1. Summary

We have just enjoyed a interesting class last week which divided us into 4 groups and compete with each other. I leared a lot on my topic, however, I failed to understand other groups' topic at once.

Our topic is Recursive Least Square method which is pretty intuitive. The origin least square problem need a full matrix A with all of the observations in it which may take a lot of time to get the ovservations. So, this motivated recursive method to be carried out. The crucial idea of recursive method is that the observation and the calculation can be done parallelly. Once some new obsevation is carried out, the matrix A can be updated and the $\hat{x}$ can be calculated.

Two groups discussed the Steepest Descent method and the Conjugate Gradient method. Those two methods basically are recursively approaching the least square problem.

The last group, who is also the best presenter, carried out the kalman filter. Kalman filter is kind of like a better version of Recursive Least Square method. It requires less complexity.

2.

6.2. (QR and Least Square, 2 pts) Let $A = QR$ is a $m \times n$ $(m > n)$ matrix of rank n and corresponding QR decomposition. Q is an $m \times m$ orthogonal matrix and R is an $m \times n$ upper-triangular matrix. Then when it comes to the problem $y = Ax$, we can derive a solution $\hat{x}$ by simple elimination procedure of equation $Rx = Q^T y$. Show that $\hat{x}$ is exactly the least square solution.

The least square problem is solving Ax = b and find the $\hat{x}$ to minimize the Ax.

$$\begin{aligned} Ax &= b \\ QRx &= b \\ Q^T QRx &= Q^T b \end{aligned}$$

since Q is orthogonal:

$$\begin{aligned} Q^T QRx &= Q^T b \\ Rx &= Q^T b \end{aligned}$$

Since A is a $m \times n$ matrix of rank n:

$$\begin{aligned} Rx &= Q^T b \\ \begin{bmatrix} U \\ R' \end{bmatrix} x &= \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}^T b \end{aligned}$$

U is a $n \times n$ matrix of rank n, R' is a $(m-n) \times n$ matrix. Q_1 is a $n \times n$ matrix, Q_2 is a $n \times (m-n)$

$$Ux = Q_1^T b$$
$$x = U^{-1} Q_1^T b$$

$$R'x = Q_2^T b$$

So, we have:

$$||Ax - b||^2 = ||x - U^{-1} Q_1^T b||^2 + ||Q_2^T b - R'x||$$

when $x = \hat{x}$, $||x - U^{-1} Q_1^T b||^2 = 0$, $||Ax - b||^2 = ||Q_2^T b - R'x||$, which is the minimal value. So, $\hat{x}$ is the least square answer.

3.

6.3. (SD and CG, 3 pts) Consider the optimal condition $Hx = b$ corresponding to the problem

$$\min_x \frac{1}{2} x^T H x - b^T x$$

where

$$H = \begin{bmatrix} 3 & -1 \\ -1 & 2 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Calculate the first two steps for Steepest Descent (SD) and Conjugate Gradients (CG) and compare their result, both starting form $x_0 = [1, 1]^T$.

SD

step 1:

$$
\begin{aligned}
r_0 &= b - Hx_0 \\
&= \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 3 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} -1 \\ -1 \end{bmatrix} \\
\alpha_0 &= \frac{r_0^T r_0}{r_0^T H r_0} \\
&= \frac{\begin{bmatrix} -1 \\ -1 \end{bmatrix}^T \begin{bmatrix} -1 \\ -1 \end{bmatrix}}{\begin{bmatrix} -1 \\ -1 \end{bmatrix}^T \begin{bmatrix} 3 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \end{bmatrix}} \\
&= \frac{2}{3} \\
x_1 &= x_0 + \alpha_0 x_0 \\
&= \begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \end{bmatrix}
\end{aligned}
$$

step 2:

$$
\begin{aligned}
r_1 &= b - Hx_1 \\
&= \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 3 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \end{bmatrix} \\
\alpha_1 &= \frac{r_1 0^T r_1}{r_1^T H r_1} \\
&= \frac{\begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \end{bmatrix}^T \begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \end{bmatrix}}{\begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \end{bmatrix}^T \begin{bmatrix} 3 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \end{bmatrix}} \\
&= 0 \\
x_2 &= x_1 + \alpha_1 x_1 \\
&= \begin{bmatrix} \frac{1}{3} \\ -\frac{1}{3} \end{bmatrix}
\end{aligned}
$$

