard deviation     0.40214 0.38626 0.34434
## Proportion of Variance 0.01617 0.01492 0.01186
## Cumulative Proportion  0.97322 0.98814 1.00000
```

```
plot(dat.pca)
```



Example 3 - genetic background

A function to simulate some SNPs.

```
my.snp.pc.data=function(n,maf){
  hm=length(maf)
  z=matrix(ncol=hm,nrow=n)
  for(j in 1:hm){
    p=1-maf[j]
    mp=c(p^2,2*p*(1-p),(1-p)^2)
    z[,j]=apply(rmultinom(n,1,mp),2,order)[3,]-1
  }
  return(z)
}
```

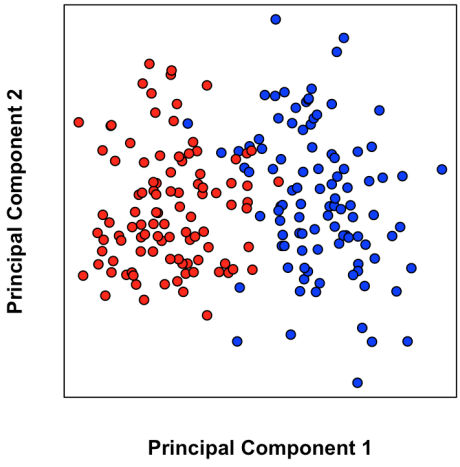
Simulate two populations, 100 subjects each, and 50 SNPs.

```
n1 <- n2 <- 100
ns <- 50

set.seed(1)
maf1 <- runif(ns,0.1,0.5)
maf2 <- runif(ns,0.1,0.5)
z1 <- my.snp.pc.data(n1,maf1)
z2 <- my.snp.pc.data(n2,maf2)
z <- data.frame(rbind(z1,z2))
head(z)
```

```
##   X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 X11 X12 X13 X14 X15 X16 X17 X18 X19 X20
## 1  1  0  0  2  0  0  0  1  2   0   0   0   0   0   0   1   2   1   0   1
## 2  0  0  0  0  1  0  2  0  1   0   0   0   2   0   1   0   2   1   0   0
## 3  1  0  1  0  1  2  1  1  1   1   0   0   0   1   1   0   1   1   1   1
## 4  0  2  1  0  2  1  0  0  1   0   0   1   1   1   1   0   1   1   0   2
## 5  0  0  0  0  1  1  2  0  0   0   1   1   0   1   0   0   1   0   0   0
## 6  0  0  1  1  0  1  0  1  1   0   0   0   2   0   1   0   2   0   1   1
##   X21 X22 X23 X24 X25 X26 X27 X28 X29 X30 X31 X32 X33 X34 X35 X36 X37 X38
## 1   1   0   0   0   1   0   0   1   2   1   0   1   0   0   2   0   0   0
## 2   2   1   1   0   0   0   0   0   1   0   0   0   0   0   1   0   1   0
## 3   1   0   1   2   0   1   1   1   2   1   0   2   0   1   1   1   0   0
## 4   1   0   2   0   0   0   0   1   0   0   0   2   0   0   1   1   2   0
## 5   2   1   0   0   1   0   1   0   0   1   1   0   0   0   1   0   1   0
## 6   0   0   0   0   0   1   0   0   1   0   1   2   1   1   1   1   1   0
##   X39 X40 X41 X42 X43 X44 X45 X46 X47 X48 X49 X50
## 1   0   1   0   0   2   0   1   2   1   0   0   1
## 2   0   0   1   1   0   0   1   1   0   1   0   1
## 3   2   0   0   1   0   0   0   0   1   0   0   1
## 4   2   0   1   0   0   0   0   0   0   0   1   1
## 5   1   0   2   1   1   0   1   0   0   1   1   0
## 6   0   1   0   0   1   1   1   1   1   0   1   0
```

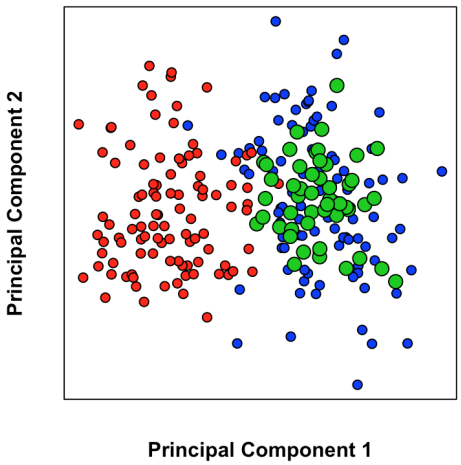
PCA

```
snp.pca <- prcomp(z,retx=TRUE,center=TRUE,scale.=TRUE)
snp.p <- predict(snp.pca)
par(mfrow=c(1,1),font.lab=2,mgp=c(1.5,1,0),pty="s")
plot(snp.p[,1:2],pch=21,bg=rep(c("blue","red"),c(n1,n2)),xaxt="n",yaxt="n",
     xlab="Principal Component 1",ylab="Principal Component 2")
```



Simulate a new set of 50 samples from population 1.

```
n3 <- 50
z3 <- data.frame(my.snp.pc.data(n3,maf1))
p3 <- predict(snp.pca,z3)

par(mfrow=c(1,1),font.lab=2,mgp=c(1.5,1,0),pty="s")
plot(snp.p[,1:2],pch=21,bg=rep(c("blue","red"),c(n1,n2)),xaxt="n",yaxt="n",
     xlab="Principal Component 1",ylab="Principal Component 2")
points(p3[,1:2],pch=21,bg="green3",cex=1.5)
```



End of code