Nomor 1

```
1;
function x = backwardSub(U, b)
  % This solves Ux = b, U is unit upper triangular matrix
  [n n] = size(U); x = zeros(n, 1); x(n) = b(n);
  for i = (n - 1):-1:1
    x(i) = b(i) - U(i, i+1:n) * x(i+1:n);
  endfor
endfunction

function x = forwardSub(L, b)
  % This solves Lx = b, L is unit lower triangular matrix
  [n n] = size(L);  x = zeros(n, 1);  x(1) = b(1);
  for i = 2:n
    x(i) = b(i) - L(i, 1:i-1) * x(1:i-1);
  endfor
endfunction

function x = diagSub(D, b)
  % This solves Dx = b, D is a diagonal matrix
  [n n] = size(D);  x = zeros(n, 1);
  for i = 1:n
    x(i) = b(i)/D(i, i);
  endfor
endfunction

function [A] = getSymmetric(n)
  % Generate a symmetric matrix with uniform distribution
  A = zeros(n, n);
  for i=1:n
    for j=1:n
      A(i, j) = rand();
      if(i > j)
        A(i, j) = A(j, i);
      endif
    endfor
  endfor
endfunction

function [L D] = LDLT(A)
   % Input: A, symmetric matrices
   % Output L and D such that A = L D L'
   [n n] = size(A);
   L = zeros(n, n);
   D = zeros(n);
   for k=1:n
     D(k) = A(k, k) - sum(L(k,1:k-1).^2.*D(1:k-1));
     for i=1:n
       % Cari pembaginya
       L(i, k) = (A(i, k) - sum(L(i, 1:k-1).*L(k, 1:k-1).*D(1:k-1)))/D(k);
     endfor
   endfor
   D = D(1:n,1);
```

```
endfunction

function [L D P] = LDLTwP(A)
   % Input: A, symmetric matrices
   % Output L and D such that A = L D L'
   [n n] = size(A);
   p = 1:n;
   L = zeros(n, n);
   D = zeros(n);
   for k=1:n
     % Extract maximum diagonal of submatrix k .. n : k .. n
     [M pos] = max(diag(A(k:n,k:n)))
     % p is the relative position
     pos += k - 1;

     % Swap pos and k
     tmp = p(pos);
     p(pos) = p(k);
     p(k) = tmp;

     % Swap Diagonal p and k
     tmp = A(pos,1:n)
     A(pos, 1:n) = A(k, 1:n);
     A(k, 1:n) = tmp;

     tmp = A(1:n, pos);
     A(1:n, pos) = A(1:n, k);
     A(1:n, k) = tmp;

     % Swap Diagonal p and k
     tmp = L(pos,1:n)
     L(pos, 1:n) = L(k, 1:n);
     L(k, 1:n) = tmp;

     tmp = L(1:n, pos);
     L(1:n, pos) = L(1:n, k);
     L(1:n, k) = tmp;

     D(k) = A(k, k) - sum(L(k,1:k-1).^2.*D(1:k-1));
     for i=1:n
       % Cari pembaginya
       L(i, k) = (A(i, k) - sum(L(i, 1:k-1).*L(k, 1:k-1).*D(1:k-1)))/D(k);
     endfor
   endfor
   D = D(1:n,1);


   P = zeros(n, n);
   for i=1:n
     P(i, p(i)) = 1
   endfor

endfunction

function [x] = solve(A, b)
  [L D P] = LDLTwP(A);
  disp(L);
  disp(D);
```

```
  disp(P);
  disp(L * diag(D) * L');
  disp(P * A * P');
  b = P * b;
  z = forwardSub(L, b);
  y = diagSub(diag(D), z);
  w = backwardSub(L', y);
  x = P' * w;
endfunction

A = [1, 2, 3, 4; 2, 5, 6, 7; 3, 6, 8, 9; 4, 7 , 9, 0];
b = [4; 1; 5; 8];

A = getSymmetric(3);
b = rand(3, 1);
disp(solve(A, b));
disp(A\b);
```

Jika kita diberikan persamaan $Ax = b$, dan diketahui dekomposisi $A = LDL^\intercal$, maka bisa kita tuliskan $LDL^\intercal x = b$. Pertama, kita akan menyelesaikan dari bagian terluarnya.

