

Curs Biostatistica 2017 - Laborator 7 & 8

Regresie

Contents

1	Regresie liniară simplă	1
1.1	Introducere	1
1.2	Exercițiul 1	3
2	Regresie liniară multiplă	16

1 Regresie liniară simplă



1.1 Introducere

Regresia liniară simplă (sau *modelul liniar simplu*) este un instrument statistic utilizat pentru a descrie relația dintre două variabile aleatoare, X (variabilă *cauză*, *predictor* sau *covariabilă*) și Y (variabilă *răspuns* sau *efect*) și este definit prin

$$\mathbb{E}[Y|X = x] = \beta_0 + \beta_1 x$$

sau altfel spus

$$Y = \beta_0 + \beta_1 X + \varepsilon.$$

În relațiile de mai sus, β_0 și β_1 sunt cunoscute ca ordonata la origine (*intercept*) și respectiv panta (*slope*) drepte de regresie.

Ipotezele modelului sunt:

- Linearitatea:** $\mathbb{E}[Y|X = x] = \beta_0 + \beta_1 x$
- Homoscedasticitatea:** $\text{Var}(\varepsilon_i) = \sigma^2$, cu σ^2 constantă pentru $i = 1, \dots, n$
- Normalitatea:** $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ pentru $i = 1, \dots, n$
- Independența erorilor:** $\varepsilon_1, \dots, \varepsilon_n$ sunt independente (sau necorelate, $\mathbb{E}[\varepsilon_i \varepsilon_j] = 0$, $i \neq j$, deoarece sunt presupuse normale)

Altfel spus

$$Y|X = x \sim \mathcal{N}(\beta_0 + \beta_1 x, \sigma^2)$$



- Nicio ipoteză nu a fost făcută asupra repartiției lui X (poate fi sau deterministă sau aleatoare)
- Modelul de regresie presupune că Y **este continuă** datorită normalității erorilor. În orice caz, X **poate fi o variabilă discretă!**

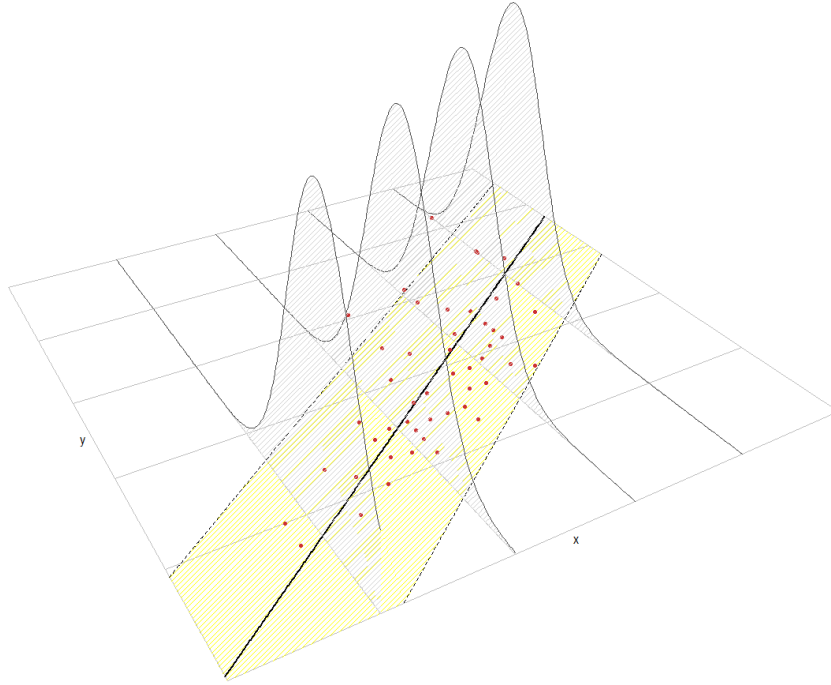


Figure 1: Regresia liniara simpla

Dat fiind un eșantion $(X_1, Y_1), \dots, (X_n, Y_n)$ pentru variabilele X și Y putem estima coeficienții necunoscuți β_0 și β_1 minimizând *suma abaterilor pătratice reziduale* (*Residual Sum of Squares* - RSS)

$$\text{RSS}(\beta_0, \beta_1) = \sum_{i=1}^n (Y_i - \beta_0 - \beta_1 X_i)^2$$

ceea ce conduce la

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}, \quad \hat{\beta}_1 = \frac{s_{xy}}{s_x^2} = \frac{\sum_{i=1}^n (X_i - \bar{X}) Y_i}{\sum_{i=1}^n (X_i - \bar{X})^2}$$

unde

- $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$ este *media eșantionului*
- $s_x^2 = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2$ este *varianța eșantionului*
- $s_{xy} = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})$ este *covarianța eșantionului*

Odată ce avem estimatorii $(\hat{\beta}_0, \hat{\beta}_1)$, putem defini:

- *valorile prognozate (fitted values)* $\hat{Y}_1, \dots, \hat{Y}_n$ (valorile verticale pe dreapta de regresie), unde

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i, \quad i = 1, \dots, n$$

- *reziduurile estimate (estimated residuals)* $\hat{\varepsilon}_1, \dots, \hat{\varepsilon}_n$ (distanțele verticale dintre punctele actuale (X_i, Y_i) și cele prognozate (X_i, \hat{Y}_i)), unde

$$\hat{\varepsilon}_i = Y_i - \hat{Y}_i, \quad i = 1, \dots, n$$

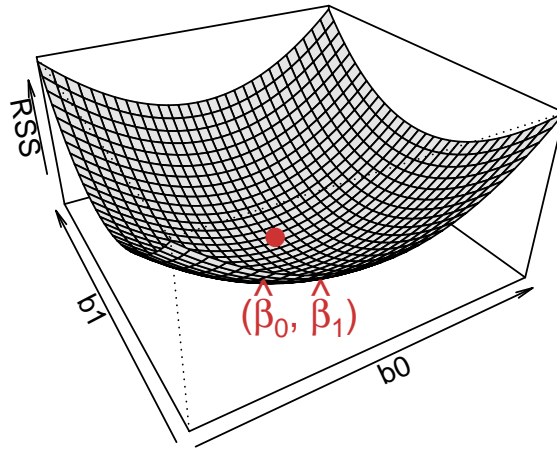


Figure 2: Graficul funcției RSS pentru modelul $y = -0.5 + 1.5x + e$.

