Laborator 5

Elemente de regresie liniară simplă

Obiectivul acestui laborator este de a prezenta câteva exemple legate de problema de regresie liniară simplă.

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*) dreptei de regresie.

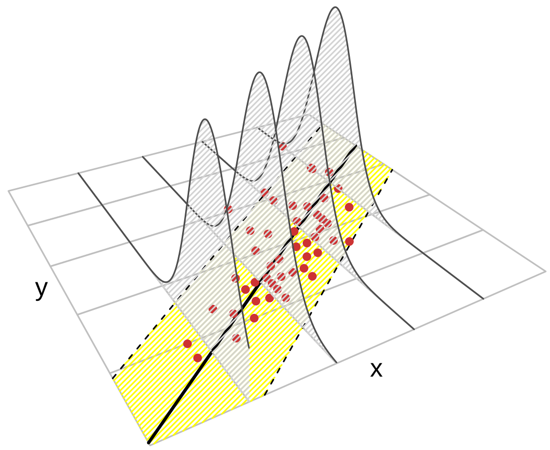
Ipotezele modelului sunt:

i. **Linearitatea**: $\mathbb{E}[Y|X = x] = \beta_0 + \beta_1 x$
ii. **Homoscedasticitatea**: $\mathbb{V}\text{ar}(\varepsilon_i) = \sigma^2$, cu σ^2 constantă pentru $i = 1, \dots, n$
iii. **Normalitatea**: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ pentru $i = 1, \dots, n$
iv. **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**!



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)

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

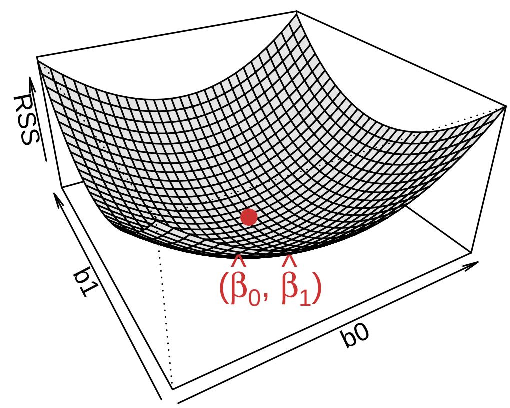
ceea ce conduce la

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

unde folosim notațiile

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

Graficul functiei RSS pentru modelul $y = -0.5 + 1.5x + e$:



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

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

Estimatorul pentru σ^2 este

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

2 Aplicație

Ne propunem să investigăm relația dintre volumul vânzărilor dintr-un anumit produs (calculate în mii de unități) și bugetul (în milioane RON) alocat pentru publicitatea la televizor. Pentru aceasta vom folosi setul de date advertising care conține informații despre volumul vânzărilor și bugetul alocat publicității TV în 200 de piețe de desfacere.

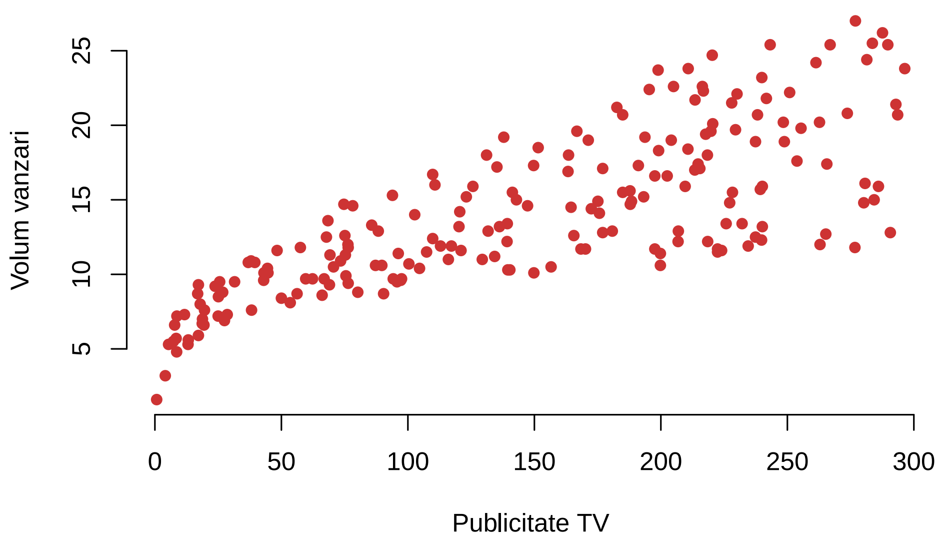
Începem prin a înregistra setul de date

```
advertise = read.csv("dataIn/advertising.csv", row.names = 1)

summary(advertise)
       TV              radio           newspaper          sales
```

```
 Min.   :  0.70   Min.   : 0.000   Min.   :  0.30   Min.   : 1.60  
 1st Qu.: 74.38   1st Qu.: 9.975   1st Qu.: 12.75   1st Qu.:10.38  
 Median :149.75   Median :22.900   Median : 25.75   Median :12.90  
 Mean   :147.04   Mean   :23.264   Mean   : 30.55   Mean   :14.02  
 3rd Qu.:218.82   3rd Qu.:36.525   3rd Qu.: 45.10   3rd Qu.:17.40  
 Max.   :296.40   Max.   :49.600   Max.   :114.00   Max.   :27.00  
```