- Carilah z sehingga $Lz = b$. Perhatikan bahwa disini $DL^\intercal x = z$.
- Carilah y sehingga $Dy = z$. Perhatikan bahwa disini $L^\intercal x = y$.
- Carilah x sehingga $L^\intercal x = y$. Perhatikan bahwa x adalah solusi yang kita inginkan.

Perhatikan bahwa L ialah matriks **unit segitiga bawah**, artinya entri pada diagonal utamanya berisi satu, kita bisa buat fungsi backward dan forward substitution untuk L dan $L^\intercal$.

```
function x = backwardSub(U, b)
  % This solves Ux = b, U is unit upper triangular matrix
  [n n] = size(U); x = zeros(n, 1); x(n) = b(n);
  for i = (n - 1):-1:1
    x(i) = b(i) - U(i, i+1:n) * x(i+1:n);
  endfor
endfunction

function x = forwardSub(L, b)
  % This solves Lx = b, L is unit lower triangular matrix
  [n n] = size(L);  x = zeros(n, 1);  x(1) = b(1);
  for i = 2:n
    x(i) = b(i) - L(i, 1:i) * x(1:i);
  endfor
endfunction

function x = diagSub(D, b)
  % This solves Dx = b, D is a diagonal matrix
  [n n] = size(D);  x = zeros(n, 1);
  for i = 1:n
    x(i) = b(i)/D(i, i);
  endfor
endfunction
```

Bagaimana kasusnya untuk $PAP^\intercal = LDL^\intercal$? Pertama secara intuitif bisa kita tuliskan $A = P^{-1}LDL^\intercal(P^\intercal)^{-1}$. Perhatikan bahwa $PP^\intercal = I \iff P^\intercal = P^{-1}$.

Proof

$$(PP^\intercal)_{ij} = \begin{cases} 1 & P_{ik} \wedge P^\intercal_{kj} \iff i = j \\ 0 & \text{otherwise} \end{cases}$$

Akan kita selesaikan persamaan sebagai berikut.

$$\begin{aligned} P^{-1}LDL^\intercal(P^\intercal)^{-1}x &= b \\ LDL^\intercal(P^\intercal)^{-1}x &= Pb \\ LDL^\intercal(P^\intercal)^\intercal x &= Pb \\ LDL^\intercal Px &= Pb \end{aligned}$$

Untuk kemudahan komputasi, lakukan $b := Pb$

- Carilah z sehingga $Lz = b$. Perhatikan bahwa disini $DL^\intercal Px = z$.
- Carilah y sehingga $Dy = z$. Perhatikan bahwa disini $L^\intercal Px = y$.
- Carilah w sehingga $L^\intercal w = y$. Perhatikan bahwa disini $Px = w$.
- Carilah x sehingga $x = P^\intercal w$.

Total Flops:

- $LDL' = \frac{n^3}{3}$ (Setengah dari LU)
- Backward substitution = Forward Substitution = $\frac{n(n-1)}{2}$.
- Diagonal = n.
- Perkalian matriks permutasi (Kasus diagonal pivoting) n^3 (?)