Estimatorul pentru σ^2 este

$$\hat{\sigma}^2 = \frac{RSS(\hat{\beta}_0, \hat{\beta}_1)}{n-2} = \frac{\sum_{i=1}^n \hat{\epsilon}_i^2}{n-2}.$$

1.2 Exercițiul 1

În acest exercițiu vrem să investigăm relația dintre consumul de clorură de sodiu (sarea de bucătărie) și tensiunea arterială la persoanele trecute de 65 de ani. Pentru aceasta vom folosi setul de date `saltBP` care conține informații despre tensiunea arterială a 25 de pacienți.

Începem prin a înregistra setul de date

```
saltBP = read.table("data/saltBP.txt", header = T)
```

```
plot(saltBP$salt, saltBP$BP,
     xlab = "sare",
     ylab = "tensiunea arteriala",
     col = "brown3",
     pch = 16,
     bty="n")
```

```
summary(saltBP)
```

```
##          BP          salt          saltLevel
##  Min.   :128.3  Min.   : 1.130  Min.   :0.0
## 1st Qu.:131.8 1st Qu.: 2.650 1st Qu.:0.0
##  Median :135.7  Median : 5.210  Median :0.0
```

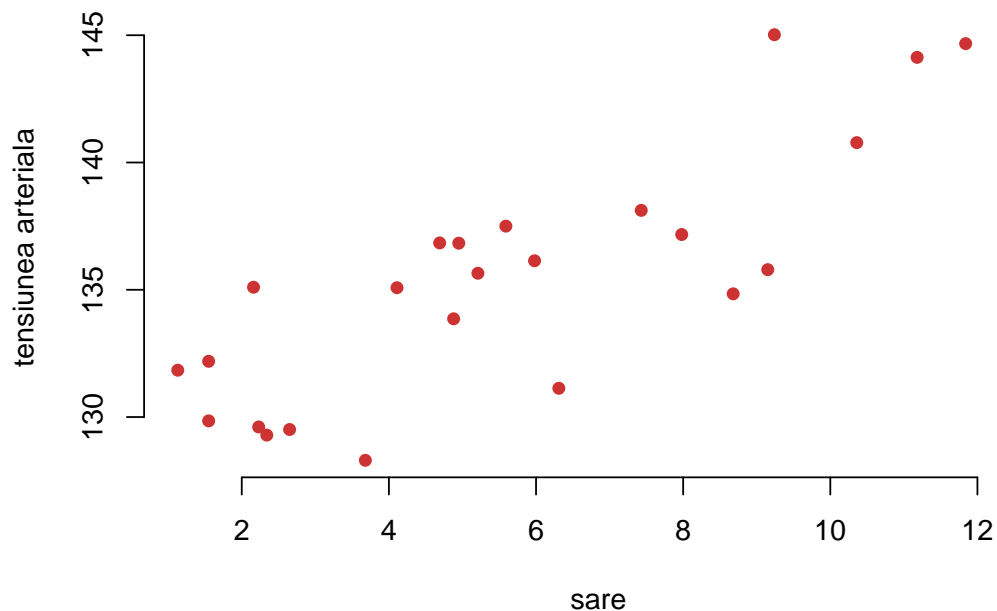


Figure 3: Diagrama de imprastiere

```
## Mean :135.7 Mean : 5.898 Mean :0.4
## 3rd Qu.:137.5 3rd Qu.: 8.680 3rd Qu.:1.0
## Max. :145.0 Max. :12.570 Max. :1.0
```

1.2.1 Estimarea parametrilor

Considerăm modelul de regresie $Y = \beta_0 + \beta_1 X + \varepsilon$ (unde $X = \text{saltBP}\$salt$ iar $Y = \text{saltBP}\$BP$), $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, a cărei parametrii sunt β_0 , β_1 și σ^2 .

- estimatorii parametrilor β_0 și β_1

```
# pentru b1
```

```
b1 = cov(saltBP$salt, saltBP$BP)/var(saltBP$salt)
cat("b1 = ", b1)
```

```
## b1 = 1.196894
```

```
# sau
```

```
sum((saltBP$salt-mean(saltBP$salt))*(saltBP$BP))/sum((saltBP$salt-mean(saltBP$salt))^2)
```

```
## [1] 1.196894
```

```
# pentru b0
```

```
b0 = mean(saltBP$BP) - b1*mean(saltBP$salt)
cat("b0 = ", b0)
```

```
## b0 = 128.6164
```

sau folosind functia lm:

```
saltBP_model = lm(BP~salt, data = saltBP)
names(saltBP_model)
```

```
## [1] "coefficients" "residuals"      "effects"      "rank"
## [5] "fitted.values" "assign"        "qr"          "df.residual"
## [9] "xlevels"      "call"         "terms"       "model"
```

```
saltBP_model$coefficients
```

```
## (Intercept)      salt
## 128.616397    1.196894
```

Dreapta de regresie este:

```
plot(saltBP$salt, saltBP$BP,
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     col = "brown3",
     pch = 16,
     bty="n",
     main = paste("y = ", format(b0, digits = 4), " + ", format(b1, digits = 4), " x"))

abline(a = b0, b = b1, col = "grey", lwd = 2)
points(mean(saltBP$salt), mean(saltBP$BP), pch = 16, col = "dark green", cex = 1.2)
text(mean(saltBP$salt), mean(saltBP$BP)-1.3, col = "dark green", cex = 1.2,
     labels = expression(paste("(", bar(x), ",", bar(y), ")")))
```

