**程式設計與統計分析專題期末考**

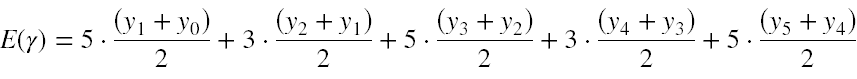
**R05454008 周哲宇**

**1 Hanging Chain Problem**

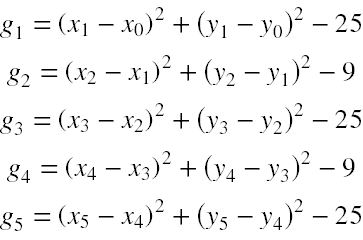
**以五條長度分別為[5*,*3*,*5*,*3*,*5] 公尺的鐵條，起始點為(0*,* 0)，終點為(11*,-*3)，設立一地板為 *y* = *-*0*.*5*x -* 4，請問中間的四個接點座標。**

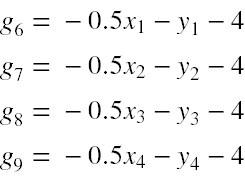
1. 首先，請寫出：

(a) 最小化目標式(*E*)



(b) 等式與不等式限制式(*CI*)



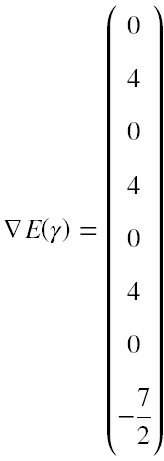


(c)

