Prova de síntesi. 13 de juny de 2014

MÈTODES NO PARAMÈTRICS I DE REMOSTREIG. Grau en Estadística. Curs 2013-14

estudiant	probabilitats	discretes
1	65	52
2	88	57
3	83	78
4	92	76
5	50	30
6	67	67
7	100	96
8	100	74
9	73	65
10	90	87
11	83	78
12	94	89

Problema 1. Les dades de l'esquerra corresponen a l'avaluació contínua de 12 estudiants d'un mateix curs d'introducció a l'estadística matemàtica. La columna "probabilitats" correspon a una avaluació de la part "càlcul de probabilitats", mentre que la columna "discretes" correspon a una avaluació similar de la part "variables aleatòries i distribucions discretes" feta posteriorment als mateixos estudiants. Són notes sobre 100.

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) Indica el nom d'una prova d'hipòtesis basada en rangs que sigui adequada per a intentar demostrar que la nota mediana ha variat de l'avaluació de probabilitats a la de discretes. Indica les condicions de validesa de la prova que has triat. Indica a quina prova paramètrica normal (en aquest cas, per les mitjanes) seria comparable.

2) Indica justificadament les hipòtesis, el resultat i la conclusió final d'aquesta prova. Què demostra aquest resultat?

NOM I COGNOMS: **FIRMA:**

3) Indica justificadament el valor de l'estimació puntual i l'interval de confiança per al canvi experimentat en la mediana de les diferències de notes.

4) Els resultats anteriors no són exactes (per exemple, el p-valor que s'obtindria no seria del tot correcte), per quina raó? Per a la mateixa situació, seria exacta una prova de permutacions? Per quina raó?

5) Realitza un prova de permutacions per intentar demostrar que la nota **mitjana** ha variat entre l'avaluació de probabilitats i la de discretes. Indica clarament i de forma justificada les hipòtesis d'aquest test, el p-valor obtingut i la conclusió final.

Problema 2. Per les mateixes dades del problema anterior, atès que cada parella de notes es refereix al mateix alumne, seria d'esperar que hi hagués un cert grau de dependència entre les variables *X* = 'probabilitats' i *Y* = 'discretes'.

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. Quan realitzis una prova d'hipòtesis has d'expressar clarament les hipòtesis nul·la i alternativa:

1) Ignorant el empats, calcula el coeficient de correlació de Kendall entre *X* i *Y* i determina si és significatiu. Explica el significat del valor obtingut (que no vol dir el mateix que "ser significatiu estadísticament") d'aquest coeficient.

2) Indica el valor del coeficient de correlació de Spearman.

3) Realitza una prova de permutacions per a determinar si el coeficient de correlació lineal de Pearson és significatiu.

4) Calcula l'interval de confiança bootstrap percentil pel coeficient de correlació de Pearson (bootstrap no paramètric).

5) Calcula l'interval de confiança bootstrap-t per al coeficient de correlació de Pearson (bootstrap no paramètric).