- estimatorul lui σ ($\hat{\sigma}$)

```
n = length(saltBP$BP)
e_hat = saltBP$BP - (b0+b1*saltBP$salt)

rss = sum(e_hat^2)

sigma_hat = sqrt(rss/(n-2))
sigma_hat
```

```
## [1] 2.745374
```

sau cu ajutorul functiei lm

```
sqrt(deviance(saltBP_model)/df.residual(saltBP_model))
```

```
## [1] 2.745374
```

sau încă

```
saltBP_model_summary = summary(saltBP_model)
# names(saltBP_model_summary)
saltBP_model_summary$sigma
```

```
## [1] 2.745374
```

1.2.2 Intervale de încredere pentru parametrii

Repartițiile lui $\hat{\beta}_0$ și $\hat{\beta}_1$ sunt

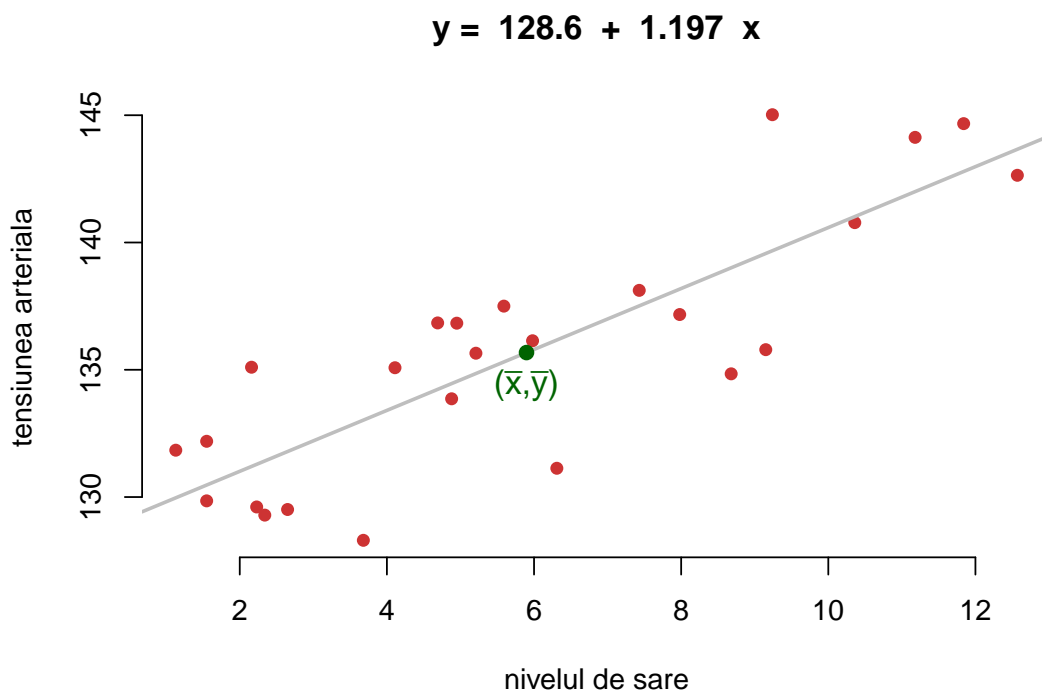


Figure 4: Dreapta de regresie

$$\hat{\beta}_0 \sim \mathcal{N}(\beta_0, \text{SE}(\hat{\beta}_0)^2), \quad \hat{\beta}_1 \sim \mathcal{N}(\beta_1, \text{SE}(\hat{\beta}_1)^2)$$

unde

$$\text{SE}(\hat{\beta}_0)^2 = \frac{\sigma^2}{n} \left[1 + \frac{\bar{X}^2}{s_x^2} \right], \quad \text{SE}(\hat{\beta}_1)^2 = \frac{\sigma^2}{n s_x^2}.$$

Folosind estimatorul $\hat{\sigma}^2$ pentru σ^2 obținem că

$$\frac{\hat{\beta}_0 - \beta_0}{\hat{\text{SE}}(\hat{\beta}_0)} \sim t_{n-2}, \quad \frac{\hat{\beta}_1 - \beta_1}{\hat{\text{SE}}(\hat{\beta}_1)} \sim t_{n-2}$$

unde

$$\hat{\text{SE}}(\hat{\beta}_0)^2 = \frac{\hat{\sigma}^2}{n} \left[1 + \frac{\bar{X}^2}{s_x^2} \right], \quad \hat{\text{SE}}(\hat{\beta}_1)^2 = \frac{\hat{\sigma}^2}{n s_x^2}$$

prin urmare, intervalele de încredere de nivel $1 - \alpha$ pentru β_0 și β_1 sunt

$$IC = \left(\hat{\beta}_j \pm \hat{\text{SE}}(\hat{\beta}_j) t_{n-2; \alpha/2} \right), \quad j = 0, 1.$$

```
alpha = 0.05

# trebuie avut grija ca functia var si sd calculeaza impartind la (n-1) si nu la n !!!
se_b0 = sqrt(sigma_hat^2*(1/n+mean(saltBP$salt)^2/((n-1)*var(saltBP$salt))))
se_b1 = sqrt(sigma_hat^2/((n-1)*var(saltBP$salt)))

lw_b0 = b0 - qt(1-alpha/2, n-2)*se_b0
up_b0 = b0 + qt(1-alpha/2, n-2)*se_b0

cat("CI pentru b0 este (", lw_b0, ", ", up_b0, ")\n")

## CI pentru b0 este ( 126.337 , 130.8958 )

lw_b1 = b1 - qt(1-alpha/2, n-2)*se_b1
up_b1 = b1 + qt(1-alpha/2, n-2)*se_b1

cat("CI pentru b1 este (", lw_b1, ", ", up_b1, ")")

## CI pentru b1 este ( 0.8617951 , 1.531993 )

Același rezultat se obține apelând funcția confint :
```

```
confint(saltBP_model)
```

```
##                2.5 %      97.5 %
## (Intercept) 126.3369606 130.895834
## salt        0.8617951  1.531993
```

Putem construi și o regiune de încredere pentru perechea (β_0, β_1) :

```
plot(ellipse(saltBP_model, c(1,2)), type = "l", col = "grey30",
      xlab = expression(beta[0]),
      ylab = expression(beta[1]),
      bty = "n")
points(coef(saltBP_model)[1], coef(saltBP_model)[2], pch = 18, col = "brown3")
abline(v = confint(saltBP_model)[1,], lty = 2)
abline(h = confint(saltBP_model)[2,], lty = 2)
```

1.2.3 ANOVA pentru regresie

Este predictorul X folositor în prezicerea răspunsului Y ? Vrem să testăm ipoteza nulă $H_0 : \beta_1 = 0$.

