UNIVERSITAT DE BARCELONA

Departament d'Estadística
Av. Diagonal, 643
08028 Barcelona

MÈTODES NO PARAMÈTRICS I DE REMOSTREIG. Grau en Estadística. 2014-15

Prova de síntesi. 18 de juny de 2015.

Respon als mateixos fulls de l'examen. Si no tens prou espai: fes servir la darrera plana, en blanc, o fulls "UB" addicionals.

lloc	agost	novembre
1	8.1	11.2
2	10.0	16.3
3	16.5	15.3
4	13.6	15.6
5	9.5	10.5
6	8.3	15.5
7	18.3	12.7
8	13.3	11.1
9	7.9	19.9
10	8.1	20.4
11	8.9	14.2
12	12.6	12.7
13	13.4	36.8

Problema 1. En un estudi sobre els efectes de la contaminació en els boscos, es van escollir 13 llocs a l'atzar d'una zona molt contaminada, i per cada lloc es va mesurar el nivell d'alumini (en micrograms per gram de fusta) d'un pollancre. Per cada lloc la mesura es va fer el mes d'agost i el mes de novembre.

Respon les següents qüestions, utilitzant els llistats del final de l'enunciat de l'examen quan ho creguis convenient. En tot moment considerarem un nivell de significació de 0.05 o un nivell de confiança de 0.95. Quan realitzis una prova d'hipòtesis has d'expressar clarament les hipòtesis nul·la i alternativa.

1) Realitza una prova d'hipòtesis basada en rangs que sigui adequada per a intentar demostrar que la mediana del nivell d'alumini ha crescut d'agost a novembre. En concret, respon les següents qüestions:
    a. Nom de la prova triada, raons per triar-la i condicions de validesa (0.5 punts):

    b. Hipòtesis nul·la i alternativa, valor de l'estadística de test (justificant els passos per obtenir-lo) i conclusió final (0.75 punts):

(*Hi ha més espai a la pàgina següent*)

COGNOMS, NOM:	**FIRMA:**

2) Que l'increment de contaminació sigui significatiu (o no) i que sigui important no són la mateixa cosa. Per valorar aquest segon aspecte pot ser útil disposar d'una estimació de la mediana de les diferències.
    a. Indica el valor de l'estimació puntual que correspondria a la prova d'hipòtesis realitzada a l'apartat anterior (0.5 punts):

    b. Calcula l'interval de confiança **bilateral** que correspondria a la prova d'hipòtesis realitzada a l'apartat anterior (estadísticament, seria un interval unilateral, però aquí es demana el bilateral, que també pot ser interessant quant a la interpretació) (0.75 punts):

3) Realitza un prova de permutacions per intentar demostrar que el nivell d'alumini **mitjà** ***ha variat*** d'agost a novembre. Indica clarament, explicant els passos o càlculs realitzats quan calgui:
    a. Les hipòtesis contrastades i el valor de l'estadístic de test (0.5 punts):

b. El p-valor i la conclusió final (0.5 punts):

4) Suposa que, en les mateixes condicions d'abans (els mateixos llocs del bosc en mesos diferents, etc.), el nivell d'alumini s'hagués mesurat en més de 2 mesos (per exemple: agost, octubre i desembre). Indica el nom d'una prova basada en rangs adequada per a intentar demostrar que la mediana del nivell d'alumini ha variat segons els mesos (0.5 punts).

5) Continuant amb la suposició de la pregunta anterior (més de 2 mesos), explica com realitzaries una prova de permutacions per demostrar diferències en les mitjanes dels mesos (no l'has de fer, només explicar com ho faries). En concret, respon les següents qüestions:
   a. Què caldria permutar? (0.5 punts)

   b. Justifica **si seria possible** (o no) fer una prova de permutacions **exacta** per aquestes dades (0.5 punts):

Problema 2. Per les mateixes dades del problema anterior, atès que cada parella de valors de contaminació per alumini es refereix al mateix lloc del bosc, seria d'esperar que hi hagués un cert grau de dependència entre les variables *X* = 'agost' i *Y* = 'novembre'.

Respon les següents qüestions, utilitzant els llistats del final de l'enunciat quan ho creguis convenient. En tot moment considerarem un nivell de significació de 0.05 o un nivell de confiança de 0.95.

