Lecture 12

A few other notes about variable elimination without Pivoting
* Elements of U and L inverse
* Eliminating a block of variables.

Variable elimination with Pivoting!

Then Variable elimination for least squares problems.

- HW problem 1 corrected
- office hours cancelled today → 11:00 am tomorrow

Last time

$A = L_1^{-1} \cdots L_{n-1}^{-1} D U_{n-1}^{-1} \cdots U_1^{-1}$

$L_i = \begin{bmatrix} I & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -d_i/\alpha_i & I \end{bmatrix}$

Goal Now: Show

$L_1^{-1} \cdots L_{n-1}^{-1}$ is Easy to Compute!

Each Term

$L_i = I + \underline{v}_i \underline{e}_i^T$

$\underline{v}_i = \begin{bmatrix} 0 \\ 0 \\ -d_i/\alpha_i \end{bmatrix}$ $\underline{v}_i$ is 0 in rows 1 to i.

L_1 Comes from first step

$A = \begin{bmatrix} \alpha & c^T \\ \underline{d} & R \end{bmatrix}$

$L_1 = \begin{bmatrix} 1 & 0 \\ -\underline{d}/\alpha & I \end{bmatrix}$

$\underline{v}_1 = \begin{bmatrix} 0 \\ -\underline{d}/\alpha \end{bmatrix}$

Now, I claim

$L_i^{-1} = I - \underline{v}_i \underline{e}_i^T$

Pf: $(I + \underline{v}_i \underline{e}_i^T)(I - \underline{v}_i \underline{e}_i^T) =$

$I + \underline{v}_i \underline{e}_i^T - \underline{v}_i \underline{e}_i^T + \underline{v}_i \overset{=0 \text{ by}}{\underline{e}_i^T \underline{v}_i} \underline{e}_i^T$

$L_i^{-1} L_j^{-1}$ is "Easy" if $j > i$

$(I - \underline{v}_i \underline{e}_i^T)(I - \underline{v}_j \underline{e}_j^T) =$

$I - \underline{v}_i \underline{e}_i^T - \underline{v}_j \underline{e}_j^T + \underline{v}_i \underbrace{\underline{e}_i^T \underline{v}_j}_{=0} \underline{e}_j^T =$

$I - \underline{v}_i \underline{e}_i^T - \underline{v}_j \underline{e}_j^T.$

$L_1^{-1} \cdots L_{n-1}^{-1} =$

$I - \underline{v}_1 \underline{e}_1^T - \underline{v}_2 \underline{e}_2^T - \underline{v}_3 \underline{e}_3^T - \cdots - \underline{v}_{n-1} \underline{e}_{n-1}^T$

$L = L_1^{-1} \cdots L_{n-1}^{-1}$ is lower tri

The same holds for U. (but upper tri!)

```
function myreduce_all(A::Matrix)
  A = copy(A) # save a copy
  n = size(A,1)
  L = Matrix(1.0I,n,n)
  U = Matrix(1.0I,n,n)
  d = zeros(n)
  for i=1:n-1
    α = A[i,i]
    d[i] = α
    U[i,i+1:end] = A[i,i+1:end]/α
    L[i+1:end,i] = A[i+1:end,i]/α
    A[i+1:end,i+1:end] -= A[i+1:end,i]*A[i,i+1:end]'/α
  end
  d[n] = A[n,n]
  return L,U,d
end
```

$A = LDU$

This is called an LU factorization

Most folks write

$A = L\underbrace{(DU)}_{U}.$

The idea generalizes

$\underset{m \times m}{A} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} = \begin{bmatrix} \underline{b}_1 \\ \underline{b}_2 \end{bmatrix} \begin{matrix} m \\ n-m \end{matrix}$

$A_1 \underline{x}_1 + A_2 \underline{x}_2 = \underline{b}_1$

$A_3 \underline{x}_1 + A_4 \underline{x}_2 = \underline{b}_2$

if A_1 is non-sing, then

$\underline{x}_1 = A_1^{-1}(\underline{b}_1 - A_2 \underline{x}_2)$

$A_3(A_1^{-1}\underline{b}_1 - A_1^{-1}A_2 \underline{x}_2) + A_4 \underline{x}_2 = \underline{b}_2$

$(A_4 - A_3 A_1^{-1} A_2)\underline{x}_2 =$

$\underline{b}_2 - A_3 A_1^{-1} \underline{b}_1$

↳ is called a Schur Complement.