și a ilustra grafic diagrama de împrăștiere

```r
plot(advertise$TV, advertise$sales,
     xlab = "Publicitate TV",
     ylab = "Volum vanzari",
     col = "brown3",
     pch = 16,
     bty="n")
```



2.1 Estimarea parametrilor

Considerăm modelul de regresie $Y = \beta_0 + \beta_1 X + \varepsilon$ (unde $X =$advertise$TV iar $Y =$advertise$sales), $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, a cărui parametrii sunt β_0, β_1 și σ^2.

Observăm că estimatorii parametrilor β_0 și β_1 sunt

```r
# pentru b1

b1 = cov(advertise$TV, advertise$sales)/var(advertise$TV)
cat("b1 = ", b1, "\n")
b1 =  0.04753664

# sau

sum((advertise$TV-mean(advertise$TV))*(advertise$sales))/sum((advertise$TV-mean(advertise$TV))^2)
```

```
[1] 0.04753664

# pentru b0

b0 = mean(advertise$sales) - b1*mean(advertise$TV)
cat("b0 = ", b0)
b0 =  7.032594
```

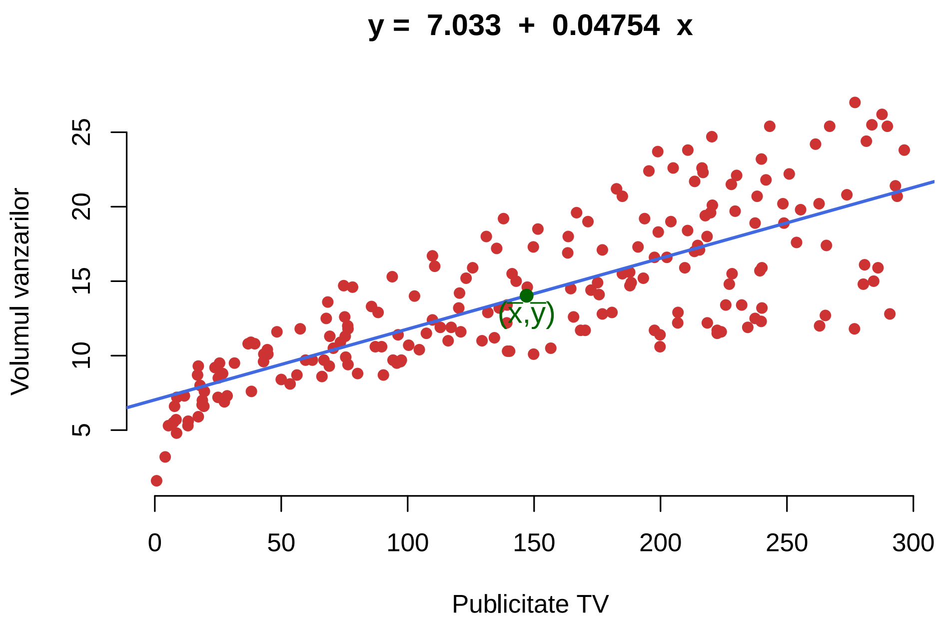
sau folosind funcția `lm()`:

```
advertise_TV_model = lm(sales~TV, data = advertise)
names(advertise_TV_model)
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "xlevels"       "call"          "terms"         "model"
```

```
advertise_TV_model$coefficients
(Intercept)          TV
 7.03259355  0.04753664
```