1) Pel coeficient de correlació de Kendall, ignorant el empats:
   a. Obtén la seva estimació puntual (0.5 punts):

b. Determina si és significativament diferent de zero, indicant les hipòtesis, l'estadístic de test, el valor crític de les taules i la conclusió final (0.5 punts):

c. El resultat del test anterior, demostra que els nivells d'alumini d'agost i novembre són estocàsticament independents? (0.5 punts)

2) Indica justificadament el valor del coeficient de correlació de Spearman (0.5 punts):

3) Realitza una prova de permutacions per a determinar si el coeficient de correlació lineal de Pearson entre *X* i *Y* és negatiu. En concret:
   a. Indica les hipòtesis nul·la i alternativa i el valor de l'estadístic de test (0.5 punts):

   b. Calcula el p-valor i indica la conclusió final (0.5 punts):

4) Calcula els següents intervals de confiança bootstrap (no paramètric) **bilaterals** pel coeficient de correlació de Pearson ρ:
   a. Percentil bootstrap (0.5 punts):

b. Bootstrap-t (0.5 punts):

c. Bootstrap-t simetritzat (0.5 punts):

d. A la darrera pàgina de llistats, a partir del comentari: # 10000 rèpliques bootstrap no paramètric d'r i del seu error estàndard: hi ha les instruccions R per realitzar la simulació bootstrap i obtenir els intervals de confiança anteriors. Indica **què caldria canviar** per obtenir els intervals de confiança **bootstrap paramètric** suposant que (*X*,*Y*) segueix una distribució normal bivariant:

LLISTATS R

```
> # ********************************************************************
> #      PROBLEMA 1
> # ********************************************************************
> pollancres = read.table("pollancres.txt", header = TRUE)
>
> agost = pollancres$agost
> novembre = pollancres$novembre
>
> # Diferència dins cada lloc (13 valors possibles):
> d = agost - novembre
> d
 [1]  -3.1  -6.3   1.2  -2.0  -1.0  -7.2   5.6   2.2 -12.0 -12.3  -5.3  -0.1 -23.4
> n = length(d)
> n
[1] 13
> # Valors absoluts de les diferències:
> abs.d = abs(d)
> abs.d
 [1]  3.1  6.3  1.2  2.0  1.0  7.2  5.6  2.2 12.0 12.3  5.3  0.1 23.4
> #
```

```
> # Rangs dels valors absoluts de les diferències:
> rabs.d = rank(abs.d)
> rabs.d
 [1]  6  9  3  4  2 10  8  5 11 12  7  1 13
> #
> # Suma de rangs de diferencies positives:
> r.plus = sum(rabs.d[d > 0])
> r.plus
[1] 16
> # Suma de rangs de diferencies negatives:
> r.minus = sum(rabs.d[d < 0])
> r.minus
[1] 75
> #
> # Mediana de totes les diferències:
> median(d)
[1] -3.1
> # Mediana de totes les semisumes entre diferències:
> sSums = outer(d, d, "+") / 2
> median(sSums[lower.tri(sSums, diag = TRUE)])
[1] -4.1
> #
> #
> #
> # Totes les dades en un únic vector:
> alumini = c(agost, novembre)
>
> N = length(alumini)
> N
[1] 26
> n1 = n
> n2 = n
>
> # rang de cada observació dins el total de N = 26 valors:
> rangs <- rank(alumini)
> rangs
 [1]  2.5  7.0 22.0 16.0  6.0  4.0 23.0 14.0  1.0  2.5  5.0 11.0 15.0 10.0 21.0
[16] 18.0 20.0  8.0 19.0 12.5  9.0 24.0 25.0 17.0 12.5 26.0
> # rangs de les observacions "agost"
> rangs[1:n1]
 [1]  2.5  7.0 22.0 16.0  6.0  4.0 23.0 14.0  1.0  2.5  5.0 11.0 15.0
> # rangs de les observacions "novembre"
> rangs[(n1+1):N]
 [1] 10.0 21.0 18.0 20.0  8.0 19.0 12.5  9.0 24.0 25.0 17.0 12.5 26.0
>
> # Sumes de rangs dins cada grup:
> sum(rangs[1:n1])
[1] 129
> sum(rangs[(n1+1):N])
[1] 222
>
> # Totes les diferències possibles (13 * 13 = 169 valors)
> # entre "agost" i "novembre":
> dd = outer(agost, novembre, "-")
> dd
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]  [,9] [,10] [,11] [,12] [,13]
 [1,] -3.1 -8.2 -7.2 -7.5 -2.4 -7.4 -4.6 -3.0 -11.8 -12.3  -6.1  -4.6 -28.7
 [2,] -1.2 -6.3 -5.3 -5.6 -0.5 -5.5 -2.7 -1.1  -9.9 -10.4  -4.2  -2.7 -26.8
 [3,]  5.3  0.2  1.2  0.9  6.0  1.0  3.8  5.4  -3.4  -3.9   2.3   3.8 -20.3
 [4,]  2.4 -2.7 -1.7 -2.0  3.1 -1.9  0.9  2.5  -6.3  -6.8  -0.6   0.9 -23.2
 [5,] -1.7 -6.8 -5.8 -6.1 -1.0 -6.0 -3.2 -1.6 -10.4 -10.9  -4.7  -3.2 -27.3
 [6,] -2.9 -8.0 -7.0 -7.3 -2.2 -7.2 -4.4 -2.8 -11.6 -12.1  -5.9  -4.4 -28.5
 [7,]  7.1  2.0  3.0  2.7  7.8  2.8  5.6  7.2  -1.6  -2.1   4.1   5.6 -18.5
 [8,]  2.1 -3.0 -2.0 -2.3  2.8 -2.2  0.6  2.2  -6.6  -7.1  -0.9   0.6 -23.5
 [9,] -3.3 -8.4 -7.4 -7.7 -2.6 -7.6 -4.8 -3.2 -12.0 -12.5  -6.3  -4.8 -28.9
[10,] -3.1 -8.2 -7.2 -7.5 -2.4 -7.4 -4.6 -3.0 -11.8 -12.3  -6.1  -4.6 -28.7
[11,] -2.3 -7.4 -6.4 -6.7 -1.6 -6.6 -3.8 -2.2 -11.0 -11.5  -5.3  -3.8 -27.9
[12,]  1.4 -3.7 -2.7 -3.0  2.1 -2.9 -0.1  1.5  -7.3  -7.8  -1.6  -0.1 -24.2
[13,]  2.2 -2.9 -1.9 -2.2  2.9 -2.1  0.7  2.3  -6.5  -7.0  -0.8   0.7 -23.4
>
> # mediana de les 169 diferències:
> median(dd)
[1] -3.2
>
> # Les 169 diferències ordenades (continua a la página següent):
> sort(dd)
  [1] -28.9 -28.7 -28.7 -28.5 -27.9 -27.3 -26.8 -24.2 -23.5 -23.4 -23.2 -20.3 -18.5
 [14] -12.5 -12.3 -12.3 -12.1 -12.0 -11.8 -11.8 -11.6 -11.5 -11.0 -10.9 -10.4 -10.4
```

```
 [27]  -9.9  -8.4  -8.2  -8.2  -8.0  -7.8  -7.7  -7.6  -7.5  -7.5  -7.4  -7.4  -7.4
