Inferență statistică în ML

Cap 7. Modelul regresiei liniare. Reziduuri. Inferența cu regresie.

April 18, 2024

Modelul regresiei: zgomot Gaussian adăugat

- metoda celor mai mici pătrate realizează o estimare
- ne interesează să tragem concluzii privitoare la întreaga populație (inferență)
- considerăm dezvoltarea unui model probabilist pentru regresie:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i, \qquad i = 1 \dots N$$

- presupunem $\epsilon_i \in N(0, \sigma^2)$, pentru care:
  - $E[Y_i|X_i = x_i] = \mu_i = \beta_0 + \beta_1 x_i$
  - $Var[Y_i|X_i = x_i] = \sigma^2$
- ϵ_i sunt considerate a fi erori independente, răspunsul unor variabile care nu au fost incluse în model, și al căror comportament poate fi modelat ca erori gaussiene independente
- $Var[Y_i|X_i = x_i]$ este dispersia în jurul liniei de regresie, nu este dispersia răspunsului Y_i; va fi mai mică decât dispersia răspunsului Y_i, pentru că o parte din variabilitatea lui Y este explicată de regresia liniară

Interpretarea coeficienților regresiei: intercept

- β_0 este valoarea așteptată a răspunsului dacă predictorul (X) este zero:

$$E[Y|X=0] = \beta_0 + \beta_1 * 0 = \beta_0$$

- nu e interesant întotdeauna, de exemplu când $X = 0$ este mult în afara setului de date (X este înălțimea sau tensiunea sangvină)
- intercept-ul poate varia:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i = \underbrace{\beta_0 + a\beta_1}_{\tilde{\beta}_1} + \beta_1(X_i - a) + \epsilon_i$$

$$= \tilde{\beta}_1 + \beta_1(X_i - a) + \epsilon_i$$

- translatarea valorilor X_i cu o valoare a nu modifică panta (slope), doar intercept-ul
- de regulă a se alege ca fiind $\bar{X}$, astfel că în acest caz intercept-ul este interpretat ca fiind valoarea așteptată când predictorul X este egal cu media $\bar{X}$

Interpretarea coeficienților regresiei: slope

- β_1 este modificarea răspunsului cauzată de modificarea cu o unitate a predictorului:

$$E[Y|X = x + 1] - E[Y|X = x] = \beta_0 + \beta_1(x + 1) - \beta_0 - \beta_1 x = \beta_1$$

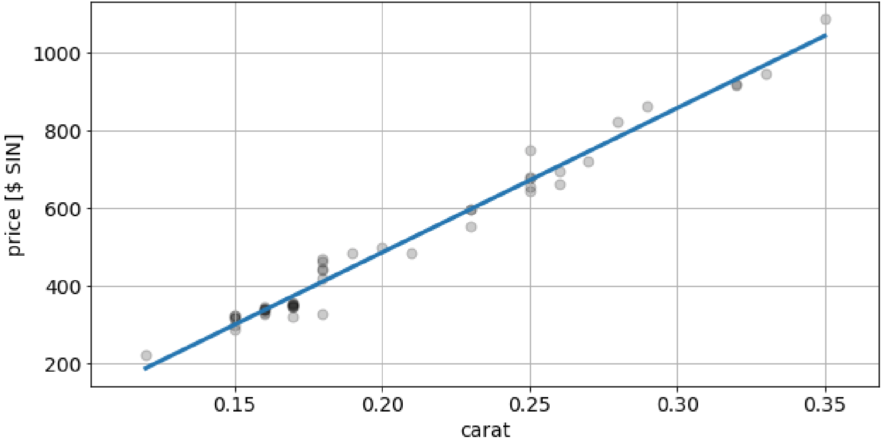
- modificarea pantei înseamnă scalarea lui X cu un factor constant a:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i = \beta_0 + \frac{\beta_1}{a}(X_i a) + \epsilon_i = \beta_0 + \tilde{\beta}_1(X_i a) + \epsilon_i$$