Dreapta de regresie este:

```
plot(advertise$TV, advertise$sales,
     xlab = "Publicitate TV",
     ylab = "Volumul vanzarilor",
     col = "brown3",
     pch = 16,
     bty="n",
     main = paste("y = ", format(b0, digits = 4), " + ", format(b1, digits = 4), " x"))

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



De asemenea pentru calculul estimatorului lui σ ($\hat{\sigma}$) avem

```
n = length(advertise$sales)
e_hat = advertise$sales - (b0+b1*advertise$TV)

rss = sum(e_hat^2)

sigma_hat = sqrt(rss/(n-2))
sigma_hat
[1] 3.258656
```

sau cu ajutorul funcției `lm()`

```
sqrt(deviance(advertise_TV_model)/df.residual(advertise_TV_model))
[1] 3.258656
```

sau încă

```
advertise_TV_model_summary = summary(advertise_TV_model)

advertise_TV_model_summary$sigma
[1] 3.258656
```

2.2 Intervale de încredere pentru parametrii

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

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

unde

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

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

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

unde

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

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

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

În R avem

```
alpha = 0.05

# trebuie avut grija ca functia var si sd se calculeaza
# impartind la (n-1) si nu la n !!!

se_b0 = sqrt(sigma_hat^2*(1/n+mean(advertise$TV)^2/((n-1)*var(advertise$TV))))
se_b1 = sqrt(sigma_hat^2/((n-1)*var(advertise$TV)))

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 ( 6.129719 ,  7.935468 )

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 ( 0.04223072 ,  0.05284256 )
```

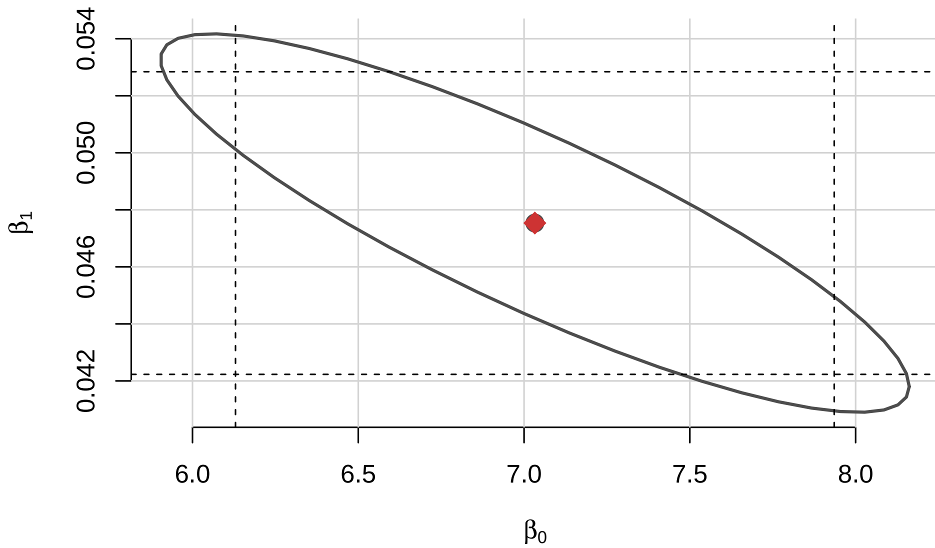
Același rezultat se obține apelând funcția `confint()` :

```
confint(advertise_TV_model)
                 2.5 %     97.5 %
(Intercept) 6.12971927 7.93546783
TV          0.04223072 0.05284256
```

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

```
library(ellipse)

par(bty = "n")
confidenceEllipse(advertise_TV_model,
                  xlab = expression(beta[0]),
                  ylab = expression(beta[1]),
```