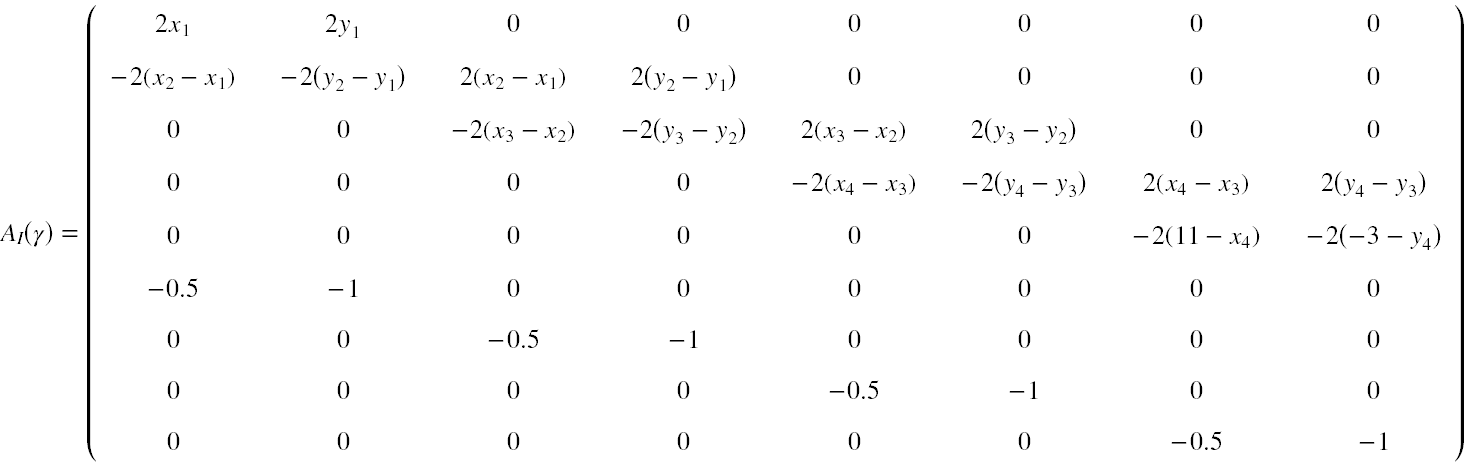


2. 其次，為了要進行sequential quadratic programming，我們需要先求出（列出結果）：

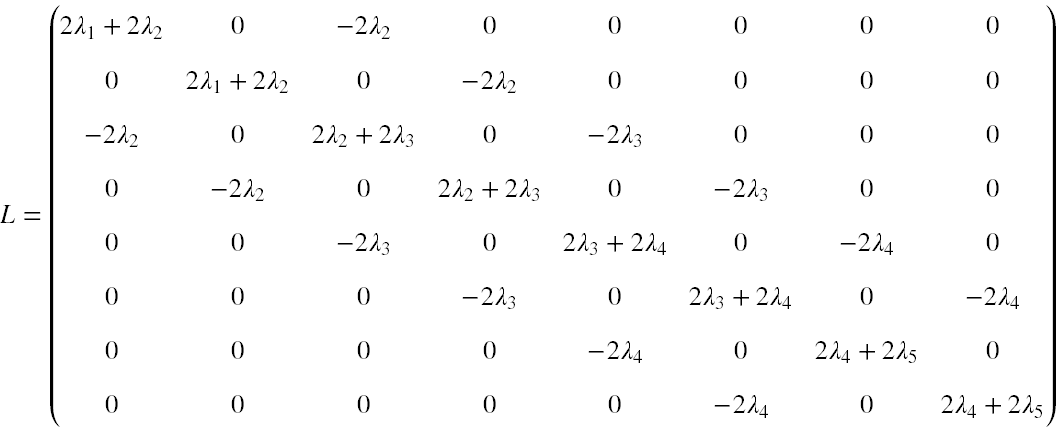
(a) *∇E* =



(b) *AI* =



(c) *L* =



3. 其次，我們需要將上面的資訊寫進程式中，請利用我們星期二寄發給大家的範例為基礎，求得參數解。

1. 請附上使用的程式碼，規則如之前的習題

record=ones(1,32)\*-999;

LT=[5 3 5 3 5]';

cont\_a=-0.5; cont\_b=-1; cont\_v=-4;

a0=0;

b0=0;

a=11;

b=-3;

r0=[1 0 1 0 1 -1 1 -2]'; % [x1 y1 x2 y2 x3 y3 x4 y4]

lamda=ones(9,1);

kk=1; run=1; stop=1;

runmax=1000; tolv=10^-4;

record(1,1)=0;

record(1,2:9)=r0';

record(1,10:17)=zeros(1,8);

record(1,18:26)=lamda';

record(1,27)=0.5\*(LT(1)\*(b0+r0(2))+LT(2)\*(r0(2)+r0(4))+LT(3)\*(r0(4)+r0(6))+LT(4)\*(r0(6)+r0(8))+LT(5)\*(r0(8)+b));

record(1,28:32)=ones(1,5)\*-999;

**while** stop>0 && run<runmax

g1=(a0-r0(1))^2+(b0-r0(2))^2-LT(1)^2;

g2=(r0(3)-r0(1))^2+(r0(4)-r0(2))^2-LT(2)^2;

g3=(r0(5)-r0(3))^2+(r0(6)-r0(4))^2-LT(3)^2;

g4=(r0(7)-r0(5))^2+(r0(8)-r0(6))^2-LT(4)^2;

g5=(a-r0(7))^2+(b-r0(8))^2-LT(5)^2;

g6=cont\_a\*r0(1)+cont\_b\*r0(2)+cont\_v;

g7=cont\_a\*r0(3)+cont\_b\*r0(4)+cont\_v;

g8=cont\_a\*r0(5)+cont\_b\*r0(6)+cont\_v;

g9=cont\_a\*r0(7)+cont\_b\*r0(8)+cont\_v;

ci=[g1 g2 g3 g4 g5 g6 g7 g8 g9]';

df=[0 (LT(1)+LT(2))/2 0 (LT(2)+LT(3))/2 0 (LT(3)+LT(4))/2 0 (LT(4)+LT(5))/2]';

