Proiect Probabilități și Statistică

Observații și explicații

Grupa 231
Gălățan Alexandru-Cristian
Ungureanu Andrei-Liviu

Introducere

Setul de date *CO2* conține măsurători pentru absorbția de dioxid de carbon a unor plante din regiuni diferite (Quebec/Mississippi), la temperaturi diferite și luând în considerare cantitatea de CO2 din proximitatea plantelor.

	Plant	Type	Treatment	conc	uptake
1	Qn1	Quebec	nonchilled	95	16.0
2	Qn1	Quebec	nonchilled	175	30.4
3	Qn1	Quebec	nonchilled	250	34.8
4	Qn1	Quebec	nonchilled	350	37.2
5	Qn1	Quebec	nonchilled	500	35.3
6	Qn1	Quebec	nonchilled	675	39.2
7	Qn1	Quebec	nonchilled	1000	39.7
8	Qn2	Quebec	nonchilled	95	13.6

	Plant	Type	Treatment	conc	uptake
1	0	0	0	95	16.0
2	0	0	0	175	30.4
3	0	0	0	250	34.8
4	0	0	0	350	37.2
5	0	0	0	500	35.3
6	0	0	0	675	39.2
7	0	0	0	1000	39.7
8	1	0	0	95	13.6

Pentru simplitate, am atribuit fiecărei valori de tip șir de caractere o valoare numerică și am transformat coloanele respective de la tipul de dată *factor* la tipul de dată *numeric*.

```
model <- CO2
model$Plant <- as.character(model$Plant)
model$Plant[model$Plant=="Qn1"]<- 0
#[...]
model$Plant <- as.numeric(model$Plant)
```

Astfel, setul creat este identic cu cel original, însă planta *Qn1* a devenit *0*, *Qn2* a devenit *1*, ş. a. m. d., *Quebec* a luat valoarea *0*, *Mississippi* valoarea *1*, iar pentru valorile de sub coloana *Treatment*, *nonchilled* a devenit *0* şi *chilled* a devenit *1*. În continuare, vom folosi setul de date *model*.

Subiectul I

În continuare, am efectuat operaţii de statistică descriptivă pe diverse categorii de date din setul inițial.

```
#variabila “Quebec” va contine toate plantele din Quebec
Quebec <- subset(model,
                 Type==0, #Quebec
                 select=c(Plant, Treatment, conc, uptake))
#procesul este similar pentru toate celelalte categorii
```

Astfel, am calculat media absorbției în diferite situații şi am observat că locul din care provine planta este cel mai important factor pentru absorbția dioxidului de carbon, urmat de temperatura acesteia (prin urmare, în medie, o plantă din Mississippi cu temperatura scăzută va avea un randament mai mic de absorbție decât o planta din Quebec ce nu a avut temperatura scăzută).

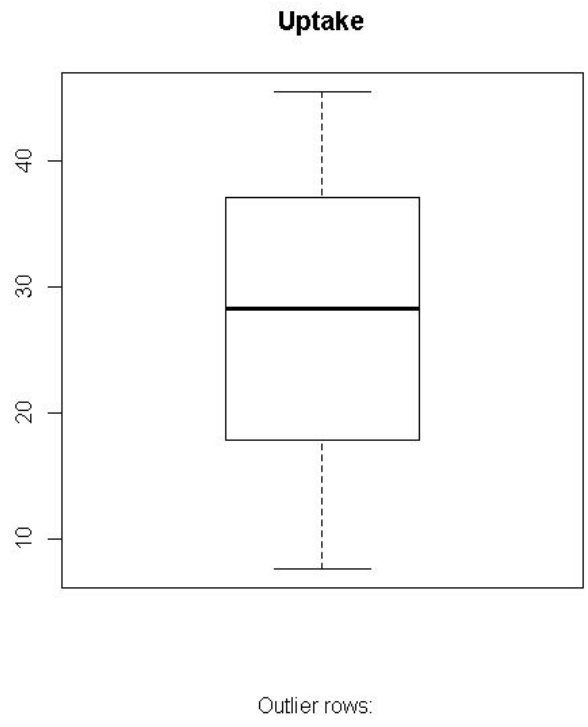
```
mean(model$uptake)                  # 27.21310 medie pentru toate plantele
mean(chilledMississippi$uptake)     # 15.81429
mean(nonchilledMississippi$uptake)  # 25.95238
mean(chilledQuebec$uptake)          # 31.75238
mean(nonchilledQuebec$uptake)       # 35.33333
```