```
                 col = "grey30")
points(coef(advertise_TV_model)[1], coef(advertise_TV_model)[2],
       pch = 18, col = "brown3",
       cex = 2)
abline(v = confint(advertise_TV_model)[1,], lty = 2)
abline(h = confint(advertise_TV_model)[2,], lty = 2)
```



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*:

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

Avem următoarea descompunere ANOVA

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

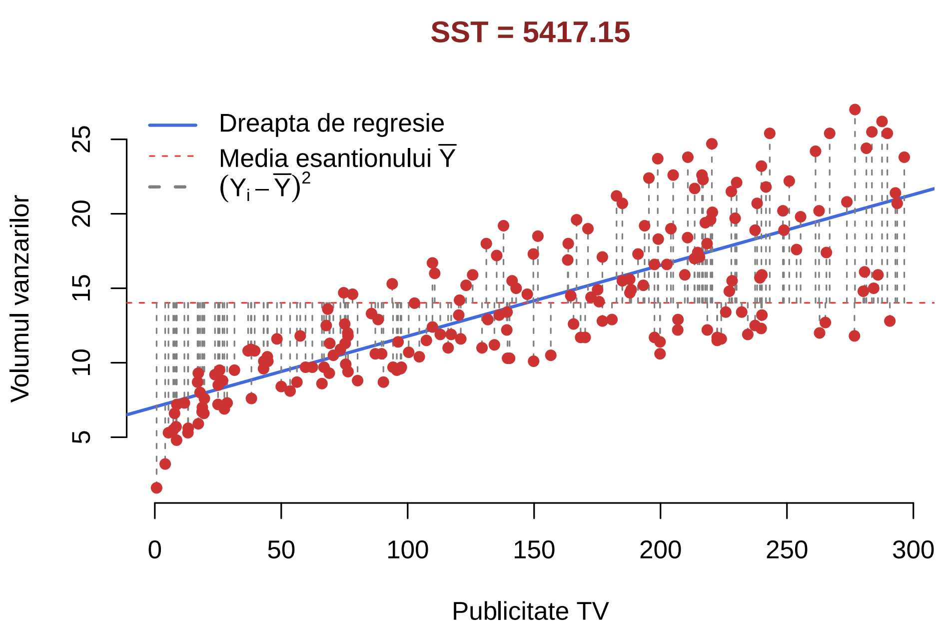
și tabelul ANOVA corespunzător

	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:

a) *suma abaterilor pătratice totală*:

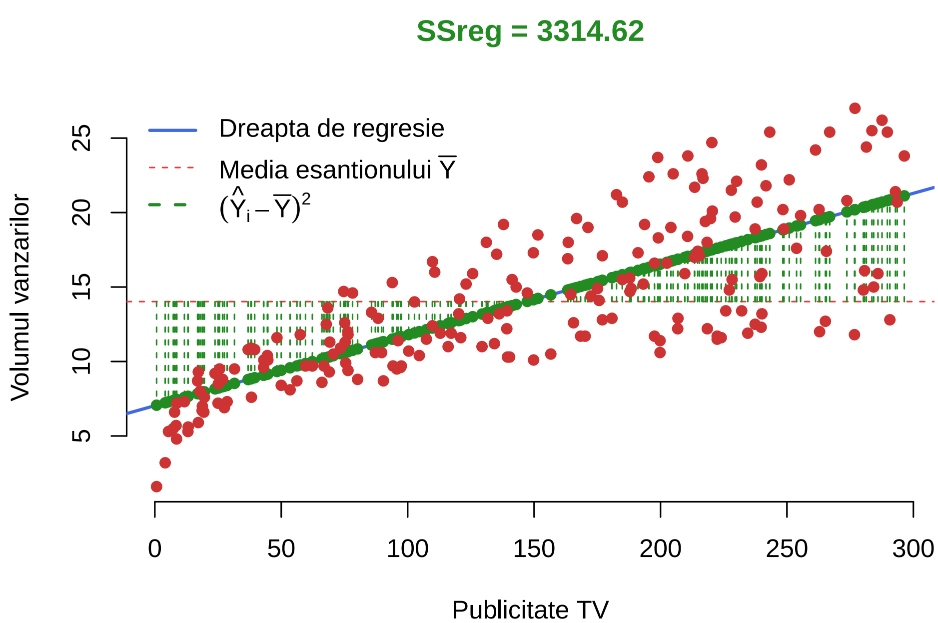
```
plot(advertise$TV, advertise$sales, pch = 16, type = "n",
     main = paste("SST =", round(sum((advertise$sales - mean(advertise$sales))^2), 2)),
     col.main = "brown4",
     xlab = "Publicitate TV",
     ylab = "Volumul vanzarilor",
     bty = "n")