- multiplicarea lui X cu un factor constant determină un coeficient slope obținut prin împărțirea vechiului slope la a

Regresia prețului diamantelor în funcție de masă: diamond dataset

```
intercept: -259.6259071915547 coefficient: 3721.024851550472
```



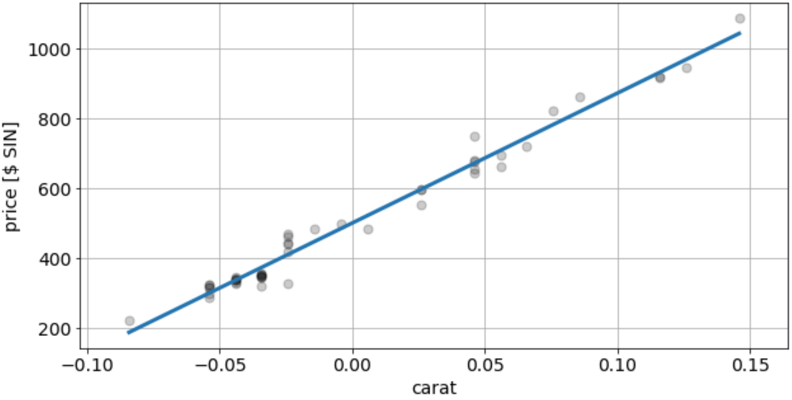
Regresia prețului diamantelor în funcție de masă: diamond dataset (2)

```
x, y = np.array(diamond['carat'].values), np.array(diamond['price'].values)

xext = sm.add_constant(x)

lm = sm.OLS(y, xext).fit()
beta0, beta1 = lm.params[0], lm.params[1]
print('intercept:', beta0, 'coefficient:', beta1)

x1 = np.linspace(np.min(x), np.max(x), 100)
y1 = beta0 + beta1 * x1

fig, ax = plt.subplots(1, 1, figsize=(10, 5))
ax.scatter(x, y, c='k', alpha = .2, s=50)
ax.plot(x1, y1, lw=3)
ax.set(xlabel="carat", ylabel="price [$ SIN]")
ax.grid(True)
plt.show()
```

Regresia prețului diamantelor în funcție de masă: diamond dataset (3)

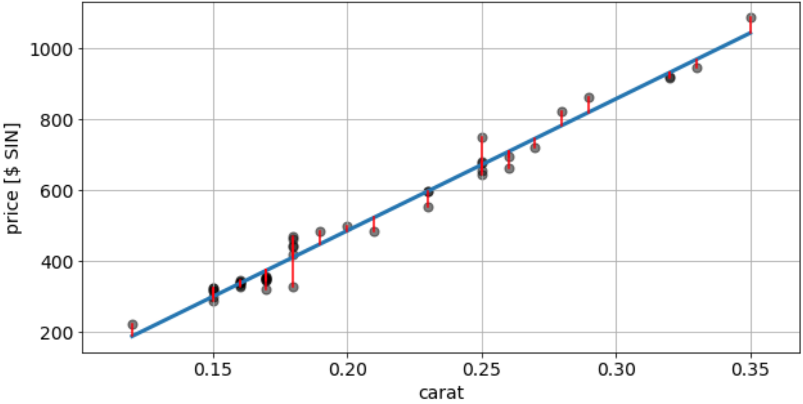
```
x -= np.mean(x)
```

```
mean(X): 0.20416666666666667
intercept: 500.08333333333334 coefficient: 3721.024851550472
```



Residuals

- dacă am observa doar prețurile, fără masa asociată, variabilitatea variabilei preț ar fi foarte mare
- dacă observăm prețul în contextul masei, variabilitatea este redusă și se manifestă în jurul dreptei de regresie



Residuals (2)

- modelul este $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$, unde $\epsilon_i \sim N(0, \sigma^2)$
- valoarea observată este Y_i pentru valoarea predictorului X_i
- predicția calculată este $\hat{Y}_i$ pentru valoarea predictorului X_i:

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$$

- reziduul ϵ_i este diferența dintre valoarea observată și valoarea prezisă de regresie:

$$e_i = Y_i - \hat{Y}_i$$

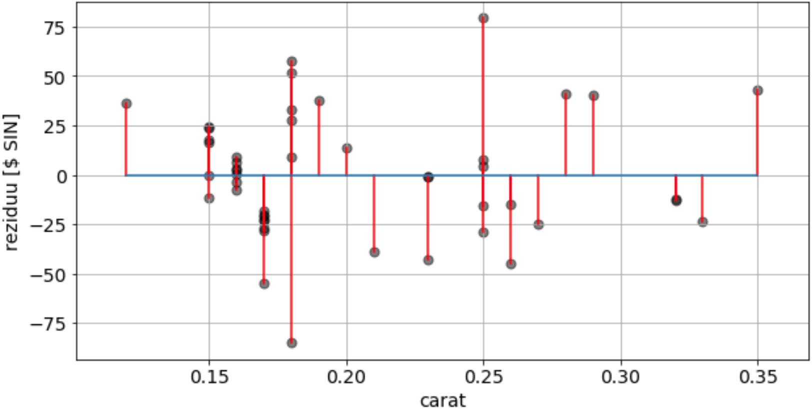
- este distanța pe verticală dintre punctul observat și dreapta de regresie
- metoda celor mai mici pătrate (LMS) minimizează $\sum_{i=1}^{N} e_i^2$ (v. desen)
- e_i poate fi privit ca un estimator al lui ϵ_i

Proprietățile reziduurilor

- $E[e_i] = 0$
- $\sum_{i=1}^{N} e_i = 0$
- $\sum_{i=1}^{N} e_i X_i = 0$

- reziduurile sunt utile pentru a investiga potriviri slabe ale modelului cu regresia calculată (liniile roșii devin mai lungi)
- reziduurile pozitive sunt situate deasupra liniei de regresie, cele negative - dedesubt
- reziduul poate fi interpretat ca ieșirea Y din care se scade asocierea liniară a predictorului X
- variația reziduală (din care s-a scăzut variabilitatea datorată regresorului X) nu trebuie confundată cu variația explicată de model, $\hat{Y}$

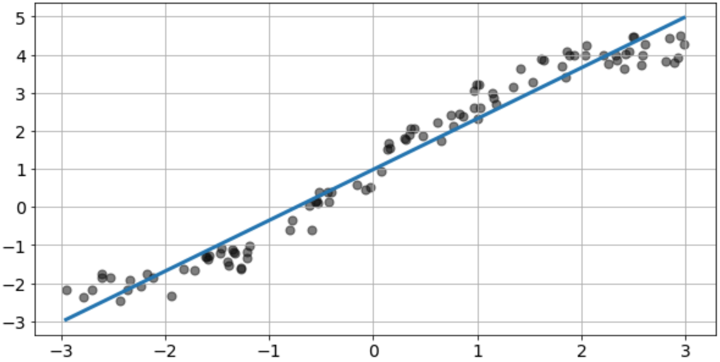
Variația reziduală



- variația reziduală nu trebuie să urmeze vreun pattern
- suma reziduurilor este zero, ele vor fi situate oarecum echilibrat, și în partea superioară și în cea inferioară, 'aranjate' aleator

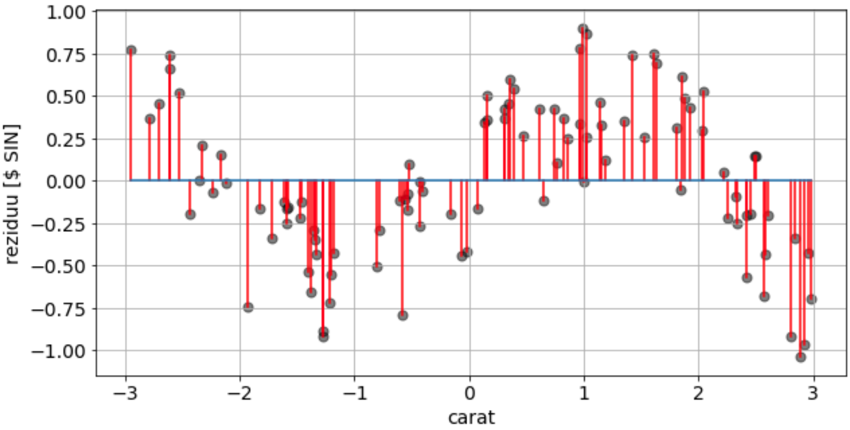
Variația reziduală (1)

```
x = np.random.rand(100) * 6 - 3
y = x + np.sin(x) + np.random.rand(100) + np.sqrt(0.2)
```