Din calculul variației, putem afirma faptul că şi cantitatea de CO2 din proximitatea plantelor este un factor în absorbția acestuia.

```
#selectam planta Qc1 din Quebec, chilled si verificam varianta absorbtiei
#la diferite cantitati de CO2 din proximitate
chilledQuebec1 <- subset(model,
                         Plant==3       #Qc1
                         & Treatment==1 #chilled
                         & Type==0,     #Quebec
                         select=c(conc, uptake))

var(chilledQuebec1$uptake)             # 69.46571
```

Folosind funcţia boxplot, putem observa că mediana este aproape de mijloc, fapt ce sugerează o distribuție normală pentru absorbția de CO2.

```
boxplot(model$uptake,
      main="Uptake",
      sub=paste("Outlier rows: ",
                boxplot.stats(model$uptake)$out))
```



În continuare, am folosit funcția *quantile* pentru a afla mai multe date despre setul de date folosit.

```
quantile(model$uptake)
#   0%    25%    50%    75%   100%
# 7.700 17.900 28.300 37.125 45.500
```

Astfel, putem afirma că în 50% din cazuri, valoarea coloanei *uptake* este mai mică sau egală cu 28,3 şi nu scade niciodată sub 7,7. Deci, orice situație am alege, planta selectată absoarbe CO2.

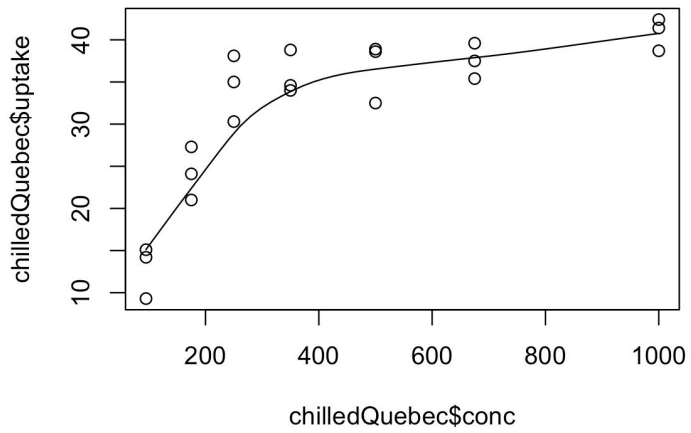
Subiectul II

Pentru a ne alege datele pentru **modelul de regresie liniară simplă**, am calculat corelaţiile dintre cantitatea de dioxid de carbon din atmosfera şi cantitatea de absorbţie a acestuia în toate cele 4 medii. Cea mai mare corelaţie a fost la plantele cu temperatura scăzută din Quebec.

```
cor(CO2$uptake, CO2$conc)                                    # 0.4851774
cor(chilledMississippi$uptake ,chilledMississippi$conc)      # 0.5586601
cor(nonchilledMississippi$uptake ,nonchilledMississippi$conc) # 0.7019991
cor(chilledQuebec$uptake, chilledQuebec$conc)                # 0.7422454
cor(nonchilledQuebec$uptake, nonchilledQuebec$conc)          # 0.7038936
```

Pentru a observa compatibilitatea cu un model de regresie liniară am folosit funcţia *scatter*, apoi am construit şi verificat modelul.

```
scatter.smooth(x=chilledQuebec$conc, y=chilledQuebec$uptake)
```



Odată ce am ales setul de date, l-am împărţit într-o proporție de 80-20 pentru a antrena modelul de regresie cu 80% din date, apoi să-l testăm cu restul de 20%. După realizarea modelului, am obținut următoarele valori care ne confirmă acurateţea regresiei:

- Valoarea lui *R squared* este 0.5291
- *Adjusted R-squared* este 0.4954
- *Std. Error* apropiat de zero: 0.005961
- *T-statistic* (3.966) mai mare decât 1.96 şi *p-value* (0.00141) mai mic decât 0.05
- Valoare mică pentru *AIC*: 110.6037
- Valoare mică pentru *BIC*: 112.9215
- Valoare mică pentru *MAPE* value: 0.21
- Valoare mare pentru *Min_Max Accuracy*: 0.84

