



Hemuppgift 3 i 5B1760 Linjär och kvadratisk optimering 2003/2004.
Inlämnas senast torsdag 27 november 2003 kl. 12.00.

Examinator: Krister Svanberg, tel 790 71 37

Ange namn, personnummer och *e-postadress* på rapportens framsida.

I denna uppgift är samarbete tillåtet vid utformningen av Matlab-beräkningarna, men sedan ska varje student *själv*, med egna ord, skriva en kortfattad rapport i vilken redovisas hur uppgiften lösts och vad resultatet blev. Ange även vem eller vilka du samarbetat med. Kopiering av någon annans rapport är inte tillåtet!

Matlab-kod och utskrifter ska bifogas. Vissa studenter kommer (delvis slumpmässigt) att väljas ut för enskild muntlig redovisning av uppgifterna. Kallelse sker via e-post, så kontrollera regelbundet denna.

3.1. Givet nio punkter $(t_1, s_1), \dots, (t_9, s_9)$ till vilka man vill anpassa en linje $s = kt + \ell$. Antag först att man vill välja konstanterna k och ℓ i enlighet med övre halvan av avsnitt 4.2 i Gröna häftet, dvs så att den *största* vertikala avvikelserna mellan linjen och de givna punkterna blir så liten som möjligt.

Skriv en m-fil i Matlab som för givna vektorer $\mathbf{t} = (t_1, \dots, t_9)^T$ och $\mathbf{s} = (s_1, \dots, s_9)^T$ löser LP-problemet med hjälp av Matlab-funktionen **linprog**.

Som resultat skall erhållas dels optimala värden på k och ℓ , dels en vektor $\mathbf{g} = (g_1, \dots, g_9)^T$ med $g_i = kt_i + \ell$, dvs $\mathbf{g} = k \cdot \mathbf{t} + \ell \cdot \mathbf{e}$ där $\mathbf{e} = (1, \dots, 1)^T$.

3.2. Antag nu i stället att man vill välja konstanterna k och ℓ i enlighet med nedre halvan av avsnitt 4.2 i Gröna häftet, dvs så att *summan* av de nio vertikala avvikelserna mellan linjen och de givna punkterna blir så liten som möjligt.

Skriv en m-fil i Matlab som för givna vektorer $\mathbf{t} = (t_1, \dots, t_9)^T$ och $\mathbf{s} = (s_1, \dots, s_9)^T$ löser LP-problemet med hjälp av Matlab-funktionen **linprog**.

Som resultat skall erhållas dels optimala värden på k och ℓ , dels en vektor $\mathbf{h} = (h_1, \dots, h_9)^T$ med $h_i = kt_i + \ell$, dvs $\mathbf{h} = k \cdot \mathbf{t} + \ell \cdot \mathbf{e}$ där $\mathbf{e} = (1, \dots, 1)^T$.

3.3. Antag nu i stället att man vill välja konstanterna k och ℓ i enlighet med avsnitt 13.1 i Gröna häftet, dvs så att *kvadratsumman* av de nio vertikala avvikelserna mellan linjen och de givna punkterna blir så liten som möjligt.

Skriv en m-fil i Matlab som för givna vektorer $\mathbf{t} = (t_1, \dots, t_9)^T$ och $\mathbf{s} = (s_1, \dots, s_9)^T$ löser detta MK-problem genom att ställa upp och lösa normalekvationerna.

Som resultat skall erhållas dels optimala värden på k och ℓ , dels en vektor $\mathbf{p} = (p_1, \dots, p_9)^T$ med $p_i = kt_i + \ell$, dvs $\mathbf{p} = k \cdot \mathbf{t} + \ell \cdot \mathbf{e}$ där $\mathbf{e} = (1, \dots, 1)^T$.

- 3.4. Generera \mathbf{t} och \mathbf{s} , dvs de nio punkterna $(t_1, s_1), \dots, (t_9, s_9)$, på följande sätt, där du byter ut 222222 mot de sex första siffrorna i ditt födelsenummer.

```
>> rand('state',222222)
>> t = [1:9]';
>> e = ones(9,1);
>> s = 2*e + 0.5*t + rand(9,1);
```

Bestäm optimala värden på k och ℓ för de tre fallen ovan, samt skriv ut dessa.

Beräkna även följande resultatmatris:

$$\mathbf{U}(1,1) = \max_i |g_i - s_i|, \quad \mathbf{U}(1,2) = \max_i |h_i - s_i|, \quad \mathbf{U}(1,3) = \max_i |p_i - s_i|,$$

$$\mathbf{U}(2,1) = \sum_i |g_i - s_i|, \quad \mathbf{U}(2,2) = \sum_i |h_i - s_i|, \quad \mathbf{U}(2,3) = \sum_i |p_i - s_i|,$$

$$\mathbf{U}(3,1) = \sqrt{\sum_i (g_i - s_i)^2}, \quad \mathbf{U}(3,2) = \sqrt{\sum_i (h_i - s_i)^2}, \quad \mathbf{U}(3,3) = \sqrt{\sum_i (p_i - s_i)^2},$$

samt skriv ut denna 3×3 -matris.

Följande olikheter ska förstas gälla om du räknat rätt:

$$\mathbf{U}(1,1) < \mathbf{U}(1,2), \quad \mathbf{U}(1,1) < \mathbf{U}(1,3), \quad \mathbf{U}(2,2) < \mathbf{U}(2,1),$$

$$\mathbf{U}(2,2) < \mathbf{U}(2,3), \quad \mathbf{U}(3,3) < \mathbf{U}(3,1), \quad \mathbf{U}(3,3) < \mathbf{U}(3,2).$$

Dessutom ska för varje $j = 1, 2, 3$ gälla att $\mathbf{U}(1, j) < \mathbf{U}(3, j) < \mathbf{U}(2, j)$.

Använd slutligen Matlab-funktionen **plot** för att i en och samma figur plotta dels de nio genererade punkterna, dels de tre framräknade linjerna.

Exempelvis kan du pröva att skriva **plot(t,s,'o',t,g,'-',t,h,'-.',t,p,'--')**.

Lycka till!