abline(advertise_TV_model$coefficients, col = "royalblue", lwd = 2)
abline(h = mean(advertise$sales), col = "brown2", lty = 2)

segments(x0 = advertise$TV, y0 = mean(advertise$sales),
         x1 = advertise$TV, y1 = advertise$sales,
         col = "grey50", lwd = 1, 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("royalblue", "brown2", "grey50"),
       lty = c(1, 2, 2),
       bty = "n")

points(advertise$TV, advertise$sales, pch = 16, col = "brown3")
```



b) *suma abaterilor pătratice de regresie*

```
plot(advertise$TV, advertise$sales, pch = 16, type = "n",
     main = paste("SSreg =",
                  round(sum((advertise_TV_model$fitted.values - mean(advertise$sales))^2), 2)),
```

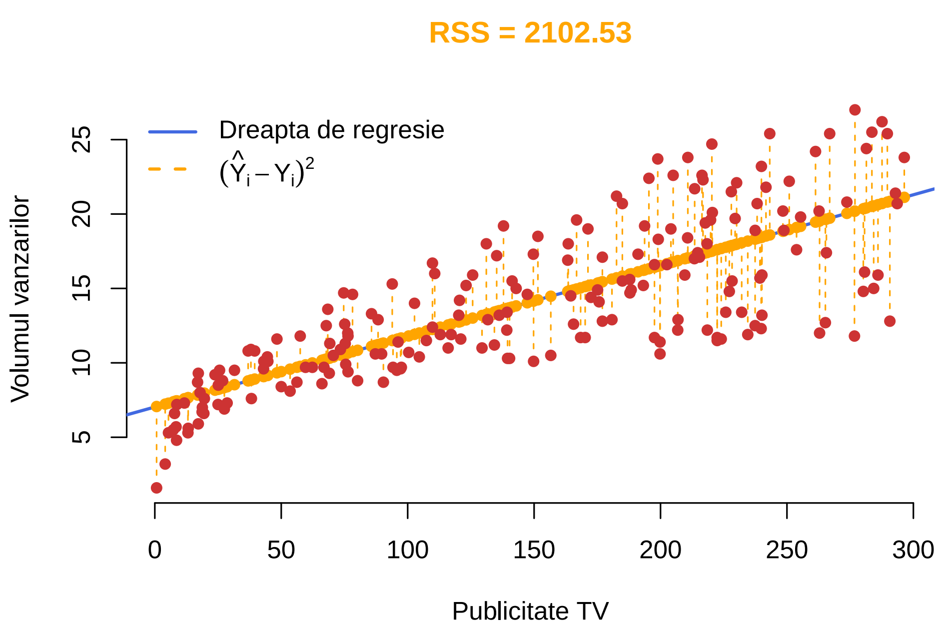
```
     col.main = "forestgreen",
     xlab = "Publicitate TV",
     ylab = "Volumul vanzarilor",
     bty = "n")

abline(advertise_TV_model$coefficients, col = "royalblue", lwd = 2)
abline(h = mean(advertise$sales), col = "brown2", lty = 2)

segments(x0 = advertise$TV, y0 = mean(advertise$sales),
         x1 = advertise$TV, y1 = advertise_TV_model$fitted.values,
         col = "forestgreen", lwd = 1, lty = 2)

points(advertise$TV, advertise_TV_model$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("royalblue", "brown2", "forestgreen"),
       lty = c(1, 2, 2),
       bty = "n")

points(advertise$TV, advertise$sales, pch = 16, col = "brown3")
```



c) *suma abaterilor pătratice reziduale*

```
plot(advertise$TV, advertise$sales, pch = 16, type = "n",
     main = paste("RSS =",
                  round(sum((advertise$sales - advertise_TV_model$fitted.values)^2), 2)),
     col.main = "orange",
     xlab = "Publicitate TV",
     ylab = "Volumul vanzarilor",
```