 [40]  -7.4  -7.3  -7.3  -7.2  -7.2  -7.2  -7.1  -7.0  -7.0  -6.8  -6.8  -6.7  -6.6
 [53]  -6.6  -6.5  -6.4  -6.3  -6.3  -6.3  -6.1  -6.1  -6.1  -6.0  -5.9  -5.8  -5.6
 [66]  -5.5  -5.3  -5.3  -4.8  -4.8  -4.7  -4.6  -4.6  -4.6  -4.6  -4.4  -4.4  -4.2
 [79]  -3.9  -3.8  -3.8  -3.7  -3.4  -3.3  -3.2  -3.2  -3.2  -3.1  -3.1  -3.0  -3.0
 [92]  -3.0  -3.0  -2.9  -2.9  -2.9  -2.8  -2.7  -2.7  -2.7  -2.7  -2.6  -2.4  -2.4
[105]  -2.3  -2.3  -2.2  -2.2  -2.2  -2.2  -2.1  -2.1  -2.0  -2.0  -1.9  -1.9  -1.7
[118]  -1.7  -1.6  -1.6  -1.6  -1.6  -1.2  -1.1  -1.0  -0.9  -0.8  -0.6  -0.5  -0.1
[131]  -0.1   0.2   0.6   0.6   0.7   0.7   0.9   0.9   0.9   1.0   1.2   1.4   1.5
[144]   2.0   2.1   2.1   2.2   2.2   2.3   2.3   2.4   2.5   2.7   2.8   2.8   2.9
[157]   3.0   3.1   3.8   3.8   4.1   5.3   5.4   5.6   5.6   6.0   7.1   7.2   7.8
> #
> #
> # Permutacions sobre el vector de 13 diferències
> # ==============================================
> # Enumeració de TOTES les permutacions possibles, maneres segons les quals
> # podem permutar DINS cada parella de valors (agost, novembre).
> # En altres paraules, maneres possibles segons les quals podem donar
> # un signe - o + a les diferències:
> sgn = c(-1, +1)
> signsTab = expand.grid(as.data.frame(matrix(rep(sgn, n), ncol = n)))
> signsTab = apply(signsTab, 1, "*", abs.d)
> # Cada columna de 'signsTab' conté les diferències sobre una permutació
> # possible. Per exemple les 10 primeres:
> signsTab[,1:10]
     [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
V1   -3.1   3.1  -3.1   3.1  -3.1   3.1  -3.1   3.1  -3.1   3.1
V2   -6.3  -6.3   6.3   6.3  -6.3  -6.3   6.3   6.3  -6.3  -6.3
V3   -1.2  -1.2  -1.2  -1.2   1.2   1.2   1.2   1.2  -1.2  -1.2
V4   -2.0  -2.0  -2.0  -2.0  -2.0  -2.0  -2.0  -2.0   2.0   2.0
V5   -1.0  -1.0  -1.0  -1.0  -1.0  -1.0  -1.0  -1.0  -1.0  -1.0
V6   -7.2  -7.2  -7.2  -7.2  -7.2  -7.2  -7.2  -7.2  -7.2  -7.2
V7   -5.6  -5.6  -5.6  -5.6  -5.6  -5.6  -5.6  -5.6  -5.6  -5.6
V8   -2.2  -2.2  -2.2  -2.2  -2.2  -2.2  -2.2  -2.2  -2.2  -2.2
V9  -12.0 -12.0 -12.0 -12.0 -12.0 -12.0 -12.0 -12.0 -12.0 -12.0
V10 -12.3 -12.3 -12.3 -12.3 -12.3 -12.3 -12.3 -12.3 -12.3 -12.3
V11  -5.3  -5.3  -5.3  -5.3  -5.3  -5.3  -5.3  -5.3  -5.3  -5.3
V12  -0.1  -0.1  -0.1  -0.1  -0.1  -0.1  -0.1  -0.1  -0.1  -0.1
V13 -23.4 -23.4 -23.4 -23.4 -23.4 -23.4 -23.4 -23.4 -23.4 -23.4
> # Nombre de permutacions possibles:
> nperm = ncol(signsTab)
> nperm
[1] 8192
> #
> # Estimació de la mitjana de les diferències sobre cada possible permutació:
> m.perm = apply(signsTab, 2, mean)
> #
> # La mitjana de les diferències a la mostra original és:
> m.d = mean(d)
> m.d
[1] -4.9
> #
> sum(m.perm >= m.d)
[1] 8070
> sum(abs(m.perm) >= abs(m.d))
[1] 246
> sum(m.perm <= m.d)
[1] 123
>
>
>
>
>
> # ********************************************************************
> #      PROBLEMA 2
> # ********************************************************************
> x = pollancres$agost
> y = pollancres$novembre
>
> # Taula amb totes les possibles diferències entre x[i] i x[j]:
> difs.x = outer(x, x, "-")
> # Descartem les diferències de la diagonal (i == j) i de la meitat
> # triangular superior:
> difs.x = difs.x[ltri <- lower.tri(difs.x)]
> # Totes les possibles diferències entre y[i] i y[j]:
> difs.y = outer(y, y, "-")[ltri]
> #
```

```
> # Nombre de concordances:
> concor = sum(sign(difs.x)*sign(difs.y) > 0)
> concor
[1] 33
> # Nombre de discordances:
> discor = sum(sign(difs.x)*sign(difs.y) < 0)
> discor
[1] 43
> #
> #
> #
> cor.test(rank(x), rank(y))

        Pearson's product-moment correlation

data:  rank(x) and rank(y)
t = -0.4612, df = 11, p-value = 0.6536
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.6401437  0.4471824
sample estimates:
       cor
-0.137741

> #
> #
> # Coeficient de correlació lineal de Pearson sobre la mostra original:
> r = cor(x,y)
> r
[1] 0.01669772
> # En tot el problema, com a estimació de l'error estàndard del coeficient
> # de correlació de Pearson mostral farem servir l'estimació paramètrica
> # normal (1 - r^2) / sqrt(n - 3):
> se.r = (1 - r*r) / sqrt(n - 3)
> se.r
[1] 0.3161396
>
> # Una permutació aleatòria del vector 'y':
> y.perm = sample(y, replace = FALSE)
> # 9999 permutacions aleatòries i càlcul de la correlació:
> nperm = 9999
> set.seed(5719)
> r.perms = replicate(nperm, cor(x, sample(y, replace = FALSE)))
> #
> sum(r.perms >= r)
[1] 4291
> sum(abs(r.perms) >= abs(r))
[1] 9645
> sum(r.perms <= r)
[1] 5708
>
> # 1 remostra bootstrap:
> # Determino quins llocs formaran part de la mostra aleatòria i amb
> # reemplaçament:
> i.boot = sample(1:n, replace = TRUE)
> i.boot
 [1]  8 12  1  3  3  7 13  6  1  7  9  1  4
> # Remostra bootstrap:
> x[i.boot]
 [1] 13.3 12.6  8.1 16.5 16.5 18.3 13.4  8.3  8.1 18.3  7.9  8.1 13.6
> y[i.boot]
 [1] 11.1 12.7 11.2 15.3 15.3 12.7 36.8 15.5 11.2 12.7 19.9 11.2 15.6
> # Correlació sobre la remostra bootstrap:
> r.boot = cor(x[i.boot], y[i.boot])
> r.boot
[1] 0.05319807
> # (és el que normalment hauríem anomenat un valor r*)
> #
> #
> # Error estàndar de la correlació mostral per aquesta remostra
> # bootstrap concreta (se*, atenció, NO és l'error estàndard de la
> # correlació mostral per les dades originals):
> se.boot = (1 - r.boot*r.boot) / sqrt(n - 3)
> se.boot
[1] 0.3153328
> #
```

```
> #
> #
> # 10000 rèpliques bootstrap no paramètric d'r i del seu error estàndard:
> nboot = 10000
> set.seed(5719)
> r.boots = replicate(nboot,
+ {
+   i.boot = sample(1:n, replace = TRUE)
+   r.boot = cor(x[i.boot], y[i.boot])
+   se.boot = (1 - r.boot*r.boot)
+   c(r.boot, se.boot)
+ }
+ )
> rownames(r.boots) = c("r*", "se*")
> # Faltava dividir per sqrt(n - 3):
> r.boots[2,] = r.boots[2,] / sqrt(n - 3)
> #
> # r.boots és una matriu de 2 files i 10000 columnes,
> # la primera fila són les répliques bootstrap de la correlació,
> # la segona fila son els corresponents errors estàndard.
> # Les 10 primeres rèpliques bootstrap:
> r.boots[,1:10]
          [,1]       [,2]       [,3]       [,4]       [,5]       [,6]       [,7]
r*  -0.2623799 0.4303000 0.03881186 -0.0244792 0.3868921 0.09051152 -0.08380535
se*  0.2944576 0.2576756 0.31575141  0.3160383 0.2688931 0.31363712  0.31400679
           [,8]       [,9]       [,10]
r*  -0.01902641 -0.3155566 -0.05463458
se*  0.31611329  0.2847391  0.31528385
> # Valors estudentitzats:
> t.boots  = (r.boots["r*",] - r) / r.boots["se*",]
> # Els 10 primers:
> t.boots[1:10]
 [1] -0.94776831  1.60512768  0.07003654 -0.13029094  1.37673455  0.23534777
 [7] -0.32006657 -0.11301055 -1.16687290 -0.22624789
> #
> # Alguns quantils:
> quantile(r.boots["r*",], probs = c(0.025, 0.975))
      2.5%      97.5%
-0.6081624  0.4646415
> quantile(t.boots, probs = c(0.975, 0.025))
   97.5%     2.5%
 1.80654 -3.13579
> quantile(abs(r.boots["r*",]), probs = 0.95)
      95%
0.5635586
> quantile(abs(t.boots), probs = 0.95)
    95%
2.654493
```