STAT 441: Lecture 25
SVD
(or global warming)

Singular value decomposition generalizes the eigenvalue decomposition

Hence it can be used with similar objectives

SVD

Eigenvalue decomposition: $A = ULU^\top$

where U is orthogonal ($U^\top U = UU^\top = I$)

and L is diagonal (eigenvalues)

A must be a square matrix for eigenvalue decomposition

Singular value decomposition: $A = U\Lambda V^\top$

where U and V are orthogonal and Λ is $k \times m$ diagonal with diagonal elements

$\lambda_i \geqslant 0$ (singular values)

A may be any $k \times m$ matrix

```
> A = cbind(c(1,2,3),c(2,5,4))
> A
     [,1] [,2]
[1,]    1    2
[2,]    2    5
[3,]    3    4
```

Try it in R

```
> sa=svd(A,nu=3)
> sa
$d
[1] 7.6203733 0.9643188

$u
           [,1]        [,2]       [,3]
[1,] -0.2932528 -0.08121183 -0.9525793
[2,] -0.7017514 -0.65838502  0.2721655
[3,] -0.6492670  0.74828724  0.1360828

$v
           [,1]       [,2]
[1,] -0.4782649  0.8782156
[2,] -0.8782156 -0.4782649
```

And test it

```
> t(sa$u) %*% sa$u
              [,1]          [,2]          [,3]
[1,]  1.000000e-00  0.000000e+00 -9.714451e-17
[2,]  0.000000e+00  1.000000e+00 -1.387779e-16
[3,] -9.714451e-17 -1.387779e-16  1.000000e+00
> t(sa$v) %*% sa$v
     [,1] [,2]
[1,]    1    0
[2,]    0    1
> sa$v %*% t(sa$v)
     [,1] [,2]
[1,]    1    0
[2,]    0    1
> sa$u %*% diag(sa$d) %*% t(sa$v)
Error in sa$u %*% diag(sa$d) : non-conformable arguments
> sa$u %*% rbind(diag(sa$d),c(0,0)) %*% t(sa$v)
     [,1] [,2]
[1,]    1    2
[2,]    2    5
[3,]    3    4
```

Economy class

There are two versions of SVD: the one above, and the other, "economy" version, in which

If $k \geqslant m$: V is as above, Λ as above, but *square*, $m \times m$

and only first m columns of U are taken - then U has orthonormal columns, $U^\top U = I$, but is not orthogonal, because $UU^\top$ may differ from I

If $k \leqslant m$, then other way round: Λ is $k \times k$, U is orthogonal (and square) and V has orthonormal columns, $V^\top V = I$

```
> sa=svd(A)
> sa
$d
[1] 7.6203733 0.9643188
$u
           [,1]        [,2]
[1,] -0.2932528 -0.08121183
[2,] -0.7017514 -0.65838502
[3,] -0.6492670  0.74828724
$v
           [,1]       [,2]
[1,] -0.4782649  0.8782156
[2,] -0.8782156 -0.4782649
```

Test of economy version

```
> sa$u %*% diag(sa$d) %*% t(sa$v)
     [,1] [,2]
[1,]    1    2
[2,]    2    5
[3,]    3    4
> sa$u %*% t(sa$u)
          [,1]        [,2]        [,3]
[1,] 0.0925926  0.25925926  0.12962963
[2,] 0.2592593  0.92592593 -0.03703704
[3,] 0.1296296 -0.03703704  0.98148148
> t(sa$u) %*% sa$u
     [,1] [,2]
[1,]    1    0
[2,]    0    1
> sa$v %*% t(sa$v)
     [,1] [,2]
[1,]    1    0
[2,]    0    1
> t(sa$v) %*% sa$v
     [,1] [,2]
[1,]    1    0
[2,]    0    1
```

Use of SVD in computing principal components

We just do SVD of the *centered* (column means subtracted, so that we can consider them zero now) data matrix X:

$$X = U\Lambda V^{\top}$$

$$\frac{1}{n-1}X^{\top}X = \frac{1}{n-1}V\Lambda U^{\top}U\Lambda V^{\top}$$

$$= \frac{1}{n-1}V\Lambda\Lambda V^{\top} = V\left(\frac{1}{n-1}\Lambda^{2}\right)V^{\top} = VLV^{\top}$$

hence eigenvalues are squares of singular values divided by $n-1$ and eigenvectors are in V; the principal components are $U\Lambda = U\Lambda V^{\top}V = XV$

```
> sa=svd(sweep(USArrests,2,apply(USArrests,2,mean)))
> sa$d
[1] 586.12680  99.48681  45.42598  17.37953
> sqrt(sa$d^2/49)
[1] 83.732400 14.212402  6.489426  2.482790
> prcomp(USArrests)
Standard deviations:
[1] 83.732400 14.212402  6.489426  2.482790
...
```