Introducem următoarele *sume de abateri pătratice*:

- $SS_T = \sum_{i=1}^n (Y_i - \bar{Y})^2$, **suma abaterilor pătratice totală** (variația totală a lui Y_1, \dots, Y_n).
- $SS_{reg} = \sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2$, **suma abaterilor pătratice de regresie** (variabilitatea explicată de dreapta de regresie)
- $RSS = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$, **suma abaterilor pătratice reziduale**

Avem următoarea descompunere ANOVA

$$\underbrace{SS_T}_{\text{Variația lui } Y_i} = \underbrace{SS_{reg}}_{\text{Variația lui } \hat{Y}_i} + \underbrace{RSS}_{\text{Variația lui } \hat{\varepsilon}_i}$$

și tabelul ANOVA corespunzător

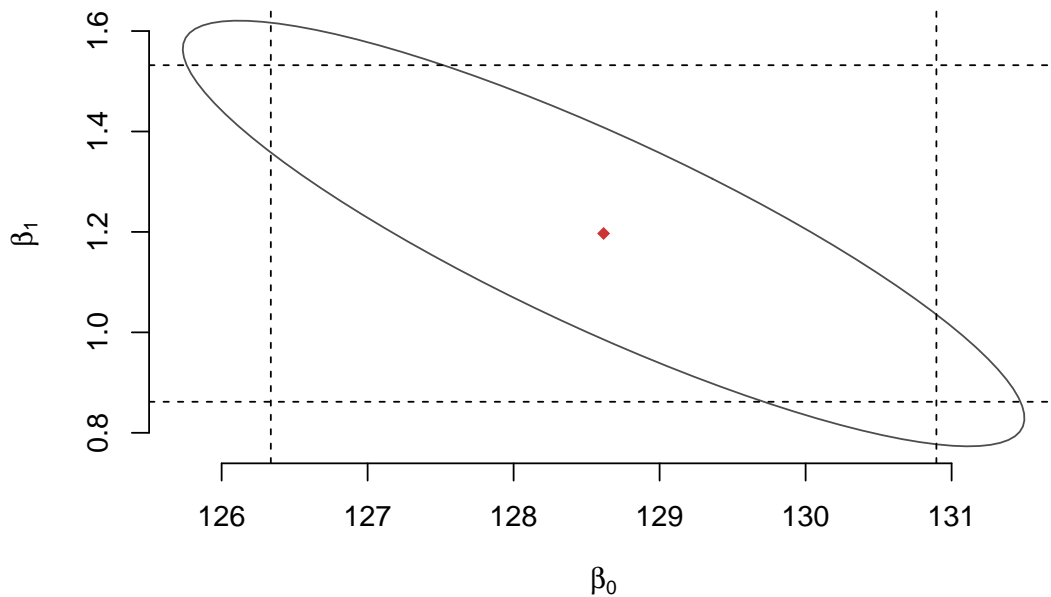


Figure 5: Regiune de incredere

	Df	SS	MS	F	p -value
Predictor	1	SS_{reg}	$\frac{SS_{reg}}{1}$	$\frac{SS_{reg}/1}{RSS/(n-2)}$	p
Residuuri	$n - 2$	RSS	$\frac{RSS}{n-2}$		

Descompunerea ANOVA pentru problema noastră poate fi ilustrată astfel:

- *suma abaterilor pătratice totală:*

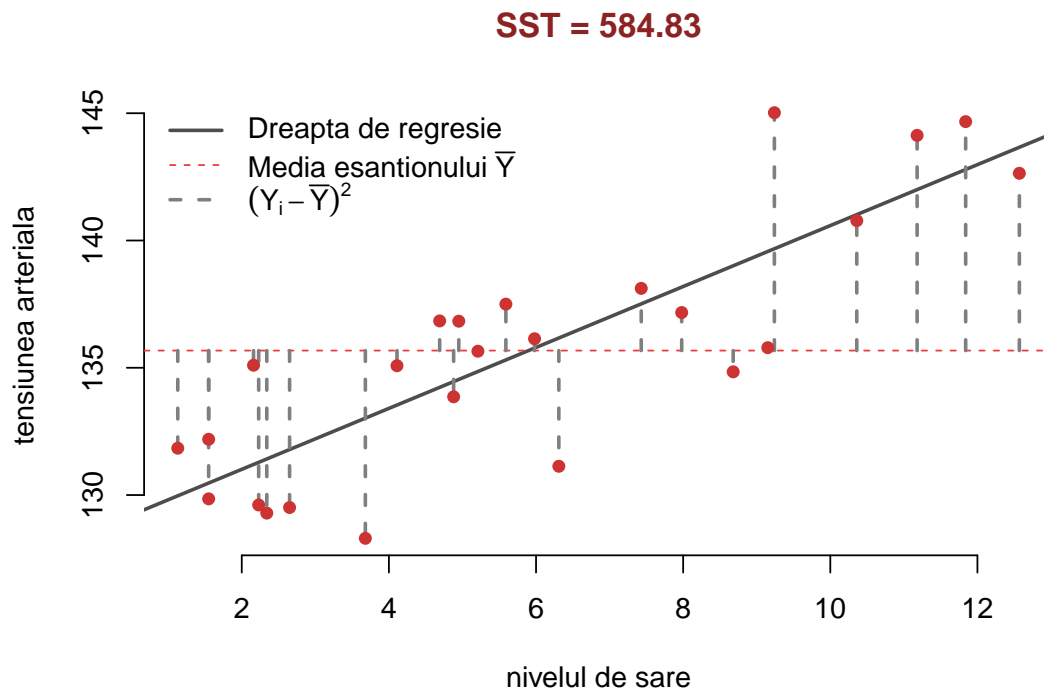
```
plot(saltBP$salt, saltBP$BP, pch = 16, type = "n",
     main = paste("SST =", round(sum((saltBP$BP - mean(saltBP$BP))^2), 2)),
     col.main = "brown4",
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     bty = "n")

abline(saltBP_model$coefficients, col = "grey30", lwd = 2)
abline(h = mean(saltBP$BP), col = "brown2", lty = 2)

segments(x0 = saltBP$salt, y0 = mean(saltBP$BP), x1 = saltBP$salt, y1 = saltBP$BP,
         col = "grey50", lwd = 2, lty = 2)

legend("topleft", legend = expression("Dreapta de regresie", "Media esantionului " * bar(Y),
                                     (Y[i] - bar(Y))^2),
       lwd = c(2, 1, 2),
       col = c("grey30", "brown2", "grey50"),
       lty = c(1, 2, 2),
       bty = "n")

points(saltBP$salt, saltBP$BP, pch = 16, col = "brown3")
```