- există o componentă liniară care explică mult din variație
- modelul nu este perfect; variația sinusoidală rămâne neexplicată

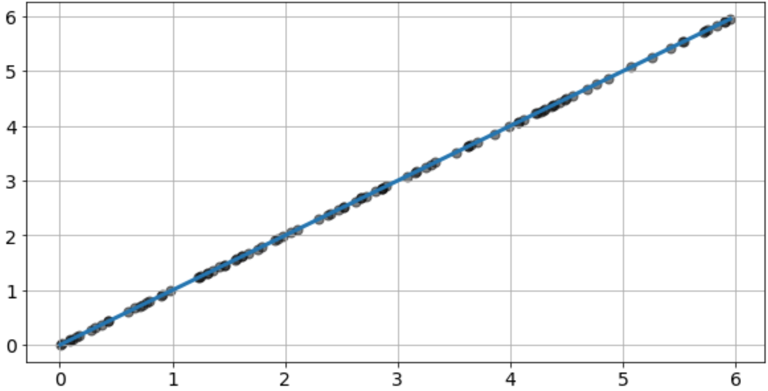
Variația reziduală (2)



- variația sinusoidală apare în reziduu (nu e random ⇒ modelul nu explică toată variația)

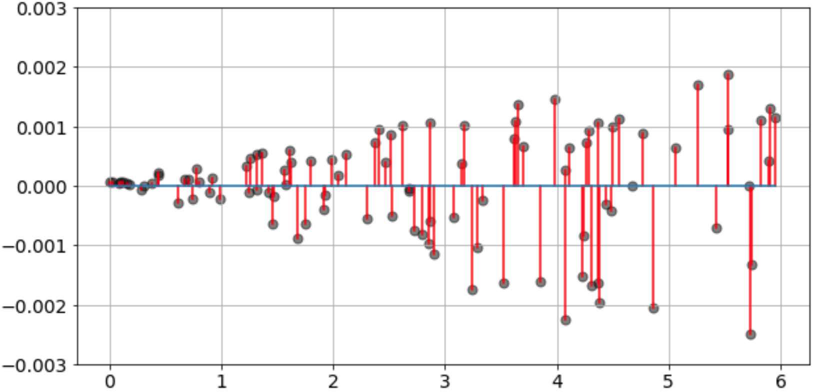
Variația reziduală (3)

```
x = np.random.rand(100) * 6
y = x + (np.random.rand(100) - 0.5) * .001 * x
```



- scatter plot: pare că modelul de regresie explică integral variabilitatea

Variația reziduală (4)



- variabilitatea reziduului devine evidentă; de fapt e vorba de dispersie diferită[1]

[1] https://en.wikipedia.org/wiki/Heteroscedasticity

Variabilitatea reziduală

- modelul este $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$, unde $\epsilon_i \sim N(0, \sigma^2)$
- estimarea pentru σ^2 este $\frac{1}{n}\sum_{i=1}^{n} e_i^2$, reziduul pătrat mediu
- de fapt se folosește:

$$\hat{\sigma^2} = \frac{1}{n-2}\sum_{i=1}^{n} e_i^2$$

- sunt $n-2$ grade de libertate deoarece sunt două constrângeri:
  - suma reziduurilor este zero
  - reziduul este calculat folosind dreapta de regresie
- impactul e semnificativ pentru valori mici, $n < 20$, pentru valori mari ale lui n, nu contează