ai1=[2\*(r0(1)-a0) 2\*(r0(2)-b0) 0 0 0 0 0 0];

ai2=[-2\*(r0(3)-r0(1)) -2\*(r0(4)-r0(2)) 2\*(r0(3)-r0(1)) 2\*(r0(4)-r0(2)) 0 0 0 0];

ai3=[0 0 -2\*(r0(5)-r0(3)) -2\*(r0(6)-r0(4)) 2\*(r0(5)-r0(3)) 2\*(r0(6)-r0(4)) 0 0];

ai4=[0 0 0 0 -2\*(r0(7)-r0(5)) -2\*(r0(8)-r0(6)) 2\*(r0(7)-r0(5)) 2\*(r0(8)-r0(6))];

ai5=[0 0 0 0 0 0 -2\*(a-r0(7)) 2\*(b+r0(8))];

ai6=[cont\_a cont\_b 0 0 0 0 0 0];

ai7=[0 0 cont\_a cont\_b 0 0 0 0];

ai8=[0 0 0 0 cont\_a cont\_b 0 0];

ai9=[0 0 0 0 0 0 cont\_a cont\_b];

ai=[ai1;ai2;ai3;ai4;ai5;ai6;ai7;ai8;ai9];

L1=[2\*lamda(1)+2\*lamda(2) 0 -2\*lamda(2) 0 0 0 0 0];

L2=[0 2\*lamda(1)+2\*lamda(2) 0 -2\*lamda(2) 0 0 0 0];

L3=[-2\*lamda(2) 0 2\*lamda(2)+2\*lamda(3) 0 -2\*lamda(3) 0 0 0];

L4=[0 -2\*lamda(2) 0 2\*lamda(2)+2\*lamda(3) 0 -2\*lamda(3) 0 0];

L5=[0 0 -2\*lamda(3) 0 2\*lamda(3)+2\*lamda(4) 0 -2\*lamda(4) 0];

L6=[0 0 0 -2\*lamda(3) 0 2\*lamda(3)+2\*lamda(4) 0 -2\*lamda(4)];

L7=[0 0 0 0 -2\*lamda(4) 0 2\*lamda(4)+2\*lamda(5) 0];

L8=[0 0 0 0 0 -2\*lamda(4) 0 2\*lamda(4)+2\*lamda(5)];

L=[L1;L2;L3;L4;L5;L6;L7;L8];

% KKT

KKT1=df+ai'\*lamda; check1=sum(abs(KKT1)>tolv);

KKT2=ci; check2=sum(KKT2>tolv);

KKT3=lamda; check3=sum(KKT3<-tolv);

KKT4=lamda'\*ci; check4=sum(abs(KKT4)>tolv);

check=check1+check2+check3+check4;

**if** check==0

stop=0;

**end**

d=quadprog(L,df,ai,-ci);

lamdaQP=(ai')\(-df-L\*d);

r1=r0+d;

lamda=lamdaQP;

record(kk+1,1)=kk;

record(kk+1,2:9)=r1';

record(kk+1,10:17)=d';

record(kk+1,18:26)=lamdaQP';

record(kk+1,27)=0.5\*(LT(1)\*(b0+r1(2))+LT(2)\*(r1(2)+r1(4))+LT(3)\*(r1(4)+r1(6))+LT(4)\*(r1(6)+r1(8))+LT(5)\*(r1(8)+b));

record(kk+1,28)=record(kk+1,21)-record(kk,21);

record(kk+1,29:32)=[check1 check2 check3 check4];

%\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Step 4 and 5: return to Step 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

r0=r1;

kk=kk+1;

run=run+1;

**end**

1. 列出四個接點座標

**1. (2.15113482651459, -4.51360376618890)**

**2. (4.57170113824618, -6.28585056911987)**

**3. (9.04383709324075, -8.52191854661689)**

**4. (11.0000000003707, -7.99997303457419)**

(c) 既然有地板條件，請列出落在地板上的接點是哪幾個？有沒有哪一條鐵條「躺」在地板上？(要想像這個情況，我們可以從地板的函數圖形開始想像...)