I can Compute $A_4 - A_3 A_1^{-1} A_2$ by M Steps of the Single Element Elimination.

The final thing to Say.

This generalizes to writing

$\underline{x} = S\underline{y} + T\underline{z}.$

So to eliminate 1, 2, … n

$S = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$ $T = \begin{bmatrix} 0 & \cdots & 0 \\ I \end{bmatrix}$

On to pivoting!

$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 5 \\ 6 \end{bmatrix}$

$1/0$

$0x + y = 5 \Rightarrow$

$x = \left(\frac{5 - y}{0}\right) \Rightarrow$

Breaks $\binom{u}{?}$

To fix this, we Swap rows!

$x + y = 6 \Rightarrow$

$x = 6 - y.$

The idea of pivoting is to re-order the rows of A to make sure we can always Eliminate x_i on Step i.

Result: This is always possible.

Pf:

$\overbrace{L_{i-1} L_{i-2} \cdots L_1}^{\text{non-sing}} A \overbrace{U_1 \cdots U_{i-1}}^{\text{non-sing}} = \begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & d_{i-1} & \\ & & & \boxed{R} \end{bmatrix}$

non-sing.

if I can't find a row of R where $R_{j,1} \neq 0$, then the first col of $R = 0$.

The idea is: find a row j where $R_{j1} \neq 0$ then swap rows i and j. (In R, and all L/U too).

Last thing, we write

$PA = LDU$

↑ Permutation of A's Rows. ↑ factors.

How to do this for least sq?

$\min \|\underline{b} - A\underline{x}\|_2$

A: m×n, b: n-vector

m>n

A = ▯ tall / over constrained

A full rank ⇒

$A\underline{x} = 0 \Leftrightarrow \underline{x} = 0.$

Let's do variable elimination

$\underline{x} = \begin{bmatrix} \gamma \\ \underline{y} \end{bmatrix}$

1st col.

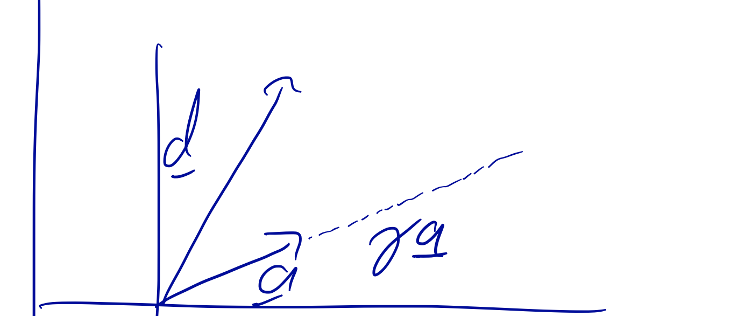
$A = \begin{bmatrix} \underline{a} & C \end{bmatrix}$

$\min \|\underline{b} - \underline{a}\gamma - C\underline{y}\|_2^2$

Assume y is known. Solve for γ.

$\min \|\underbrace{\underline{b} - C\underline{y}}_{\underline{d}} - \underline{a}\gamma\|_2$

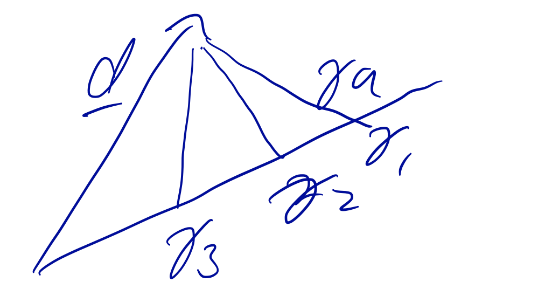
$\min \|\underline{d} - \gamma\underline{a}\|_2$



Least sq. find closest γa to $\underline{d}$

Quiz: What is γ?