Variabilitatea explicată de regresie și reziduul

- variablilitatea totală a datelor este dată de dispersie, $\sum_{i=1}^{n}(Y_i - \bar{Y})^2$
- variabilitatea explicată de regresie este diferența dintre valoarea estimată și medie, $\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2$
- variabilitatea reziduală este ceea ce a rămas neexplicat de regresie, adică $\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$

- se poate arăta că:

$$\sum_{i=1}^{n}(Y_i - \bar{Y})^2 = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 + \sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2$$

- variabilitatea totală = variabilitatea reziduală + variabilitatea regresiei

R squared

$$\sum_{i=1}^{n}(Y_i - \bar{Y})^2 = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 + \sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2$$

- R squared este procentul de variabilitate totală care este explicat de regresie:

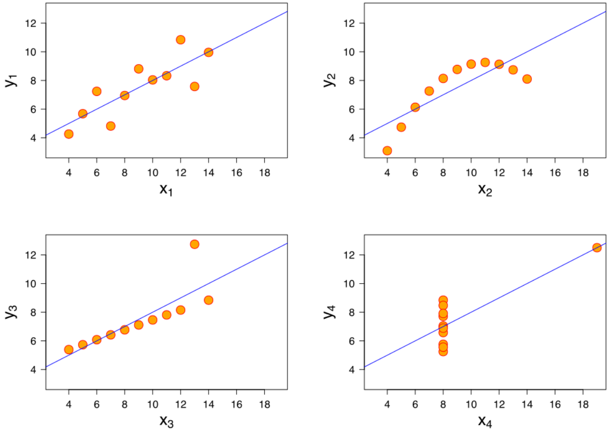
$$R^2 = \frac{\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$

Proprietăți ale R^2

- R^2 este procentul de variație explicat de model
- $0 \leq R^2 \leq 1$
- R^2 este înșelător:
    - ștergerea de puncte (date) poate crește R^2
    - adăugarea de termeni la regresie (polinomială, crește gradul) întotdeauna crește R^2

Cvartetul lui Anscombe: importanța vizualizării datelor

- `https://en.wikipedia.org/wiki/Anscombe%27s_quartet`
- dataseturi cu statistici descriptive aproape identice (medii x și y, dispersii x și y, corelație x vs. y, R^2)



Recapitularea modelului

- modelul regresiei:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

- $\epsilon_i \sim N(0, \sigma^2)$
- intercept: $\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$
- coeficientul: $\hat{\beta}_1 = Cor(Y, X) \frac{Sd(Y)}{Sd(X)}$
- ne interesează cum realizăm un test statistic bazat pe cele două ipoteze, H_0 și H_a

Testarea ipotezei

- statistica de tipul $\frac{\hat{\theta}-\theta}{\hat{\sigma}_i}$ are de regulă următoarele proprietăți:
  - este distribuită normal și are o distribuție de tip Student T, dacă dispersia estimată este înlocuită cu sample estimate;
  - poate fi folosită pentru testarea $H_0 : \theta = \theta_0$ vs. $H_a : \theta >, <, \neq \theta_0$;
  - poate fi folosită pentru a crea un interval de confidență pentru θ de forma $\hat{\theta} \pm Q_{1-\alpha/2}\hat{\sigma}_{\hat{\theta}}$, unde $Q_{1-\alpha/2}$ este quantila relevantă dintr-o distribuție normală sau una Student T, iar $\hat{\sigma}_{\hat{\theta}}$ este eroarea standard asociată eșantionului.

