

Singular Value Decomposition

or simply SVD ([Wikipedia on SVD](#)).

In brief SVD is a factorization of a rectangular data matrix. SVD have some similarities to PCA.

Here is some R code for doing a SVD.

Say, you have rectangular matrix 'ei' (e.g. genes on the rows, samples in the columns).

```
m <- ncol(ei)

X<- ei - rowMeans(ei) # X is zero centered
# or
X<-t(scale(t(ei))) # X is zero centered and with unit variance

svd <- svd(X)
V <- svd$v
S <- matrix(0, m, m)
diag(S)<-svd$d
```

Now the object 'svd' holds the results of the svd, and have the slots d,u and v [svd is a 'list', so the slots can be accessed like this: svd\$d].

One may want to do a visualization like this:

```
sample_labels<-colnames(ei) # this may be manually set
par(mfrow=c(1,2))
image( S%%t(V) , main="Singular Value Decomposition",
xaxt="n", yaxt="n", xlab="Components")
plot(V[,1], V[,2], type="n", main="SVD", xlab="First component",
ylab="Second component")
text(V[,1], V[,2], sample_labels)
par(mfrow=c(1,1))
```

To visualize other components than 1 and 2:

```
compX<-3
compY<-4
plot(V[,compX], V[,compY], type="n", main="SVD",
xlab=paste("component", compX), ylab=paste("component", compY))
text(V[,compX], V[,compY], sample_labels)
```

To get the contributed from the real genes to a given component (or eigen-gene) e.g. the first component

```
component<-1
gene_weights<-abs(svd$u[,component])
```

Note, this is most meaningful for 'x' scaled to unit variance.