- suma abaterilor pătratice de regresie

```
plot(saltBP$salt, saltBP$BP, pch = 16, type = "n",
     main = paste("SSreg =", round(sum((saltBP_model$fitted.values - mean(saltBP$BP))^2), 2)),
     col.main = "forestgreen",
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     bty = "n")

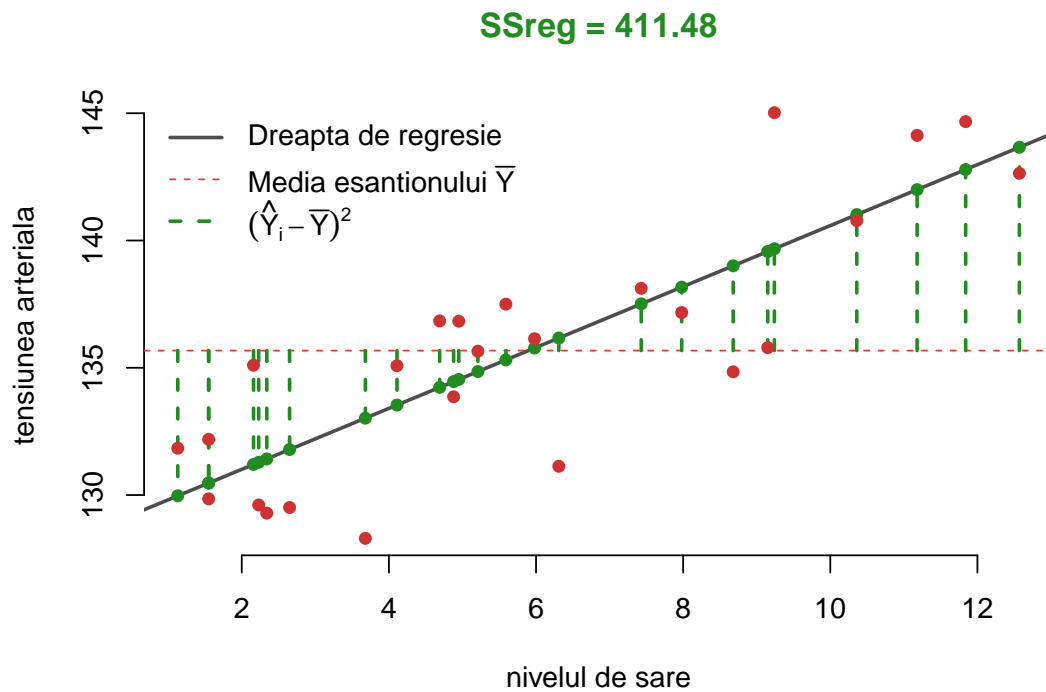
abline(saltBP_model$coefficients, col = "grey30", lwd = 2)
abline(h = mean(saltBP$BP), col = "brown2", lty = 2)

segments(x0 = saltBP$salt, y0 = mean(saltBP$BP), x1 = saltBP$salt, y1 = saltBP_model$fitted.values,
         col = "forestgreen", lwd = 2, lty = 2)

points(saltBP$salt, saltBP_model$fitted.values, pch = 16, col = "forestgreen")

legend("topleft", legend = expression("Dreapta de regresie", "Media esantionului " * bar(Y),
                                     (hat(Y)[i] - bar(Y))^2),
       lwd = c(2, 1, 2),
       col = c("grey30", "brown2", "forestgreen"),
       lty = c(1, 2, 2),
       bty = "n")

points(saltBP$salt, saltBP$BP, pch = 16, col = "brown3")
```