Nomor 2

```
2;
function [A] = getBanded(n, p, q)
  A = zeros(n, n);
  for j=1:n
    for i=max(j-p, 1):min(j+q, n)
      A(i,j) = rand();
    endfor
  endfor
endfunction

function [A] = getBanded2(n, p, q)
  A = rand(n, n);
  for i=1:n
    for j=1:n
      if(j > i + p)
        A(i,j) = 0;
      endif
      if(i > j + q)
        A(i,j) = 0;
      endif
    endfor
  endfor
endfunction

function [L U] = LU(A, p, q)
  [n n] = size(A);
  L = eye(n);
  % Looping setiap kolom
  for k=1:n-1
    % Untuk baris ke bawahnya
```

```
      for i=k+1:min(k+q, n)
        % Hitung L nya
        L(i, k) = A(i, k)/A(k,k);
        A(i, k) = 0;
        % Lakukan pengurangan untuk baris ke-i
        for j=k+1:min(k+p, n)
          % Sampai k + p saja, karena kanan sisanya sudah 0
          A(i, j) -= L(i, k) * A(k, j);
        endfor
      endfor
    endfor
    U = A;
endfunction

n = 6; p = 2; q = 3;
A = getBanded2(n, p, q);
disp(A);
[L U] = LU(A, p, q);
disp(L); disp(U);
disp(L * U - A);
```

image-20210316072656245

Operasi perhitungan L pembagian di awal akan dilakukan sebanyak sekitar $(n-q-1)q + \sum_{i=1}^{q} i = \frac{2q(n-q-1)+q(q+1)}{2} = \frac{q(2n-q-1)}{2}$.

Operasi perkalian dan pengurangannya di bawah itu masing-masing ada sekitar $(n-q-1)\min(p,q)^2 \sum_{i=1}^{\min(p,q)} i^2$. Mungkin salah hehe ga tau deh susah ngitungnya. Tapi sekitar segitu lah.

Intinya kompleksitasnya sekitar $n(p+q)^2$

Nomor 3

Teorema Konsistensi

Ada SPL biasa (kumpulan persamaan-persamaan linear) , Ax = b (Persamaan Matriks), dan dalam bentuk matriks augmented (Eliminasi gauss jordan [A | b]. Alasan menggunakan matriks:

1. Kita dapat melupakan nama unknown (Namanya tidak berpengaruh)
2. Kita dapat menggunakan operasi-operasi pada matriks yang berlaku dalam SPL, dan menghasilkan matriks lain yang ekuivalen dengan SPL di awal (Solusinya sama)

Ada bentuk keempat penyajian spl, yaitu kombinasi linear kolom kolom.

$$x_1\overrightarrow{c_1} + x_2\overrightarrow{c_1} + \cdots + x_n\overrightarrow{c_1} = \vec{b}$$

Mencari solusi sama dengan mencari nilai $x_1, x_2, \ldots, x_n$ sehingga memenuhi persamaan linear tersebut

Ruang kolom ialah himpunan semua kombinasi linear kolom kolom, himpunan semua kombinasi linear didapat dari merentangkan suatu himpunan. Secara optimal dapat merentangkan basis. Artinya semua vektor pada himpunan yang baru dibentuk dapat dinyatakan dalam kombinasi linear kolom-kolom atau Coll dari suatu matriks. Sehingga $x_1\overrightarrow{c_1} + x_2\overrightarrow{c_1} + \cdots + x_n\overrightarrow{c_1} = \vec{b}$ akan konsisten, ada nilai vektor **x** nya sehingga memenuhi persamaan tersebut. Artinya apabila vektor **b** berada dalam Coll(A) atau berada dalam ruang kolom A, maka SPL A**x** = **b** konsisten.

Apabila konsisten maka ada solusi.