În ceea ce priveşte **modelul de regresie liniară multiplă**, am obținut valori foarte bune deoarece am luat în considerare toţi parametrii pentru predicţia absorbției de dioxid de carbon. Se poate observa mai jos că toate valorile de referinţă precum *R squared (~0.70)* sunt mult mai bune decât la modelul de regresie liniară simplă.

```
> multLinMod <- lm(uptake ~ Type + Plant + Treatment + conc, data=model)
> summary(multLinMod)

Call:
lm(formula = uptake ~ Type + Plant + Treatment + conc, data = model)

Residuals:
    Min      1Q  Median      3Q     Max
-17.344  -2.860   1.424   4.288  10.765

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  28.758028   1.754319  16.393  < 2e-16 ***
Type        -15.670238   5.166684  -3.033  0.00328 **
Plant         0.501786   0.830890   0.604  0.54763
Treatment    -8.364881   2.838029  -2.947  0.00421 **
conc          0.017731   0.002306   7.688 3.53e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.218 on 79 degrees of freedom
Multiple R-squared:  0.6854,  Adjusted R-squared:  0.6694
F-statistic: 43.02 on 4 and 79 DF,  p-value: < 2.2e-16
```

Adăugarea unei noi variabile

Am ales sa luăm în considerare umiditatea din plantă ca noua variabilă ce influenţează modelul nostru de regresie liniară multiplă. O plantă nu trebuie sa fie secată, şi nici mult prea umedă pentru a putea absorbi dioxidul de carbon. Are nevoie de o cantitate moderată de apă. Din acest motiv am ales să folosim o distribuție normală cu ***m = 1*** şi ***sd = 5*** pentru a modela umiditatea plantelor. Astfel, am creat un nou model de regresie cu o variabilă nou adăugată.

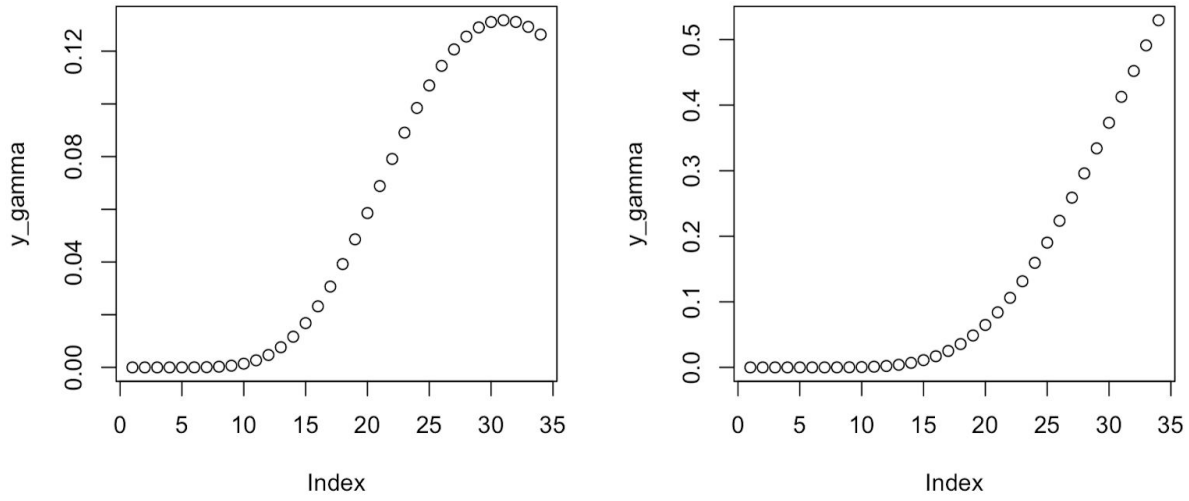
Putem observa că noua variabilă a ajutat la obținerea unui model de regresie cu o acurateţe şi mai bună, în care *R squared* are o valoare mai mare de *0.70.*

```
> set.seed(100)
> moistureRate <- rnorm(84, mean=1, sd=5)
> multLinMod <- lm(uptake ~ Type + Plant + Treatment + conc + moistureRate,
data=model)
> summary(multLinMod)

Call:
lm(formula = uptake ~ Type + Plant + Treatment + conc + moistureRate,
    data = model)

Residuals:
     Min       1Q   Median       3Q      Max
-18.1407  -3.6716   0.7423   4.0002  11.8977

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   28.167622   1.736137  16.224  < 2e-16 ***
Type         -15.544690   5.050152  -3.078  0.00287 **
Plant          0.546935   0.812363   0.673  0.50277
Treatment     -8.515156   2.774703  -3.069  0.00296 **
conc           0.017915   0.002256   7.942 1.22e-11 ***
moistureRate   0.289597   0.133599   2.168  0.03323 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.077 on 78 degrees of freedom
Multiple R-squared:  0.7032,  Adjusted R-squared:  0.6842
F-statistic: 36.97 on 5 and 78 DF,  p-value: < 2.2e-16
```