```r
     bty = "n")

abline(advertise_TV_model$coefficients, col = "royalblue", lwd = 2)

segments(x0 = advertise$TV, y0 = advertise$sales,
         x1 = advertise$TV, y1 = advertise_TV_model$fitted.values,
         col = "orange", lwd = 1, lty = 2)

points(advertise$TV, advertise_TV_model$fitted.values, pch = 16, col = "orange")

legend("topleft",
       legend = expression("Dreapta de regresie", (hat(Y)[i] - Y[i])^2),
       lwd = c(2, 2),
       col = c("royalblue", "orange"),
       lty = c(1, 2),
       bty = "n")

points(advertise$TV, advertise$sales, pch = 16, col = "brown3")
```



Tabelul ANOVA se obține prin

```r
# tabel ANOVA
anova(advertise_TV_model)
Analysis of Variance Table

Response: sales
           Df Sum Sq Mean Sq F value    Pr(>F)
TV          1 3314.6  3314.6  312.14 < 2.2e-16 ***
Residuals 198 2102.5    10.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Definiția *coeficientului de determinare* R^2 este strâns legată de descompunerea ANOVA:

$$R^2 = \frac{SS_{reg}}{SS_T} = \frac{SS_T - RSS}{SS_T} = 1 - \frac{RSS}{SS_T}$$

R^2 măsoară **proporția din variația** variabilei răspuns Y **explicată** de variabila predictor X prin regresie. Proporția din variația totală a lui Y care nu este explicată este $1 - R^2 = \frac{RSS}{SS_T}$. Intuitiv, R^2 măsoară cât de bine modelul de regresie este în concordanță cu datele (cât de strâns este norul de puncte în jurul dreptei de regresie). Observăm că dacă datele concordă *perfect* cu modelul (adică $RSS = 0$) atunci $R^2 = 1$.

Putem vedea că $R^2 = r_{xy}^2$, unde r_{xy} este *coeficientul de corelație* empiric:

$$r_{xy} = \frac{s_{xy}}{s_x s_y} = \frac{\sum_{i=1}^n \left(X_i - \bar{X}\right)\left(Y_i - \bar{Y}\right)}{\sqrt{\sum_{i=1}^n \left(X_i - \bar{X}\right)^2}\sqrt{\sum_{i=1}^n \left(Y_i - \bar{Y}\right)^2}}$$

Mai mult se poate verifica și că $R^2 = r_{y\hat{y}}^2$, adică *coeficientul de determinare este egal cu pătratul coeficientului de corelație empirică dintre* $Y_1, \ldots, Y_n$ *și* $\hat{Y}_1, \ldots, \hat{Y}_n$.

Verificăm relația $R^2 = r_{xy}^2 = r_{y\hat{y}}^2$ numeric:

```r
yHat = advertise_TV_model$fitted.values

advertise_TV_model_summary$r.squared # R^2
[1] 0.6118751
cor(advertise$TV, advertise$sales)^2 # corelatia^2 dintre x si y
[1] 0.6118751
cor(advertise$sales, yHat)^2 # corelatia^2 dintre y si yHat
[1] 0.6118751
```

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

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

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

```r
summary(advertise_TV_model)

Call:
lm(formula = sales ~ TV, data = advertise)

Residuals:
    Min      1Q  Median      3Q     Max
-8.3860 -1.9545 -0.1913  2.0671  7.2124

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.032594   0.457843   15.36   <2e-16 ***
TV          0.047537   0.002691   17.67   <2e-16 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.259 on 198 degrees of freedom
Multiple R-squared:  0.6119,    Adjusted R-squared:  0.6099
F-statistic: 312.1 on 1 and 198 DF,  p-value: < 2.2e-16
```

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.