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \dots + a_{2n}x_n = b_2$$
$$\vdots$$
$$a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \dots + a_{mn}x_n = b_m$$

2) $A\bar{x} = \bar{b}$ konsisten
ada sol
$(\beta_1, \beta_2, \dots \beta_n)$
$\beta_1\bar{c}_1 + \beta_2\bar{c}_2 + \dots \beta_n\bar{c}_n = \bar{b}$
$\bar{b}$ k l $\bar{c}_1, \bar{c}_2 \dots \bar{c}_n$
$\bar{b} \in \text{Coll}(A)$

$$x_1 \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{m1} \end{bmatrix} + x_2 \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{m2} \end{bmatrix} + \dots + x_n \begin{bmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{mn} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

1) $\bar{b} \in \text{Coll}(A)$
ada $\alpha_1, \alpha_2 \dots \alpha_n$
$\alpha_1\bar{c}_1 + \alpha_2\bar{c}_2 + \dots \alpha_n\bar{c}_n = \bar{b}$
$(\alpha_1, \alpha_2, \dots \alpha_n)$ solusi
$A\bar{x} = \bar{b}$ konsisten

$$x_1\vec{c}_1 + x_2\vec{c}_2 + \dots + x_n\vec{c}_n = \vec{b}$$

Sehingga A**x** = **b** konsisten (***jika dan hanya jika***) $\iff$ **b** adalah kombinasi linear kolom-kolom A, atau $b \in \text{Coll}(A)$.

Ortogonal Komplemen

W adalah subruang dari RHKD V.

Vektor **v** dikatakan ortogonal dengan W jika **v** ortogonal dengan setiap vektor di W.

Himpunan semua vektor yagn ortogonal dengan W adalah subruang ortogonal komplemen dari W, ditulis $W^\perp$ (Dibaca **W perp**). Sifat-sifat ortogonal komplemen.

- $W^\perp$ subruang, dari V
  - Benar, bahwa vektor nol anggota $W^\perp$, kemudian perhatikan bahwa tertutup terhadap penjumlahan dan perkalian dengan skalar.
  - $\bar{b}, \bar{a} \perp W$
    $\bar{a} + \bar{b} \perp W$
    $\bar{a} \perp W$ $\forall \bar{v} \in W$ $\langle \bar{a}, \bar{v} \rangle = 0$
    $\bar{b}$ $\langle \bar{b}, \bar{v} \rangle = 0$
    $\langle \bar{a} + \bar{b}, \bar{v} \rangle = \langle \bar{a}, \bar{v} \rangle + \langle \bar{b}, \vec{v} \rangle = 0$
    $(\bar{a} + \bar{b}) \perp W$
- W dan $W^\perp$ tidak saling asing, ada vektor nol yang merupakan anggota dari kedua subruang
- $(W^\perp)^\perp = W$
- $V^\perp$ ialah {**0**}.
- Jika $W = W^\perp$ maka V = {**0**}.
- $\text{Null}(A^T) \perp \text{Coll}(A)$ dan $\text{Row}(A) \perp \text{Null}(A)$

Perhatikan bahwa ada beberapa cara penyajian SPL:

1. Himpunan persamaan-persamaan Linear
2. Persamaan Matriks
3. Matriks Augmented
4. Kombinasi Linear kolom-kolom A
5. Hasil kali titik baris-baris

---

1. Himpunan persamaan-persamaan linear $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n$
2. Persamaan matriks. $A\bar{x} = \bar{b}$
3. Matriks augmented $[A : \bar{b}]$
4. Kombinasi lenar kolom-kolom *A*. $x_1\bar{c}_1 + x_2\bar{c}_2 + \dots + x_n\bar{c}_n = \bar{b}$
5. Hasil kali titik baris-baris

$v = \quad \bar{x} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x \end{bmatrix}$

$$\begin{matrix} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \dots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \dots + a_{mn}x_n = b_m \end{matrix} \Rightarrow \begin{matrix} \mathbf{r}_1 \bullet \cancel{\mathbf{x}}\mathbf{v} = b_1 \\ \mathbf{r}_2 \bullet \cancel{\mathbf{x}}\mathbf{v} = b_2 \\ \vdots \\ \mathbf{r}_m \bullet \cancel{\mathbf{x}}\mathbf{v} = b_n \end{matrix} \Rightarrow \underbrace{\begin{bmatrix} \mathbf{r}_1\cancel{\mathbf{x}}\mathbf{v} \\ \mathbf{r}_2\cancel{\mathbf{x}}\mathbf{v} \\ \vdots \\ \mathbf{r}_m\cancel{\mathbf{x}}\mathbf{v} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}$$