Soln:



γ_2 is closer,

so $(\underline{d} - \gamma\underline{a}) \perp \underline{a}$

$\underline{a}^T(\underline{d} - \gamma\underline{a}) = 0$

$\gamma = \frac{\underline{a}^T\underline{d}}{\underline{a}^T\underline{a}}$

or normal Eq. $\min \|\underline{b} - A\underline{x}\|_2$, $A^TA\underline{x} = A^T\underline{b}$

$\underline{a}^T\underline{a}\,\gamma = \underline{a}^T\underline{d}$

$\gamma(y) = \frac{\underline{a}^T(\underline{b} - C\underline{y})}{\underline{a}^T\underline{a}}$

$\min \|\underbrace{\underline{b} - C\underline{y} - \underline{a}\gamma}\|$

$\underline{b} - C\underline{y} - \frac{\underline{a}(\underline{a}^T(\underline{b} - C\underline{y}))}{\underline{a}^T\underline{a}} =$

$\underline{b} - \frac{\underline{a}\underline{a}^T\underline{b}}{\underline{a}^T\underline{a}} - C\underline{y} + \frac{\underline{a}\underline{a}^T}{\underline{a}^T\underline{a}}C\underline{y} =$

$\left(I - \frac{\underline{a}\underline{a}^T}{\underline{a}^T\underline{a}}\right)\underline{b} - \left(I - \frac{\underline{a}\underline{a}^T}{\underline{a}^T\underline{a}}\right)C\underline{y}$

min $\|b - Ax\|_2$ m×n

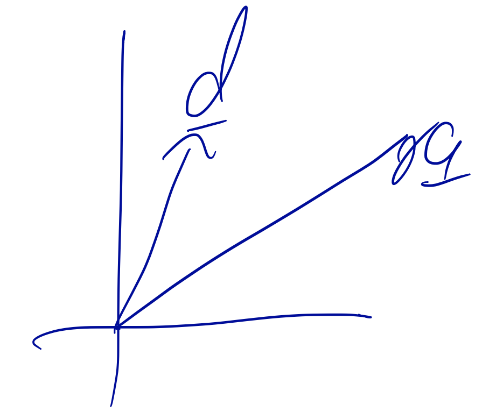
$A = [\underline{a} \quad A]$

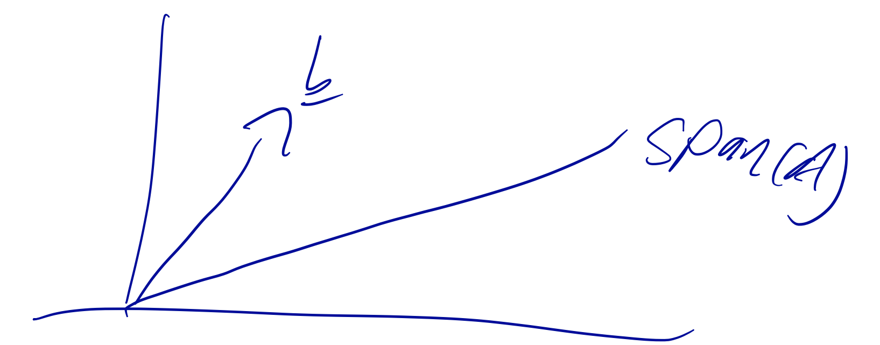
$\Downarrow$

min $\left\| \left(I - \frac{aa^T}{a^Ta}\right)\underline{b} - \left(I - \frac{aa^T}{a^Ta}\right) C y \right\|_2$ m×(n−1) dim prob.

```
function least_squares_eliminate(A,b)
  a = A[:,1]
  na = norm(a)
  a = a/na
  if size(A,2) == 1
    return [a'*b]/na
  end
  y = least_squares_eliminate(A[:,2:end]-a*a'*A[:,2:end], b - a*a'*b)
  γ = a'*(b - A[:,2:end]*y)/na
  return pushfirst!(y,γ)
end
```

A more general geometric view on least sq.





What is the closest point in span(A) to b?

a soln is

$A^T(b - A\underline{x}) = 0$

$A^T\underline{b} = A^TA\underline{x}$