2.4.1 Predicție

Pentru un nou set de predictori, 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:

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

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

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

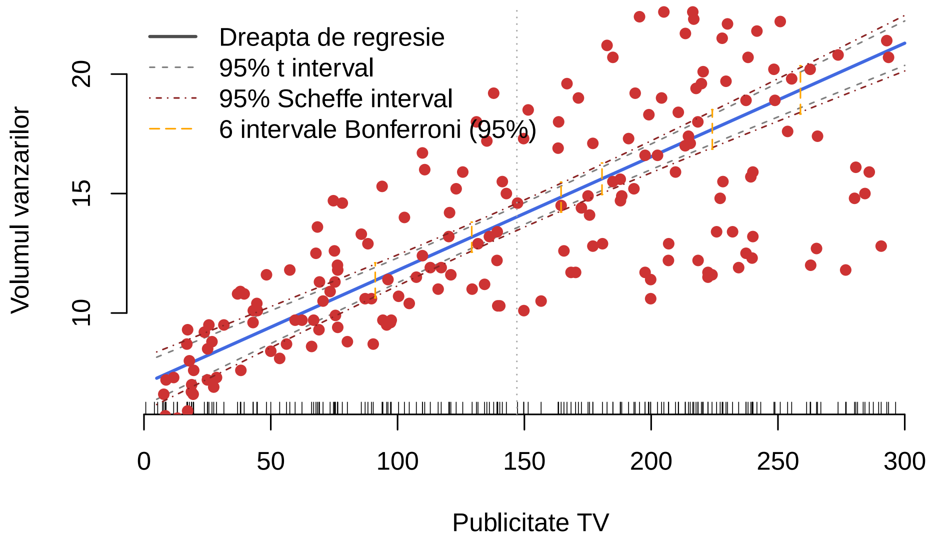
Pentru a găsi aceste intervale vom folosi funcția `predict()`:

```
newData = data.frame(TV = 150)
newData2 = data.frame(TV = c(130, 140, 150))

# Predictie
predict(advertise_TV_model, newdata = newData)
       1
14.16309

# Predictie pentru valoarea raspunsului mediu
predict(advertise_TV_model, newdata = newData, interval = "confidence")
       fit      lwr      upr
1 14.16309 13.70842 14.61776
predict(advertise_TV_model, newdata = newData2, interval = "confidence")
       fit      lwr      upr
1 13.21236 12.74905 13.67566
2 13.68772 13.23179 14.14365
3 14.16309 13.70842 14.61776

# Predictie asupra observatiilor viitoare
predict(advertise_TV_model, newdata = newData, interval = "prediction")
       fit      lwr      upr
1 14.16309 7.720898 20.60528
predict(advertise_TV_model, newdata = newData2, interval = "prediction")
```

```
       fit      lwr      upr
1 13.21236 6.769550 19.65516
2 13.68772 7.245442 20.13000
3 14.16309 7.720898 20.60528
```

Volumul de vânzări prezis impreună cu intervalul de încredere de nivel 95% pentru răspunsul mediu este ilustrat în figura următoare

```
g = seq(5,300,0.5)

p = predict(advertise_TV_model, data.frame(TV = g), se = T, interval = "confidence")
matplot(g, p$fit, type = "l", lty = c(1,2,2),
        lwd = c(2,1,1),
        col = c("royalblue", "grey50", "grey50"),
        xlab = "Publicitate TV",
        ylab = "Volumul vanzarilor",
        bty = "n")
rug(advertise$TV)
points(advertise$TV, advertise$sales, col = "brown3", pch = 16)
abline(v = mean(advertise$TV), lty = 3, col = "grey65")

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

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

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

# Bonferroni bounds

x0 = runif(6, min = 10, max = 290)
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(advertise$TV))^2/s_xx)
up_bonf = b0 + b1*x0 + t_bonf*sigma_hat*sqrt(1/n+(x0-mean(advertise$TV))^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")
```