Buktikan $\mathrm{Row}(A) \perp \mathrm{Null}(A)$

Langkah pembuktian:

1. Ambil sembarang vektor **v** yang ortogonal dengan setiap vektor di ruang baris *A*, buktikan bahwa *A***v** = **0**, **v** elemen Null(*A*)
2. Jika *A***v** = **0**, buktikan **v** ortogonal terhadap setiap vektor di ruang baris *A*.

$A\bar{v} = \bar{0}$

v ortogonal dengan setiap vektor di ruang baris *A*, maka **v** ortogonal dengan vektor ruang baris $r_1, r_2, \dots, r_m$

$$\mathbf{r}_1.\mathbf{v} = \mathbf{r}_2.\mathbf{v} = \dots = \mathbf{r}_m\mathbf{v} = 0$$

$$\begin{bmatrix} \mathbf{r}_1.\mathbf{v} \\ \mathbf{r}_2.\mathbf{v} \\ \dots \\ \mathbf{r}_m.\mathbf{v} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow$$

spl *A***x** = **0**

v elemen Null(*A*)

Langkah pembuktian:

1. Ambil sembarang vektor **v** yang ortogonal dengan setiap vektor di ruang baris *A*, buktikan bahwa *A***v** = 0, **v** elemen Null(*A*)
2. Jika *A***v** = 0, buktikan **v** ortogonal terhadap setiap vektor di ruang baris *A*.

$A\vec{v} = 0$, maka $\vec{v}$ $0 = r_1 . v = a_{11} v_1 + a_{12} v_2 + \dots$ $\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \end{bmatrix}$ $\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ r_1

$A\vec{v} = 0$, maka $\mathbf{r}_1 . \mathbf{v} = \mathbf{r}_2 . \mathbf{v} = \dots = \mathbf{r}_m \mathbf{v} = 0$ $\left(\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}\right) \left(\begin{bmatrix} v_1 \\ v_2 \\ v_m \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

Jika **k** sembarang vektor di Row(*A*), maka **k** dapat dinyatakan sebagai kombinasi linier dari vektor-vektor di Row(*A*): $\mathbf{k} = c_1 \mathbf{r}_1 + c_2 \mathbf{r}_2 + \cdots + c_m \mathbf{r}_m$

Maka $\mathbf{k}.\mathbf{v} = (c_1 \mathbf{r}_1 + c_2 \mathbf{r}_2 + \cdots + c_m \mathbf{r}_m).\mathbf{v}$ $\quad R . \vec{v}$

$$= c_1(\mathbf{r}_1 . \mathbf{v}) + c_2(\mathbf{r}_2 . \mathbf{v}) + \dots + c_m(\mathbf{r}_m . \mathbf{v})$$

$$= c_1 . 0 + c_2 . 0 + \dots + c_m . 0 = 0$$

v ortogonal terhadap setiap vektor di ruang baris *A*

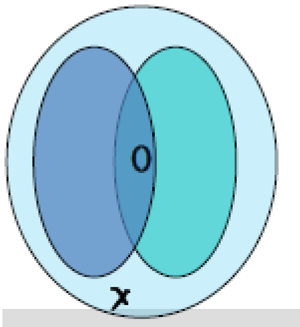
Diketahui bahwa

1. Coll(A^T) = Row(*A*)
2. $(\text{Row}(A))^{\perp} = (\text{Null}(A))$ [Row(A) dan Null(A) saling orthogonal komplemen]

Berdasarkan dua fakta di atas, maka dengan mengganti *A* dengan A^T, disimpulkan:

Coll(*A*) dan (Null(A^T)) saling komplemen orthogonal, ditulis $(\text{Coll}(A))^{\perp} = (\text{Null}(A^T))$

Buktikan bahwa subset satu sama lain.