- suma abaterilor pătratic reziduale

```
plot(saltBP$salt, saltBP$BP, pch = 16, type = "n",
     main = paste("RSS =", round(sum((saltBP$BP - saltBP_model$fitted.values)^2), 2)),
     col.main = "orange",
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     bty = "n")

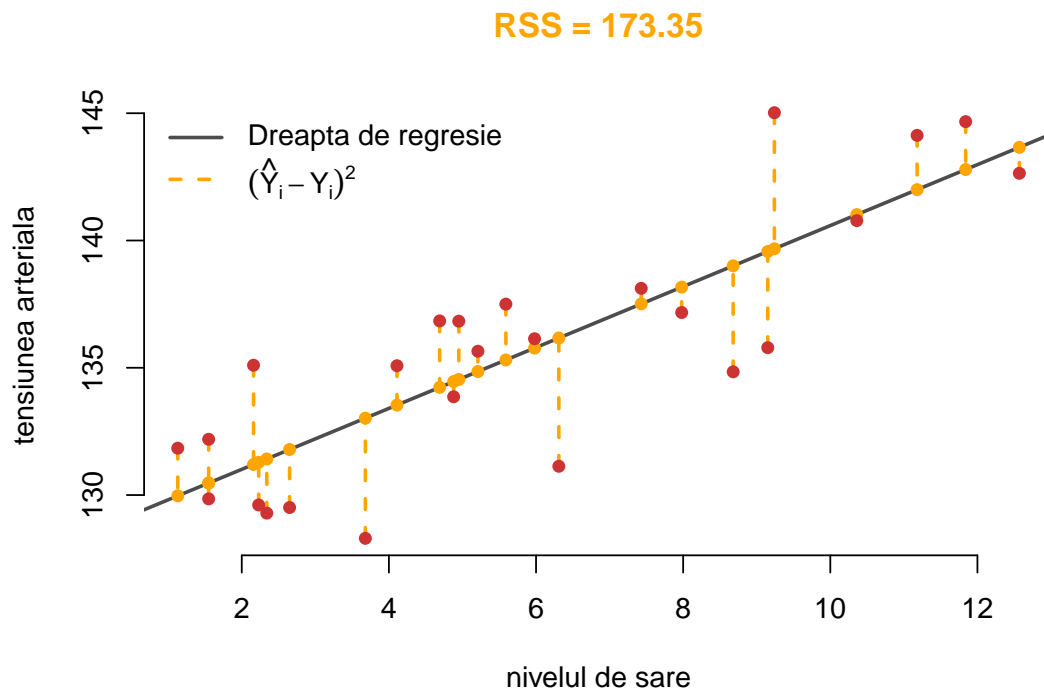
abline(saltBP_model$coefficients, col = "grey30", lwd = 2)

segments(x0 = saltBP$salt, y0 = saltBP$BP, x1 = saltBP$salt, y1 = saltBP_model$fitted.values,
         col = "orange", lwd = 2, lty = 2)

points(saltBP$salt, saltBP_model$fitted.values, pch = 16, col = "orange")

legend("topleft", legend = expression("Dreapta de regresie",  $(\hat{Y}[i] - \bar{Y})^2$ ),
      lwd = c(2, 2),
      col = c("grey30", "orange"),
      lty = c(1, 2),
      bty = "n")

points(saltBP$salt, saltBP$BP, pch = 16, col = "brown3")
```



Tabelul ANOVA se obține prin

```
# tabel ANOVA
anova(saltBP_model)
```

```
## Analysis of Variance Table
##
## Response: BP
##          Df Sum Sq Mean Sq F value    Pr(>F)
## salt      1 411.48   411.48   54.594 1.631e-07 ***
## Residuals 23 173.35     7.54
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1.2.4 Inferență asupra parametrilor

Este predictorul X folositor în prezicerea răspunsului Y ? Vrem să testăm ipoteza nulă $H_0: \beta_j = 0$ (pentru $j = 1$ spunem că predictorul **nivel de sare** nu are un efect *liniar* semnificativ asupra **tensiunii arteriale**). Pentru aceasta vom folosi statistica de test

$$t_j = \frac{\hat{\beta}_j}{\widehat{SE}(\hat{\beta}_j)} \sim_{H_0} t_{n-2}.$$

Funcția `summary` ne întoarce p -valoarea corespunzătoare a acestor teste:

```
summary(saltBP_model)
```

```
##
## Call:
```

```
## lm(formula = BP ~ salt, data = saltBP)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.0388 -1.6755  0.3662  1.8824  5.3443
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  128.616      1.102  116.723 < 2e-16 ***
## salt          1.197       0.162   7.389 1.63e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.745 on 23 degrees of freedom
## Multiple R-squared:  0.7036, Adjusted R-squared:  0.6907
## F-statistic: 54.59 on 1 and 23 DF,  p-value: 1.631e-07
```

Observăm că ambele ipoteze sunt respinse în favoarea alternativelor bilaterale (la aceeași concluzie am ajuns și utitându-ne la intervalele de încredere - nu conțineau valoarea 0). Putem observa că t_1^2 este exact valoarea F statisticii, deci cele două abordări ne dau aceleași rezultate numerice.

1.2.5 Predicții

Pentru un nou set de predictor, x_0 , răspunsul prognozat este $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_0$ și vrem să investigăm incertitudinea din această predicție. Putem face distincția între două tipuri de predicție: predicție asupra răspunsului viitor mediu (inferență asupra mediei condiționate $\mathbb{E}[Y|X = x_0]$) sau predicție asupra observațiilor viitoare (inferență asupra răspunsului condiționat $Y|X = x_0$).

Un interval de încredere pentru răspunsul viitor mediu este:

$$\left(\hat{y} \pm t_{n-2;\alpha/2} \sqrt{\frac{\hat{\sigma}^2}{n} \left(1 + \frac{(x_0 - \bar{x})^2}{s_x^2} \right)} \right)$$

Un interval de încredere pentru valoarea prezisă (interval de predicție) este:

$$\left(\hat{y} \pm t_{n-2;\alpha/2} \sqrt{\hat{\sigma}^2 + \frac{\hat{\sigma}^2}{n} \left(1 + \frac{(x_0 - \bar{x})^2}{s_x^2} \right)} \right)$$

Pentru a găsi aceste intervale vom folosi funcția `predict`:

```
newData = data.frame(salt = 14)
newData2 = data.frame(salt = c(13, 14, 15))

# Predictie
predict(saltBP_model, newdata = newData)

##           1
## 145.3729

# Predictie pentru valoarea raspunsului mediu
predict(saltBP_model, newdata = newData, interval = "confidence")

##           fit           lwr           upr
## 1 145.3729 142.4298 148.316
```

```

predict(saltBP_model, newdata = newData2, interval = "confidence")

##          fit          lwr          upr
## 1 144.1760 141.5389 146.8132
## 2 145.3729 142.4298 148.3160
## 3 146.5698 143.3150 149.8246

# Predictie asupra observatiilor viitoare
predict(saltBP_model, newdata = newData, interval = "prediction")

##          fit          lwr          upr
## 1 145.3729 138.9764 151.7695

predict(saltBP_model, newdata = newData2, interval = "prediction")

