LEAST SQUARES AND FLOATING POINT PRECISION

NUMEDIG SEPTEMBER 2013

Created by Andreas Noack Jensen

THE LEAST SQUARES PROBLEM

Consider the problem

$$\min_b \|y - Xb\|_2$$

If X has linearly independent columns

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

so what is the problem?

HOW IS $\hat{b}$ CALCULATED?

```
bhat=inv(X'*X)*(X'*y)
```

or

```
bhat=(X'*X)\(X'*y)
```

or

```
bhat=X\y
```

or something else... What is the difference?

ONE STEP BACK!

Solve the normal equations, $X'X\hat{b} = X'y$

Linear system of equations, $Ax = b$

Never invert a matrix

Well...

SOLVING LINEAR EQUATIONS

First year math taught us to use row operations to reduce problem to echelon form. MATLAB does essentially the same thing:

The problem is transformed to triangular form and solved by backward recursion

$$\begin{pmatrix} a_{11} & a_{12} \\ 0 & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

then $x_2 = a_{22}^{-1} b_2$ and $x_1 = a_{11}^{-1}(b_1 - a_{12} x_2)$

$A = LU$

A is factorised into a *L*ower (unit) triangular and an *U*pper triangular matrix. Consider $Ax = b$ for $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$. The factorisation is

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ l_{21} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{pmatrix}$$

Now, $LUx = b$ which can be solved as the two triangular systems $Ly = b$ and $y = Ux$

This is what \ does in MATLAB and Julia when A is square

$$A = QR$$

A is factorized into a orthogonal matrix Q, i.e. $Q'Q = I$ and an upper triangular matrix R. Hence we can multiply

$$QRx = b$$

with Q' from the left and solve

$$Rx = Q'b$$

which is triangular.

CHOLESKY

If A is positive definite then $A = B'B$ for some B. If we factorise $B = QR$ then

$$A = R'Q'QR = R'R$$

with R triangular. This is known as the Cholesky decomposition/factorisation. Linear system

$$R'Rx = b$$

can be solved similarly to LU system by solving

$$R'y = b$$

and

$$y = Rx$$

THE INVERSE

How to calculate the inverse? Finding the inverse is the same as solving

$$Ax = e_i$$

for $i = 1, \ldots, n$ or in matrix form

$$AX = I.$$

Hence, faster to solve $Ax = b$ directly instead of first solving $AX = I$ to get A^{-1} and then multiplying $x = A^{-1}b$

THE PRECISION OF THE INVERSE

Saying: Never calculate the inverse. It is slow and imprecise. We have discussed the first part. What about the second?

The *condition number* measures how output error is affected by input error. Smaller condition number is better. Depends on the choice of norm, but most the induced two norm is used.

Well knwon error bound states that a solution calculated with the inverse doubles the condition number relative to a solution by backward recursion.

Druinsky and Toledo (2012) does not agree and not easy to find substantial differences.

LEAST SQUARES

Consider again the normal equations

$$X'X\hat{b} = X'y.$$

The matrix $X'X$ is positive definite so let us solve this by the Cholesky!

This is NOT what happens in MATLAB when writing `(X'*X)\(X'*y)` bacause MATLAB does not know that $X'X$ is positive definite.

MATLAB uses the LU. No gain in calculating by Cholesky because of temporary array.

NUMERICAL CONSIDERATIONS

The condition number of $X'X$ is the square of the condition number of X. But can we avoid $X'X$?

Yes, we can!.

The QR factorisation can be calculated for rectangular matrices, i.e.

$$X = QR = \begin{pmatrix} Q_0 & Q_1 \end{pmatrix} \begin{pmatrix} R_0 \\ 0 \end{pmatrix} = Q_0 R_0.$$

Notice that Q_0 is rectangular, but still $Q_0' Q_0 = I$.

QR LEAST SQUARES CONTINUED...

Then by the normal equations

$$R_0' R_0 \hat{b} = R_0' Q_0' y$$

or simply

$$R_0 \hat{b} = Q_0' y$$

which is a rectangular system. This is what \ solves in Julia and MATLAB when X is rectangular. A win/lose situation: No squaring but X has a large dimension which $X'X$ does not have. Therefore the last matrix is much faster to factorise. Doornik's point in the documentation for `olsc`.

CONCLUSION

Neat theoretical reasons to prefer `\` over `inv` but not much impact (for the statistician) in practice for two reasons:

1. $X'X$ is (often) small and therefore the inverse is very fast to calculate
2. The numerical problems with squaring X is negligible compared to the statistical problems with almost singularity of $X'X$, i.e. huge standard errors.

NEWTON ITERATION

We want for find x such that $g(x) = a$. Iterate over

$$x_{n+1} = x_i - \frac{g(x_n) - a}{g'(x_n)}$$

When calculating quantiles of a distribution with cdf F and pdf f this is

$$x_{n+1} = x_n - \frac{F(x_n) - a}{f(x_n)}$$