Perhatikan bahwa Ortogonal komplemen belum tentu komplemen dengan irisan sama dengan 0, mungkin saja ada sebuah vektor pada suatu ruang vektor yang tidak ortogonal dengan keduanya. Misalnya, W = Span({(0, 1)}) dan $W^{\perp}$ = Span({(1, 0)}), perhatikan bahwa ada vektor **v** = (1, 1) yang bukan anggota keduanya.

Melalui Dua Titik

Tentukan gradiennya terlebih dahulu, kemudian cari persamaan y = mx + c yang memenuhi keduanya.

Melalui Tiga Titik

Akan didapatkan tiga titik,

- Melalui $\boxed{2 \text{ titik}}$

$$y = ax + b$$

$$\frac{y - y_1}{x - x_1} =$$

- Melalui 3 titik

Persm umum y : $y = ax + b$

(x_1, y_1), (x_2, y_2), (x_3, y_3)

$$\left.\begin{aligned} y_1 &= ax_1 + b \\ y_2 &= ax_2 + b \\ y_3 &= ax_3 + b \end{aligned}\right\} \text{SPL}$$

unknown a, b

$$\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Akan diselesaikan SPL:

$$\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

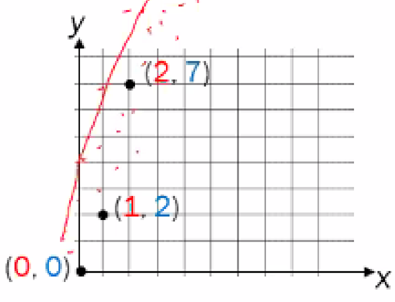
Bila ada solusi konsisten, maka, ada solusi yang melewati tiga titik.

Pada LSS kita ingin mencari persamaan garis yang cocok, meskipun tidak konsisten.

Mencocokkan garis lurus pada data

Diberikan titik-titik pada bidang (0, 0), (1, 2), (2, 7).

Ditentukan persm garis lurus yang : $y = ax + b$ **yang melaui** titik-titik tsb.



Diperoleh spl sebagai berikut

$$0 = a.0 + b$$
$$2 = a.\ 1 + b$$
$$7 = a.\ 2 + b$$

dalam bentuk matriks ditulis:

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 7 \end{bmatrix}$$

Spl tidak konsisten karena titik-titik tidak berada dalam satu garis.
Pertanyaan: **bagaimana menemukan persamaan garis yang paling cocok menyajikan data tersebut?**