**(4.57170113824618, -6.28585056911987)**

**(9.04383709324075, -8.52191854661689)**

**地2,3,4個鐵條都有碰到地板，3躺在地板上。**

**2 線性迴歸與TRMCO Model**

1. 線性迴歸模型：

(a) 線性迴歸模型的數學設定(請將變數*x\_*1 先取對數(*log*) 再進行中心化, *x*2 以中心化操作，即*x\**1 = *log*(*x*1) *– mean(log*(*x*1))

y=final(:,1); # 設定依變項

x0 =final(:,2); # 1向量

mxs1=log(final(:,3))-mean(log(final(:,3))); # 中心化x1

mxs2=final(:,4)-mean(final(:,4)); # 中心化x2

x = [x0 mxs1 mxs2]; # 設定矩陣

1. 各迴歸係數的意義

𝛽0為截距

𝛽n為斜率，表自變數每增加一單位時，因變數Y的改變量

𝛽1 為

𝛽2

(c) 採用最小平方法(最大概似法亦同) 的迴歸係數，請求出其解，並且解釋其結果。

beta = (x'\*x)\x'\*y; # 計算迴歸係數

# 結果如下

beta0: 3.241, beta1:-0.837, beta2: 0.008

1. 我們知道可由模型推出，請算出各國的民主之家分數的估計值，並檢驗是否有不合理的情況(最小值是1，最大值為7)。

sum(finalre < 1 | finalre > 7) = 2 #有兩個不合理的值

|  |
| --- |
| 3.37502142467094 |
| 3.26938203040935 |
| 2.59808822559939 |
| 3.45554932841456 |
| 1.20937105210943 |
| 1.37276008371116 |
| 4.47416905100859 |
| 2.98025804835291 |
| 1.41242431568480 |
| 4.86654285139477 |
| 3.96367789652238 |
| 2.84482009416038 |
| 2.97928903843522 |
| 5.04648373616150 |
| 5.74425858684680 |
| 4.41668337135142 |
| 1.40813501221893 |
| 4.81966291909484 |
| 2.51989184283455 |
| 2.96807727774776 |
| 3.13300143516498 |
| 4.92407293606985 |
| 2.77938132021639 |
| 2.48581685315690 |
| 1.92758923439222 |
| 2.14281829473699 |
| 1.15837845101733 |
| 3.19320411679547 |
| 3.23930264791097 |
| 3.61533103650841 |
| 3.56562714277249 |
| 2.20714396688180 |
| 5.05651944459223 |
| 3.34584480493897 |
| 1.36128829613367 |
| 1.51833829559905 |
| 3.48764021699200 |
| 1.41285045516719 |
| 4.38102877663717 |
| 2.13150808919489 |
| 3.68283535091705 |
| 4.94367040500361 |
| 5.17417697100833 |
| 4.86914807516060 |
| 4.15195633111631 |
| 2.46529236649603 |
| 1.28678739513064 |
| 4.25204939839680 |
| 3.64692975904176 |
| 3.10394195900575 |
| 1.32204385993963 |
| 1.69527204937926 |
| 1.71517405765044 |
| 3.49592619844112 |
| 2.53543650337020 |
| 4.45552877300076 |
| 2.37942407053947 |
| 2.87137897702283 |
| 4.57062777291879 |
| 5.43887256173407 |
| 2.39263663986992 |
| 1.01471763084428 |
| 5.43386434724903 |
| 5.63894348527532 |
| 4.44898131122935 |
| 2.74158969276521 |
| 2.84013886753777 |
| 3.49048247539649 |
| 2.95637136485233 |
| 4.52391965707742 |
| 4.87764536612234 |
| 1.33240370828933 |
| 4.15560666428673 |
| 5.42376185246326 |
| 0.773888317670884 |
| 4.30876736650825 |
| 2.71543100517543 |
| 3.54832267944942 |
| 3.20039053112376 |
| 3.63266024811718 |
| 2.45105079099666 |
| 2.12285842137755 |
| 2.69444457360267 |
| 2.61016095675304 |
| 5.05822412035203 |
| 4.70468983191647 |
| 3.16179726354792 |
| 2.49281328385476 |
| 5.13599020153705 |
| 2.17121738846615 |
| 1.96974335359726 |
| 4.11872423274279 |
| 3.36559683186774 |
| 1.84897468042954 |
| 1.85026704399719 |
| 3.57221346570945 |
| 1.18362432979116 |
| 0.908398060219447 |
| 4.45546994047019 |
| 3.21157184225363 |
| 5.22597620106954 |
| 3.50477512566214 |
| 2.61512178150710 |
| 3.23271326455529 |
| 5.02882270845828 |
| 3.74925468170383 |
| 1.47396258408526 |
| 1.32022591425834 |
| 2.38739037845488 |
| 3.69711243814994 |
| 4.06613489624375 |
| 4.54824381812483 |
| 4.45483266511891 |
| 4.80937488093925 |