##          fit          lwr          upr
## 1 144.1760 137.9144 150.4377
## 2 145.3729 138.9764 151.7695
## 3 146.5698 140.0240 153.1156

g = seq(1,15,0.5)

p = predict(saltBP_model, data.frame(salt = g), se = T, interval = "confidence")
matplot(g, p$fit, type = "l", lty = c(1,2,2),
        lwd = c(2,1,1),
        col = c("grey30", "grey50", "grey50"),
        xlab = "nivelul de sare",
        ylab = "tensiunea arteriala",
        bty = "n")
rug(saltBP$salt)
points(saltBP$salt, saltBP$BP, col = "brown3", pch = 16)
abline(v = mean(saltBP$salt), lty = 3, col = "grey65")

# Scheffe's bounds
M = sqrt(2*qf(1-alpha, 2, n-2))

s_xx = (n-1)*var(saltBP$salt)
lw_scheffe = b0 + b1*g - M*sigma_hat*sqrt(1/n+(g-mean(saltBP$salt))^2/s_xx)
up_scheffe = b0 + b1*g + M*sigma_hat*sqrt(1/n+(g-mean(saltBP$salt))^2/s_xx)

lines(g, lw_scheffe, lty = 4, col = "brown4")
lines(g, up_scheffe, lty = 4, col = "brown4")

# Bonferroni bounds
# x0 = c(7, 8, 13, 14)
x0 = 1 + 14*runif(6)
m = length(x0)

t_bonf = qt(1-alpha/(2*m), n-2)

lw_bonf = b0 + b1*x0 - t_bonf*sigma_hat*sqrt(1/n+(x0-mean(saltBP$salt))^2/s_xx)
up_bonf = b0 + b1*x0 + t_bonf*sigma_hat*sqrt(1/n+(x0-mean(saltBP$salt))^2/s_xx)

segments(x0 = x0, y0 = lw_bonf, x1 = x0, y1 = up_bonf, col = "orange", lty = 5)
segments(x0 = x0-0.25, y0 = lw_bonf, x1 = x0+0.25, y1 = lw_bonf, col = "orange", lty = 1)
segments(x0 = x0-0.25, y0 = up_bonf, x1 = x0+0.25, y1 = up_bonf, col = "orange", lty = 1)

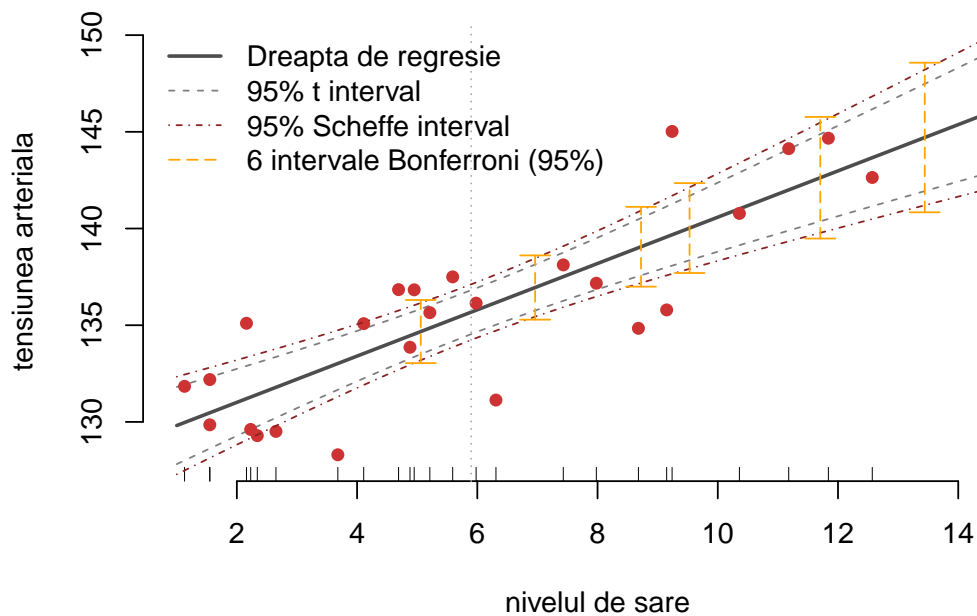
```

```

legend("topleft", legend = c("Dreapta de regresie", "95% t interval",
                             "95% Scheffe interval", paste0(m, " intervale Bonferroni (95%)")),
      lwd = c(2, 1, 1, 1),
      col = c("grey30", "grey50", "brown4", "orange"),
      lty = c(1, 2, 4, 5),
      bty = "n")

```

```
\begin{figure}
```



```
{
```

```
}
```

```
\caption{Nivelul de sare prezis impreuna cu intervalul de incredere de nivel 95% pentru raspunsul mediu}
\end{figure}
```

1.2.6 Diagnostic

În această secțiune vom vedea dacă setul nostru de date verifică ipotezele modelului de regresie liniară.

- *Independența*

Ipoteza de independență a variabilei răspuns (prin urmare și a erorilor) reiese, de cele mai multe ori, din modalitatea în care s-a desfășurat experimentul.

- *Normalitatea*

Pentru a verifica dacă ipoteza de normalitate a erorilor este satisfăcută vom trasa dreapta lui Henry (sau Q-Q plot-ul):