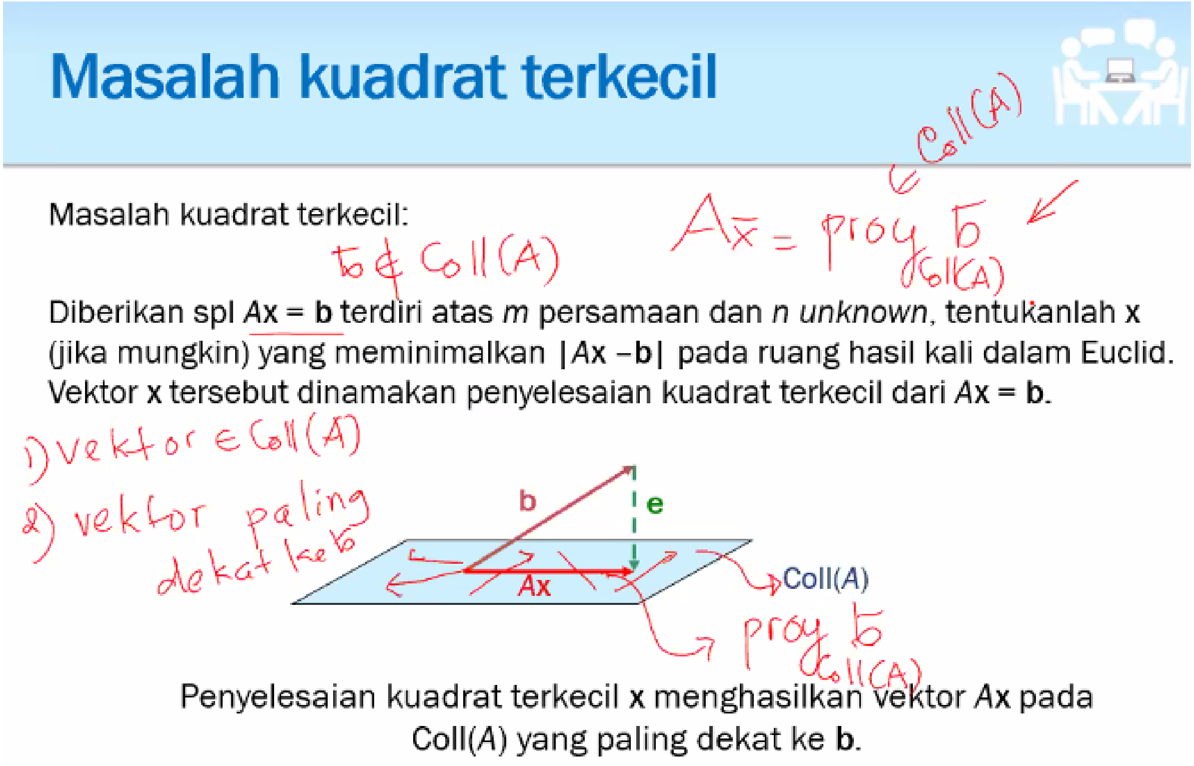
Kita ingin mencocokkan kurva pada data. Tidak apa apa tidak kolinear, tapi kita masih ingin mencari garis terbaik yang errornya minimal.

Kita harus mengerti teorema konsistensi dan perp.

Masalah kuadrat terkecil ialah kita diberikan spl A**x** = **b**, kita ingin meminimalkan |A**x**-**b**|, atau jarak euclideannya, makanya jarak kuadrat.

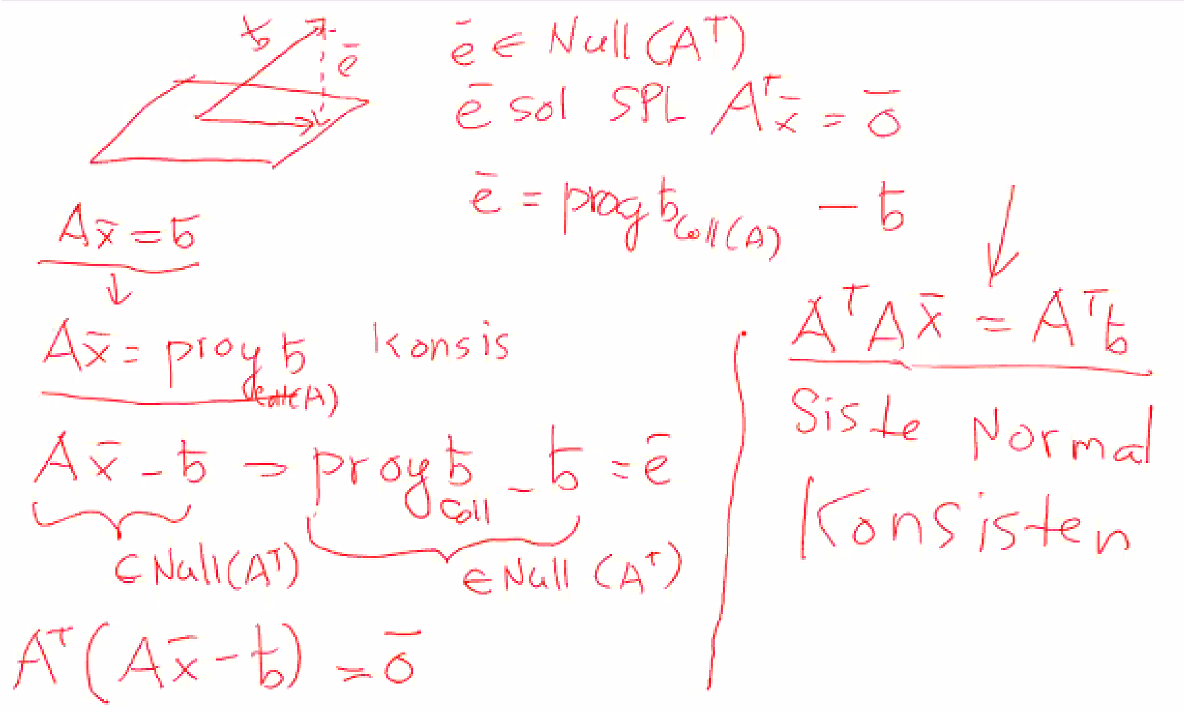
Kita ingin mencari pengganti b yang paling dekat, dan b itu harus berada dalam $\text{Coll}(A)$ dan vektornya harus paling dekat ke **b**, agar errornya minimal. Yang paling dekat ialah **proyeksi ortogonal dari b** di ruang kolom A.