1. 倘若我們知道經驗上的*x*1*, x*2 各有其最大最小值，請利用114 個國家在兩自變量所出現的數值中，所能組合出的一組最小值與一組最大值，來評估民主之家分數估計值的最大最小值，並討論所解出的最小平方解是否滿足依變項的值域限制。

maxv = [1 min(mxs1) max(mxs2)]; # 最小值組合

minv = [1 max(mxs1) min(mxs2)]; # 最大值組合

minv\*beta = 0.7631 # 小於1不合理

maxv\*beta = 5.9449 # 域值之內(1-7), 故合理

2. 請運用習題七附件中所準備的Matlab 制限最佳化程式

以下是我做的參數設定**[[1]](#footnote-1)**:

y=final(:,1);

xs1 = final(:,3);

mxs1 = log(final(:,3))-mean(log(final(:,3)));

mxs2 = final(:,4)-mean(final(:,4));

xs=[mxs1 mxs2];

r0=[1 1 1 1]';

a=1; b=7; runmax=1001; tau=10; tolv=10^-6;

xmin=[1 min(mxs1) min(mxs2)]'; xmax=[1 max(mxs1) max(mxs2)]';

(a) 評估滿足依變項界域情況下，算出的迴歸係數解為何？

beta0: 3.153, beta1:-0.757, beta2: 0.065

(b) 算出各國的民主之家分數估計值，檢驗是否有不合理的情況

sum(result < 1 | result > 7) = 9 # 有九個不合理

(c) 考量自變量在經驗上所能組合出的最小值與最大值，評估民主之家分數估計值的最大最小值

最小值不合理: 0.197

最大值不合理: 7.13

3. 假設有一國之GNI per capita 從990 成長到15; 500，Gini coefficient 從60 變為35，請根據第六題的迴歸係數結果，計算該國的自由之家指數會從多少改變至多少？若在分類上，我們可藉由該指數將國家區分為「自由」(1.0-2.5), 「半自由」(3.0-5.0), 「不自由」(5.5-7.0)，則該國在分類上又會有什麼改變？具體的意義又為何？

4. 最後，請各位根據自己的學科，思考一個可以使用迴歸分析的具體研究問題，列出依變項與可能的自變項，並簡單評估可能會遇到的依變項值域限制問題。

事後情緒狀態評估分數 = 情緒情境 + 初始情緒

限制: 情緒狀態評估分數(-5到+5) 、情緒情境(1-8)、初始情緒狀態(-5到+5)

1. **可能是我的參數設定錯誤才會出現這種結果? 但我找不出bug在哪?**  [↑](#footnote-ref-1)