```
library(car)
```

```
##
```

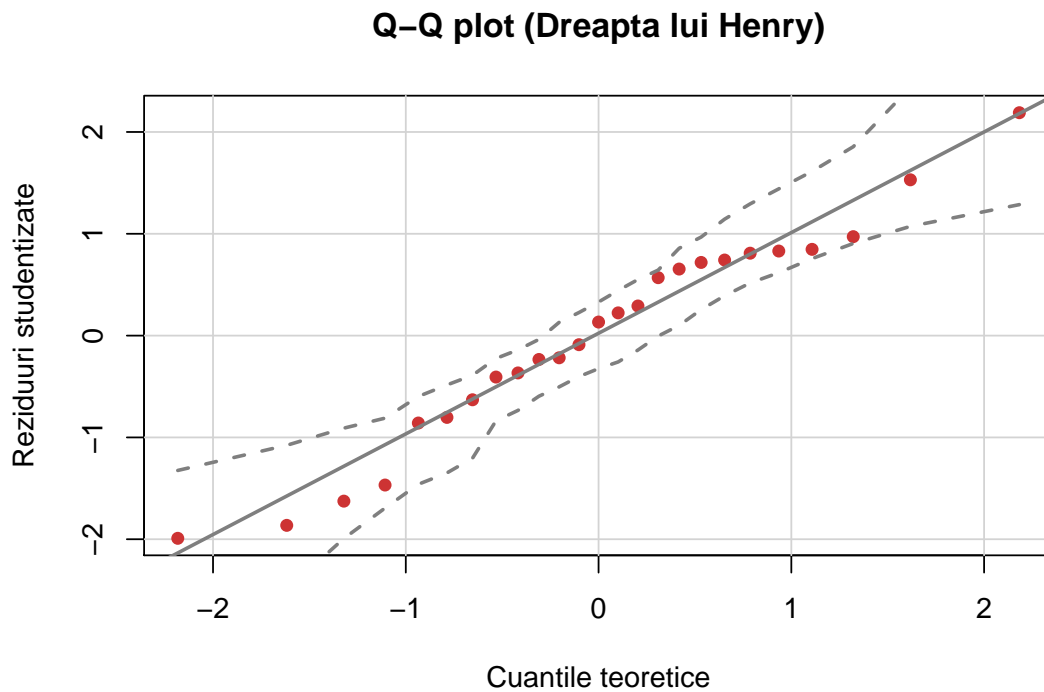


Figure 6: Q-Q plot

```
## Attaching package: 'car'
## The following object is masked from 'package:ellipse':
##
##   ellipse
qqPlot(saltBP_model, col = "brown3", col.lines = "grey50", pch = 16,
       simulate = TRUE,
       xlab = "Cuantile teoretice",
       ylab = "Reziduuri studentizate",
       main = "Q-Q plot (Dreapta lui Henry)")
```

Putem folosi și testul Shapiro-Wilk:

```
shapiro.test(residuals(saltBP_model))
```

```
##
## Shapiro-Wilk normality test
##
## data: residuals(saltBP_model)
## W = 0.96871, p-value = 0.6125
```

- *Homoscedasticitatea*

Pentru a verifica proprietatea de homoscedasticitate a erorilor vom trasa un grafic al reziduurilor versus valorile prezise (fitted), i.e. $\hat{\varepsilon}$ vs \hat{y} . Dacă avem homoscedasticitate a erorilor atunci ar trebui să vedem o variație constantă pe verticală ($\hat{\varepsilon}$).

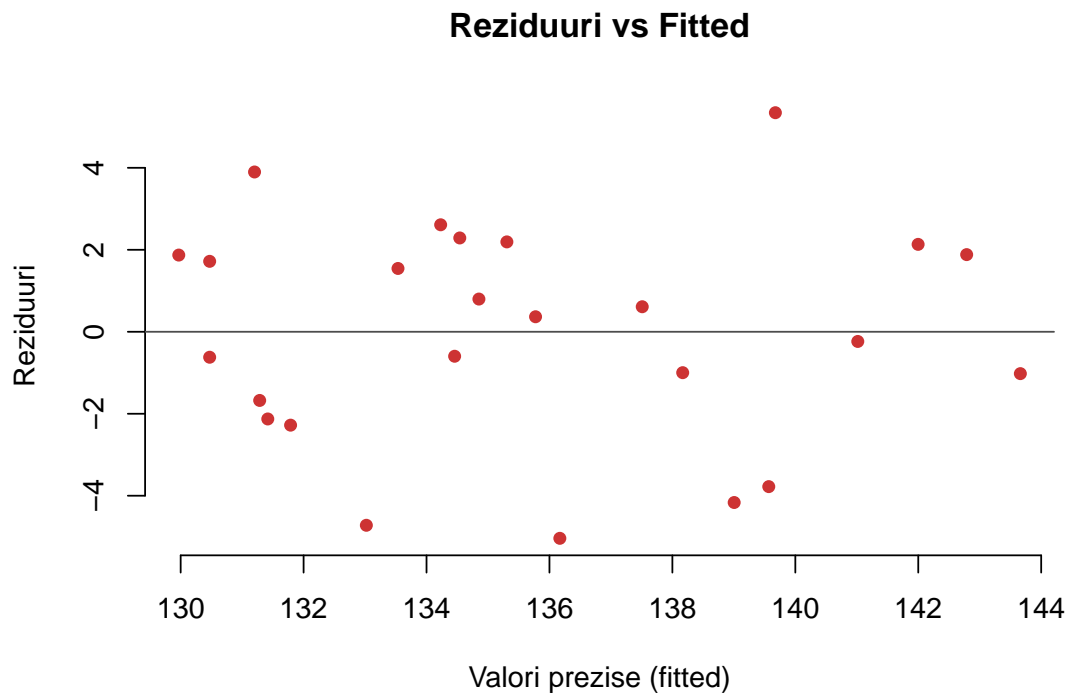


Figure 7: Reziduuri vs Valori prezise (Fitted)

```
plot(residuals(saltBP_model)~fitted(saltBP_model), col = "brown3", pch = 16,
     xlab = "Valori prezise (fitted)",
     ylab = "Reziduuri",
     main = "Reziduuri vs Fitted",
     bty = "n")

abline(h = 0, col = "grey30")
```

Tot în acest grafic putem observa dacă ipoteza de liniaritate este verificată (în caz de liniaritate între variabila răspuns și variabila cauză nu are trebui să vedem o relație sistematică între reziduuri și valorile prezise - ceea ce se și întâmplă în cazul nostru) ori dacă există o altă legătură structurală între variabila dependentă (răspuns) și cea independentă (predictor).

2 Regresie liniară multiplă