LLISTATS R

```
> # ************************************************************
> #      PROBLEMA 1
> # ************************************************************
> avaluacions = read.table("avaluacions.txt", header = TRUE)
>
> wilcox.test(avaluacions$probabilitats, avaluacions$discretes,
+   paired = FALSE, correct = FALSE, conf.int = TRUE)

        Wilcoxon rank sum test

data:  avaluacions$probabilitats and avaluacions$discretes
W = 99, p-value = 0.1186
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
 -3.000033 24.000009
sample estimates:
difference in location
              11.2736

Warning messages:
1: In wilcox.test.default(avaluacions$probabilitats, avaluacions$discretes,  :
  cannot compute exact p-value with ties
2: In wilcox.test.default(avaluacions$probabilitats, avaluacions$discretes,  :
  cannot compute exact confidence intervals with ties
> wilcox.test(avaluacions$probabilitats, avaluacions$discretes,
+   paired = FALSE, alternative = "greater", correct = FALSE, conf.int = TRUE)

        Wilcoxon rank sum test

data:  avaluacions$probabilitats and avaluacions$discretes
W = 99, p-value = 0.05932
alternative hypothesis: true location shift is greater than 0
95 percent confidence interval:
 -0.9999481        Inf
sample estimates:
difference in location
              11.2736

Warning messages:
1: In wilcox.test.default(avaluacions$probabilitats, avaluacions$discretes,  :
  cannot compute exact p-value with ties
2: In wilcox.test.default(avaluacions$probabilitats, avaluacions$discretes,  :
  cannot compute exact confidence intervals with ties
>
> # Diferències:
> d = avaluacions$probabilitats - avaluacions$discretes
> d
 [1] 13 31  5 16 20  0  4 26  8  3  5  5
> n = length(d)
> n
[1] 12
> # Valors absoluts de les diferències:
> abs.d = abs(d)
> abs.d
 [1] 13 31  5 16 20  0  4 26  8  3  5  5
> #
> # Rangs dels valors absoluts de les diferències:
> rabs.d = rank(abs.d)
> rabs.d
 [1]  8 12  5  9 10  1  3 11  7  2  5  5
> #
> # Suma de rangs de diferencies positives:
> r.plus = sum(rabs.d[d > 0])
> r.plus
[1] 77
> # Suma de rangs de diferencies negatives:
```

```
> r.minus = sum(rabs.d[d < 0])
> r.minus
[1] 0
> #
> #
> # Mediana de totes les diferències:
> median(d)
[1] 6.5
> # Mediana de totes les semisumes entre diferències:
> sSums = outer(d, d, "+") / 2
> median(sSums[lower.tri(sSums, diag = TRUE)])
[1] 10.5
>
> # Permutacions sobre el vector de 12 diferències
> # ==============================================
> # Enumeració de TOTES les permutacions possibles, maneres segons les quals
> # podem permutar DINS cada parella de valors (probabilitats, discretes).
> # 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   -13   13  -13   13  -13   13  -13   13  -13    13
V2   -31  -31   31   31  -31  -31   31   31  -31   -31
V3    -5   -5   -5   -5    5    5    5    5   -5    -5
V4   -16  -16  -16  -16  -16  -16  -16  -16   16    16
V5   -20  -20  -20  -20  -20  -20  -20  -20  -20   -20
V6     0    0    0    0    0    0    0    0    0     0
V7    -4   -4   -4   -4   -4   -4   -4   -4   -4    -4
V8   -26  -26  -26  -26  -26  -26  -26  -26  -26   -26
V9    -8   -8   -8   -8   -8   -8   -8   -8   -8    -8
V10   -3   -3   -3   -3   -3   -3   -3   -3   -3    -3
V11   -5   -5   -5   -5   -5   -5   -5   -5   -5    -5
V12   -5   -5   -5   -5   -5   -5   -5   -5   -5    -5
> # Nombre de permutacions possibles:
> nperm = ncol(signsTab)
> nperm
[1] 4096
> #
> # 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)
> #
> sum(m.perm >= m.d)
[1] 2
> sum(abs(m.perm) >= abs(m.d))
[1] 4
> sum(m.perm <= m.d)
[1] 4096

> # ****************************************************************
> #      PROBLEMA 2
> # ****************************************************************
> x = avaluacions$probabilitats
> y = avaluacions$discretes
>
> # 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 entrey 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] 51
> # Nombre de discordances:
> discor = sum(sign(difs.x)*sign(difs.y) < 0)
> discor
[1] 13
> #
> cor.test(rank(x), rank(y))

        Pearson's product-moment correlation

data:  rank(x) and rank(y)
t = 3.354, df = 10, p-value = 0.007316
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2638789 0.9181189
sample estimates:
      cor
0.7275934

> #
> # Coeficient de correlació lineal de Pearson sobre la mostra original:
> r = cor(x,y)
> r
[1] 0.8372688
> # En tot el problema, com a estimació de l'error estàndard del coeficient
> # de correlació de Pearson mostral farem servir l'estimació paramètrica normal:
> se.r = (1 - r*r) / sqrt(n - 3)
> se.r
[1] 0.09966033
>
> # 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] 12
> sum(abs(r.perms) >= abs(r))
[1] 12
> sum(r.perms <= r)
[1] 9987
>
> # 1 remostra bootstrap:
> # Determino quins estudiants formaran part de la mostra aleatòria i amb
> # reemplaçament:
> i.boot = sample(1:n, replace = TRUE)
> i.boot
 [1] 1 1 5 8 6 7 6 7 6 6 5 7
> # Remostra bootstrap:
> x[i.boot]
 [1]  65  65  50 100  67 100  67 100  67  67  50 100
> y[i.boot]
 [1] 52 52 30 74 67 96 67 96 67 67 30 96
> # Correlació sobre la remostra bootstrap:
> r.boot = cor(x[i.boot], y[i.boot])
> r.boot
[1] 0.914613
> # Error estàndar d'aquesta estimació de la correlació:
> se.boot = (1 - r.boot*r.boot) / sqrt(n - 3)
> se.boot
[1] 0.05449437
> # 10000 rèpliques bootstrap no paramètric de 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 5 primeres rèpliques bootstrap:
> r.boots[,1:5]
         [,1]      [,2]      [,3]       [,4]       [,5]
r*  0.5389235 0.7971575 0.7980396 0.98421245 0.85728882
se* 0.2365205 0.1215133 0.1210443 0.01044195 0.08835196
> # Valors "estudentitzats":
> t.boots  = (r.boots["r*",] - r) / r.boots["se*",]
> # Els 5 primers:
> t.boots[1:10]
 [1] -1.2613929 -0.3300982 -0.3240894 14.0724303  0.2265941
> #
> # Alguns quantils:
> quantile(r.boots["r*",], probs = c(0.025, 0.975))
     2.5%     97.5%
0.4268142 0.9788047
> quantile(t.boots, probs = c(0.975, 0.025))
    97.5%      2.5%
10.123830 -1.505648
> quantile(abs(r.boots["r*",]), probs = 0.95)
      95%
0.9655576
> quantile(abs(t.boots), probs = 0.95)
     95%
5.684999
```