Dispersia coeficienților regresiei (1)

- vom considera dispersia reziduurilor ca fiind σ^2, mai precis: $\sigma^2 = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2/(n-2)$
- următoarele rezultate le dăm fără demonstrație[2]:

$$\sigma^2_{\hat{\beta}_1} = Var(\hat{\beta}_1) = \sigma^2 / \sum_{i=1}^{n}(X_i - \bar{X})^2$$

- cu cât variabilitatea față de dreapta de regresie este mai mică, cu atât dispersia β_1 este mai mică
- însă cu cât punctele sunt mai 'adunate' spre media $\bar{X}$, cu atât panta dreptei este mai imprecisă

$$\sigma^2_{\hat{\beta}_0} = Var(\hat{\beta}_0) = \left(\frac{1}{n} + \frac{\bar{X}^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}\right)\sigma^2$$

[2] vezi `https://newonlinecourses.science.psu.edu/stat414/node/280/`

Dispersia coeficienților regresiei (2)

- în practică, σ este înlocuit de estimandul său, $\hat{\sigma}$
- în cazul unor erori Gaussiene iid, statistica:

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma}_{\hat{\beta}_j}}$$

  urmărește o distribuție Student T cu $n - 2$ grade de libertate, și o distribuție normală pentru n mare
- aceasta poate fi folosită pentru a crea intervale de confidență și de a realiza testarea ipotezei

Construcția statisticilor

```
x, y = np.array(diamond['carat'].values), np.array(diamond['price'].values)

xext = sm.add_constant(x)

lm = sm.OLS(y, xext).fit()
beta0, beta1 = lm.params[0], lm.params[1]
print('intercept:', beta0, 'coefficient:', beta1)

n = len(lm.resid)
sigma = np.sqrt(np.sum(lm.resid**2)/(n - 2))
print('sigma:', sigma)

sx = np.sum((x - np.mean(x))**2)
se_beta0 = np.sqrt(1/n + np.mean(x)**2 / sx) * sigma
se_beta1 = sigma / np.sqrt(sx)

stat_beta0, stat_beta1 = beta0 / se_beta0, beta1 / se_beta1
p_beta0 = 2 * stats.t.sf(np.abs(stat_beta0), df=n-2)
p_beta1 = 2 * stats.t.sf(np.abs(stat_beta1), df=n-2)
i_beta0 = beta0 + np.array([-1, 1]) * stats.t.ppf(0.975, df=n-2) * se_beta0
i_beta1 = beta1 + np.array([-1, 1]) * stats.t.ppf(0.975, df=n-2) * se_beta1
```

```
intercept: -259.6259071915547 coefficient: 3721.024851550472
sigma: 31.84052226503175
```

Construcția statisticilor: dual

```
df1 = pd.DataFrame([['beta0', beta0, se_beta0, stat_beta0, p_beta0, i_beta0[0], i_beta0[1]],
                    ['beta1', beta1, se_beta1, stat_beta1, p_beta1, i_beta1[0], i_beta1[1]]],
                columns=['Parameter', 'Estimate', 'Std. Error', 't Value', 'P(>|t|)', '[0.025', '0.975]'])
df2 = pd.DataFrame([['beta0', lm.params[0], lm.bse[0], lm.tvalues[0], lm.pvalues[0], lm.conf_int()[0][0],
                     lm.conf_int()[0][1]],
                    ['beta1', lm.params[1], lm.bse[1], lm.tvalues[1], lm.pvalues[1], lm.conf_int()[1][0],
                     lm.conf_int()[1][1]]],
                columns=['Parameter', 'Estimate', 'Std. Error', 't Value', 'P(>|t|)', '[0.025', '0.975]'])
print(df1)
print(df2)
```

```
  Parameter     Estimate  Std. Error    t Value       P(>|t|)       [0.025  \
0     beta0  -259.625907   17.318856 -14.990938  2.523271e-19  -294.486957
1     beta1  3721.024852   81.785880  45.497155  6.751260e-40  3556.398413

        0.975]
0  -224.764858
1  3885.651290
  Parameter     Estimate  Std. Error    t Value       P(>|t|)       [0.025  \
0     beta0  -259.625907   17.318856 -14.990938  2.523271e-19  -294.486957
1     beta1  3721.024852   81.785880  45.497155  6.751260e-40  3556.398413

        0.975]
0  -224.764858
1  3885.651290
```