Ingin dicari $A\vec{x} = \text{proj}_{\text{Coll(A)}}(\vec{b})$



vektor **e** adalah $\text{proj}_{\text{Coll(A)}}(\vec{b}) - \vec{b}$, perhatikan bahwa **e** ortogonal dengan **A**, maka $\vec{e} \in \text{Null}(A^T)$

maka $\vec{e}$ solusi SPL $A^T\vec{x} = \vec{0}$.



Nomor 4

Observasi penting dari dot product dua vektor $\vec{u}$ dan $\vec{v}$ ialah $u \cdot v = u^\intercal v$. Perhatikan sifat penting sebagai berikut.

$$Qu \cdot Qv = u \cdot v = (Qu)^\intercal (Qv) = u^\intercal Q^\intercal Qv = u^\intercal v = u \cdot v$$

Dari sini, bisa dibawa kepada observasi selanjutnya, bahwa:

$$||Qx||_2 = \sqrt{Qx \cdot Qx} = \sqrt{x \cdot x} = ||x||_2$$

Referensi:

- https://sites.math.rutgers.edu/~cherlin/Courses/250/Lectures/250L26.html

Jika Q_1 dan Q_2 merupakan matriks ortogonal, (pada matriks ortogonal Q, berlaku $Q^\intercal = Q^{-1}$ maka $Q_1 Q_2 Q_2^\intercal Q_1^\intercal = I$, sehingga $Q_1 Q_2 (Q_1 Q_2)^\intercal = I$, sehingga $Q_1 Q_2$ ortogonal juga.

Hint : I is symmetric and uu^T is symmetric.

Thus it follows,

$I^T = I$ and $(uu^T)^T = uu^T$ (Recall that $(AB)^T = B^T A^T$)

Hence $H^T = (I - uu^T/\beta)^T = I^T - (uu^T/\beta)^T = I - uu^T/\beta = H$

Verily , H is symmetric.

$H = I - \frac{2}{u^T u} uu^T = I + \alpha\, uu^T$

$H^T = (I + \alpha\, uu^T)^T = I^T + \alpha\, (uu^T)^T = I + \alpha (u^T)^T u^T = I + \alpha\, uu^T = H$

Where I used $I^T = I$ and the basic properties of the transposed matrix, namely:

- For scalars λ we have $(\lambda A)^T = \lambda A^T$

- $(A+B)^T = A^T + B^T$
- $(AB)^T = B^T A^T$
- $(A^T)^T = A$

D:

http://fourier.eng.hmc.edu/e176/lectures/NM/node10.html