CG

init:

$$
\begin{aligned}
r_0 &= b - Hx_0 \\
&= \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 3 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} -1 \\ -1 \end{bmatrix} \\
d_0 = r_0 &= \begin{bmatrix} -1 \\ -1 \end{bmatrix}
\end{aligned}
$$

step0:

$$
\begin{aligned}
\alpha_0 &= \frac{r_0^T r_0}{d_0^T H d_0} \\
&= \frac{2}{3}
\end{aligned}
$$

$$
\begin{aligned}
x_1 &= x_0 + \alpha_0 d_0 \\
&= \begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
r_1 &= r_0 - \alpha_0 H d_0 \\
&= \begin{bmatrix} \frac{4}{3} \\ \frac{2}{3} \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\beta_1 &= \frac{r_1^T r_1}{r_0^T r_0} \\
&= \frac{10}{9} \\
d_1 &= r_1 + \beta_1 d_0 \\
&= \begin{bmatrix} \frac{2}{9} \\ -\frac{4}{9} \end{bmatrix}
\end{aligned}
$$

step1:

$$
\begin{aligned}
\alpha_1 &= \frac{r_1^T r_1}{d_1^T H d_1} \\
&= 3
\end{aligned}
$$

$$
\begin{aligned}
x_2 &= x_1 + \alpha_1 d_1 \\
&= \begin{bmatrix} 1 \\ -1 \end{bmatrix}
\end{aligned}
$$

6.4 Kalman Filter

6.4. (Kalman Filter, 3 pts) Consider a dynamic system for 'pulse tracking problem' defined by

$$x_{k+1} = F_k x_k + \epsilon_k$$
$$y_{k+1} = A_{k+1} x_{k+1} + e_{k+1}$$

where we use current estimation as prediction ($F_k = 1$), and we assume our observation is unbiased measurement ($A_k = 1$). All noise have unit vairance.

(a) Please re-state the Kalman Filter Update Algorithm under this specific setting.

(b) Show that when k is large enough, the state estimation folmula will turn out to be a weighted sum of current observation and previous estimation

$$x_{k+1|k+1} = \alpha y_{k+1} + (1 - \alpha) x_{k|k}$$

where $\alpha = \frac{\sqrt{5}-1}{2}$.

a.

We have the following at first:

$$y_0 = A_0 x_0 + e_0$$

Then we have:

$$0 = F_0 x_0 - x_1 + \epsilon_0$$
$$y_1 = A_1 x_1 + e_1$$

In other words:

$$\begin{bmatrix} y_0 \\ 0 \\ y_1 \end{bmatrix} = \begin{bmatrix} A_0 & 0 \\ F_0 & -I \\ 0 & A_1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} + \begin{bmatrix} e_0 \\ \epsilon_0 \\ e_1 \end{bmatrix}$$

We call the matrix $\bar{A}_1$:

$$\begin{bmatrix} A_0 & 0 \\ F_0 & -I \\ 0 & A_1 \end{bmatrix}$$

We can calculate the least square problem by calculating:

$$\begin{bmatrix} x_{\hat{0}|1} \\ x_{\hat{1}|1} \end{bmatrix} = (\bar{A_1}^T \bar{A_1})^{-1} \bar{A_1}^T \bar{y_k}$$

b.

When k is very big, the following equations:

$$x_{k+1} = F_k x_k + \epsilon_k$$

$$y_{k+1} = A_{k+1}x_{k+1} + e_{k+1}$$

can be simplized into:

$$x_{k+1} = x_k + \epsilon_k$$

$$y_{k+1} = x_{k+1} + e_{k+1}$$

Now, $x_{k+1|k+1}$ is:

$$x_{k+1|k+1} = K_{k+1}(y_{k+1} - x_{k|k}) + x_{k|k}$$

where K_{k+1} is Kalman gain which can be approximated to $\frac{\sqrt{5}-1}{2}$.