Construcția statisticilor: sumar

```
print(lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.978
Model:                            OLS   Adj. R-squared:                  0.978
Method:                 Least Squares   F-statistic:                     2070.
Date:                Fri, 03 May 2019   Prob (F-statistic):           6.75e-40
Time:                        20:16:37   Log-Likelihood:                -233.20
No. Observations:                  48   AIC:                             470.4
Df Residuals:                      46   BIC:                             474.1
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       -259.6259     17.319    -14.991      0.000    -294.487    -224.765
x1          3721.0249     81.786     45.497      0.000    3556.398    3885.651
==============================================================================
Omnibus:                        0.739   Durbin-Watson:                   1.994
Prob(Omnibus):                  0.691   Jarque-Bera (JB):                0.181
Skew:                           0.056   Prob(JB):                        0.913
Kurtosis:                       3.280   Cond. No.                         18.5
==============================================================================
```

- interpretare conf. int. β_1: 95% încredere că o creștere cu 1 a masei (carat), duce la o creștere de preț între 3556 și 3886

Regresia: intervale de confidență

- considerăm predicția lui Y pentru o valoare a lui X
- de exemplu, predicția prețului diamantului în funcție de masa sa, sau predicția înălțimii copilului dată fiind înălțimea părintelui
- predicția evidentă pentru punctul x_0 este $\hat{\beta}_0 + \hat{\beta}_1 x_0$
- avem nevoie de standard error pentru a crea un interval de confidență
- facem distincție între:
    a. intervalul de confidență pentru dreapta de regresie în punctul x_0 și
    b. care ar fi valoarea prezisă pentru y în punctul x_0

    a. dreapta de regresie în x_0: stderr $= \hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x_0 - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}}$

    b. intervalul de predicție în x_0: stderr $= \hat{\sigma}\sqrt{1 + \frac{1}{n} + \frac{(x_0 - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}}$

Regresia: intervale de confidență (2)

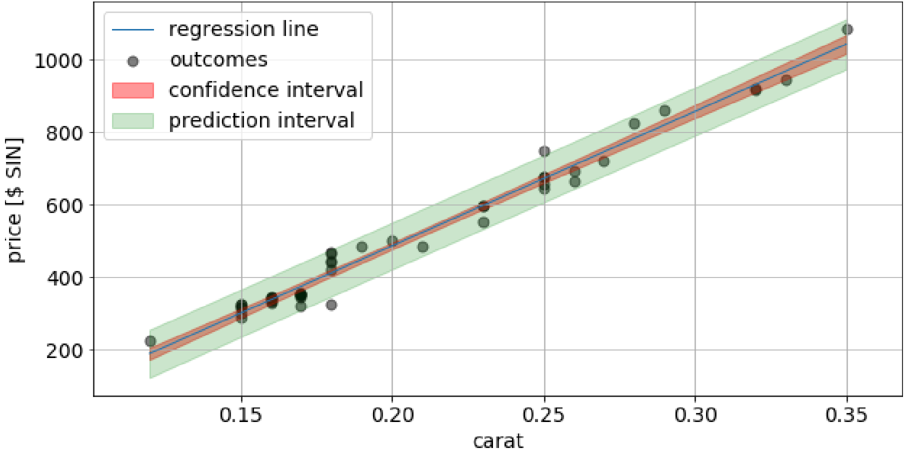
$$\text{stderr} = \hat{\sigma}\sqrt{1 + \frac{1}{n} + \frac{(x_0 - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}}$$

- termenul $\hat{\sigma}$: cu cât reziduurile au o dispersie mai mare, cu atât mai larg e intervalul de confidență
- cu creșterea lui n, standard error și deci intervalul de confidență, se restrâng
- termenul '1' arată că intervalul pentru predicție e mai larg ca intervalul pentru dreapta de regresie
- cu cât suntem mai aproape de media $\bar{X}$, cu atât mai bine, eroarea standard e mai mică
- cu cât variabilitatea lui X este mai mare, cu atât mai bine, intervalul de confidență e mai restrâns