2.5 Diagnostic

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

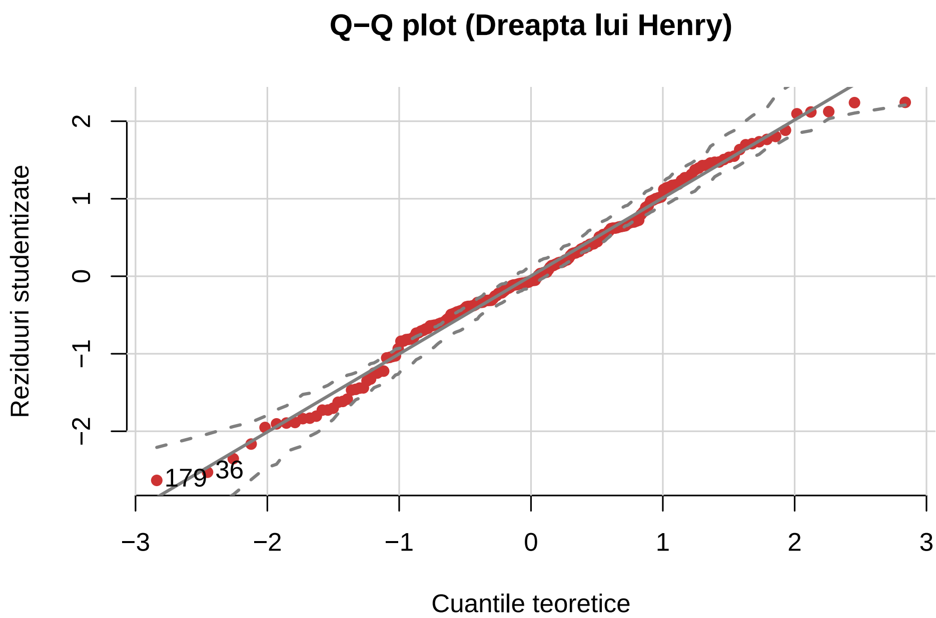
a) *Independența*

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.

b) *Normalitatea*

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

```
# library(car)
par(bty = "n")
qqPlot(advertise_TV_model, col = "brown3", col.lines = "grey50", pch = 16,
       simulate = TRUE,
       xlab = "Cuantile teoretice",
       ylab = "Reziduuri studentizate",
       main = "Q-Q plot (Dreapta lui Henry)",
       bty = "n")
```




```
[1]  36 179
```

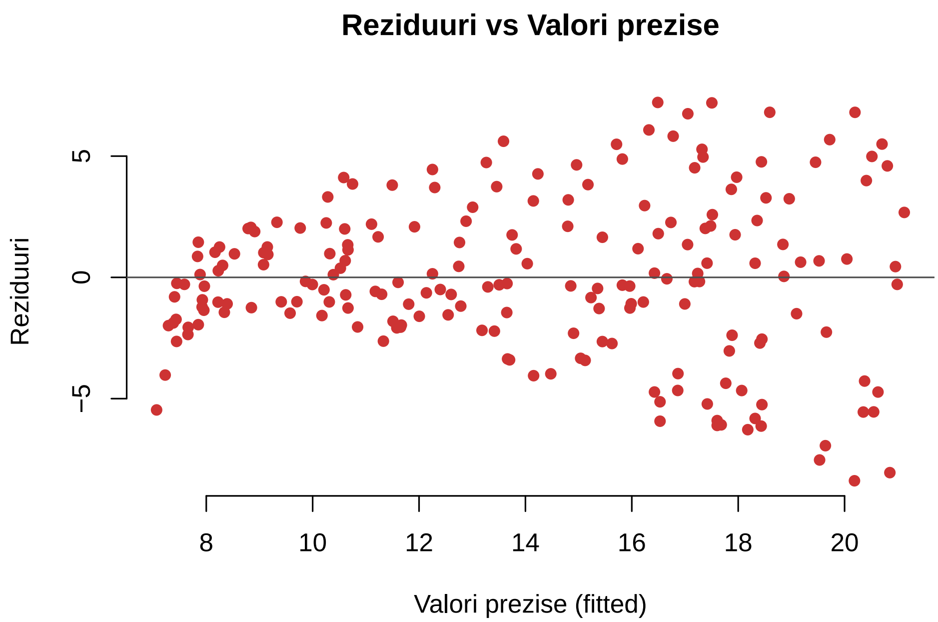
Putem folosi și testul `Shapiro-Wilk`:

```
shapiro.test(residuals(advertise_TV_model))

    Shapiro-Wilk normality test

data:  residuals(advertise_TV_model)
W = 0.99053, p-value = 0.2133
```

c) *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 ($\hat{\varepsilon}$).

```
plot(residuals(advertise_TV_model)~fitted(advertise_TV_model),  col = "brown3", pch = 16,
     xlab = "Valori prezise (fitted)",
     ylab = "Reziduuri",
     main = "Reziduuri vs Valori prezise",
     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 explicativă 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).