Subiectul III

Repartiția aleasă este repartiția Gamma. Motivul pentru care am ales-o este vasta utilitate a acesteia în domeniul ştiinţific. Printre aplicabilităţii şi domenii de utilizare se numără econometria, criptarea şi modelarea timpului de aşteptare. Parametrii repartiției Gamma sunt **k (shape)** si ***Θ* (scale)**. Putem observa din graficul funcţiei de densitate faptul că asmietria repartiției este dependentă de valoarea parametrului k (mai exact $\frac{2}{\sqrt{k}}$). Pentru a observa în R proprietățile repartiţiei am folosit funcţia *dgamma* pentru densitate şi funcţia *pgamma* pentru repartiție.

```r
par(mfrow=c(1, 2))               # format for two plots
x_gamma <- seq(0, 10, by=0.3)    # create sequence of numbers

y_gamma <- dgamma(x_gamma, 10)   # run density function
plot(y_gamma,)
y_gamma <- pgamma(x_gamma, 10)   # run distribution function
plot(y_gamma)
```



Bonus

Exercitiul 1

Problema generării unui tabel cu probabilități pentru o repartiție comună de variabile aleatoare discrete este asemenatoare cu problema rezolvării unui Sudoku. Această clasică problemă nu are o rezolvare în timp polinomial (este NP hard) şi am decis să rezolvăm cerinţa cu ajutorul algoritmului backtracking folosind câteva optimizări şi randomizări ale valorilor. Rulăm algoritmul până ce găsim prima soluție care respectă toate condițiile problemei. Rezultatul nu se va repeta pe același set de date, deoarece înainte de a începe, în vectorul de posibilităţi facem un “shuffle” al valorilor pentru a genera o soluţie “pseudo-randomizată”. Am inclus la începutul funcției mai multe variabile ce pot fi modificate pentru a genera o variabilă aleatoare după nevoie, presupunând că dispunem de timpul necesar găsirii unui rezultat.

```
precizie <- 0.1 #o zecimala
val_min  <- 0.1 #val min
val_max  <- 0.5 #val max
```

Astfel, construim repartiția comună complet, după care vom şterge câte un element de pe fiecare linie din interior. Rezultatul se află sub următoarea formă:

```
#      1   2   3  Y/pi
#1     NA 0.1 0.1  0.7
#2    0.1 0.1  NA  0.3
#X/qj 0.6 0.2 0.2  1.0
```

Exercitiul 2

Am construit funcția ***fcomplrepcom*** ce completează repartiția generată la punctul anterior, parcurgând-o linie cu linie şi calculând suma curentă pentru a determina valoarea elementului lipsă.

```
#      1   2   3  Y/pi
#1    0.5 0.1 0.1  0.7
#2    0.1 0.1 0.1  0.3
#X/qj 0.6 0.2 0.2  1.0
```

Exercitiul 3

Pentru calculul celor 3 subpuncte am folosit formulele învățate la curs pe care le-am aplicat pe modelul generat la primul exercițiu.

```
vaCov <- function(X, Y) {
  XY <- vaMult(X, Y) #inmultim variabilele
  cv <- E(XY) - E(X) * E(Y)
  cv
}
```

```
 # P(0<X<3/Y>2)
  for (i in 1:n)
     if (X[[1,i]] == 1 || X[[1,i]] == 2)
        result <- result + X[[2, i]]
```

```
# P(x > 6) = 1 - P(x <= 6) = 1 - F(6)
```

```
res <- paste("P(X>6,Y<7) =", (1 - fnF(X, 6)) * fnF(Y,7))
```

Exercitiul 4

La exercițiul 4 presupunem că repartiția comună este independentă şi parcurgem toate valorile pentru a verifica afirmaţia. Dacă am terminat de parcurs şi nu am găsit un contra-exemplu, presupunerea a fost corectă. Pentru funcția de corelație am aplicat formula matematică şi am analizat intervalul de corelație.

```
fverind <- function(repartitieComuna)
{
  #presupunem ca este
  este <- TRUE
  #[...]
  #verificam interiorul
  for (i in 1:n)
    for (j in 1:m)
    {
      rezultat <- repartitieComuna[[i, m + 1]]
                * repartitieComuna[[n + 1, j]]
      #daca valorile nu sunt egale, presupunerea facuta este falsa
      if (repartitieComuna[[i,j]] != rezultat)
        este <- FALSE
    }
  este
}
```

```
fvrnecor <- function(repartitieComuna)
{
  #construim X si Y de la repartitia comuna [...]
  vac <- vaCov(X,Y)  #calculam Cov(X, Y)
  rez <- all.equal(vac, 0)
  rez
}
```