STAT 535

MATRIX DECOMPOSITIONS

Least-Squares Fitting

$$\hat{\beta} = \text{argmin}_\beta \|y - X\beta\|^2$$

Solve

$$\frac{\partial}{\partial \beta}(y - X\beta)'(y - X\beta) = 0$$

which reduces to

$$(X'X)\beta = X'y$$

or perhaps

$$\beta = (X'X)^{-1}X'y$$

How to Implement?

```
meth1_function(y,x) {
  solve( t(x) %*% x) %*% t(x) %*% y }
```

Efficient???

Choleski Decomposition

$$A = R'R$$

A is square, symmetric, positive definite
R is upper triangular, positive diagonal entries

```
meth2_function(y,x) {
  r <- chol( t(x) %*% x)
  tmp <- backsolve(r, t(x) %*% y, transpose=T)
  backsolve(r, tmp, transpose=F) }
```

Paul's Dumb Approach to R Timing

```
datetosec <- function(dt) {
           as.numeric(substring(dt, 24-6, 24-5))  +
    60  * as.numeric(substring(dt, 24-9, 24-8))  +
  60^2  * as.numeric(substring(dt, 24-12, 24-11))  }

timings_matrix(0,20,2)
for (i in 1:20) {
  for (j in sample(1:2)) {
    sec1_datetosec(date())
    if (j==1) { meth1(y,x) } else { meth2(y,x) }
    sec2_datetosec(date())
    timings[i,j]_sec2-sec1 } }
```



```
> n <- 2000; p <- 200
> x <- cbind(1 , matrix(rnorm(n*(p-1)), n, p-1) )
> y <- x %*% seq(from=-1, to=1, length=p) + rnorm(n)

> timings
      [,1] [,2]
 [1,]    9    6
 [2,]    9    5
 [3,]   10    5
      ........
[19,]    8    6
[20,]    9    6

> apply(timings,2,mean)
[1] 8.95 5.50
```


QR Decomposition

$$A = QR$$

A is $n \times p$ ($n \geq p$, full rank)
Q is $n \times p$ with orthogonal columns
R is $p \times p$ upper-triangular

```
meth3_function(y,x) {
  tmp <- qr(x)
  qr.coef(tmp, y) }
```

Connection to Choleski!

Singular Value Decomposition

$$A = UDV'$$

A is $n \times p$
U is $n \times p$, orthogonal columns
D is diagonal
V is $p \times p$, orthogonal

```
meth4_function(y,x) {
  tmp <- svd(x)
  ### note minor 'trick' in line below
  tmp$v %*% ( (1/ tmp$d) * t(tmp$u) ) %*% y }
```

More Timing Comparisons

$n = 2000$, $p = 200$

```
> timings
     CHOL   QR  SVD
[1,]    6    4   22
[2,]    6    4   23
[3,]    6    3   22
[4,]    5    4   22
[5,]    5    4   22
```


Computational Effort

Solving $p \times p$ system is $O(p^3)$ computing effort.

A $2^{(1/3)} \approx 1.26$ increase in p doubles the computation time?

With $n = 1000$, and QR method ...

```
> timings
     p=400 p=504 p=635
[1,]     7     9    14
[2,]     6    10    14
[3,]     6     9    15
[4,]     6     9    15
[5,]     6     9    14
```


Numerical Accuracy

Numerical Accuracy?

```
> summary(meth1(y,x)-meth2(y,x))
 Min.   :-4.441e-15
 ...
 Max.   : 4.996e-15
```

Condition Numbers???

Close to Singular Matrices

```
> eps_0.0001
> x_matrix(rnorm(200),100,2)
> x_cbind(x, 0.6*x[,1]-0.3*x[,2]+rnorm(100,sd=eps))
> y_x%*%c(1,1,.5)+rnorm(100)

> meth1(y,x)
Error in solve.default(t(x) %*% x) :
singular matrix 'a' in solve

> meth2(y,x)
[1]  116.48945  -56.66959 -191.81850
```

Same answer with meth3() and meth4().

Other Issues

Sparse matrix methods (banded, for instance)

Updating methods

X is $n \times p$, but $Rank(X) < p$.
Generalized Inverse?

X is $n \times p$, but $n < p$. Bioinformatics!