Regresia: intervale de confidență (3)

```
def f(x):
    return beta0 + beta1 * x

x1 = np.linspace(np.min(x), np.max(x), 100)
y1 = f(x1)

# t quantile
t = stats.t.ppf(0.975, df=n-2)

# dispersia reziduurilor
sigma = np.sqrt(np.sum(lm.resid**2)/(n-2))

# confidence interval pentru dreapta
ci = t * sigma * np.sqrt(1/n + (x1-np.mean(x))**2 / np.sum((x-np.mean(x))**2))

# confidence interval pentru predictie (prediction interval)
pi = t * sigma * np.sqrt(1 + 1/n + (x1-np.mean(x))**2 / np.sum((x-np.mean(x))**2))

fig, ax = plt.subplots(1, 1, figsize=(10, 5))
ax.scatter(x, y, c='k', alpha = .5, s=50)
ax.plot(x1, y1, lw=1)
ax.fill_between(x1, y1-ci, y1+ci, color='red', alpha=0.4)
ax.fill_between(x1, y1-pi, y1+pi, color='green', alpha=0.2)
ax.set(xlabel="carat", ylabel="price [$ SIN]")
ax.grid(True)
ax.legend(['regression line', 'outcomes', 'confidence interval', 'prediction interval'])
plt.show()
```

Confidence Interval și Prediction Interval



Modelul celor mai mici pătrate (least squares)

- pentru un set de date, dorim să calculăm coeficienții β_0 și β_1 astfel ca valoarea calculată de funcția liniară $\hat{y}_i = \beta_0 + \beta_1 \cdot x_i$ să fie cât mai apropiată de valoarea dată y_i:

$$y_i \sim \beta_0 + \beta_1 x_i \qquad \text{pentru} \quad i = 1 \dots M$$

- asta înseamnă pentru toate datele:

$$\begin{array}{ccc} y_1 & \sim & \beta_0 + \beta_1 x_1 \\ y_2 & \sim & \beta_0 + \beta_1 x_2 \\ \dots & & \dots \\ y_M & \sim & \beta_0 + \beta_1 x_M \end{array}$$

- în mod evident acesta nu este un sistem de ecuații

Modelul least squares (2)

- relațiile:

$$\begin{array}{ccc} y_1 & \sim & \beta_0 + \beta_1 x_1 \\ y_2 & \sim & \beta_0 + \beta_1 x_2 \\ \dots & & \dots \\ y_M & \sim & \beta_0 + \beta_1 x_M \end{array}$$

- se pot scrie matricial:

$$Y \sim \underbrace{\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \dots & \dots \\ 1 & x_M \end{bmatrix}}_{X} \cdot \underbrace{\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}}_{\beta}$$

- expresia de minimizat devine:

$$\min_{\beta} J(\beta) = \min_{\beta} \|X\beta - Y\|^2$$

Modelul least squares (3)

- vom scrie expresia gradientului (vectorial):

$$
\begin{aligned}
0 &= \nabla_\beta J(\beta) = 2(X\beta - Y)^T X \\
0 &= (X\beta)^T X - Y^T X \\
(X\beta)^T X &= Y^T X \\
\beta^T X^T X &= Y^T X \\
(\beta^T X^T X)^T &= (Y^T X)^T \\
X^T X \beta &= X^T Y \\
(X^T X)^{-1}(X^T X)\beta &= (X^T X)^{-1} X^T Y \\
\beta &= (X^T X)^{-1} X^T Y
\end{aligned}
$$

- matricea $(X^T X)^{-1}$ se numește pseudoinversa Moore-Penrose[3]

[3] https://www.math.ucla.edu/~laub/33a.2.12s/mppseudoinverse.pdf

Analiza regresiei de mai multe variabile (predictors)

- când vrem să prezicem Y pe baza unui X_1, poate exista suspiciunea că de fapt, există un alt predictor X_2 care îl influențează de fapt pe X_1, deci X_2 este 'adevăratul predictor' pentru Y
- exemplu: legătura dintre bomboanele cu mentă și funcția pulmonară (măsurată ca și capacitate)
  - un posibil argument ar putea fi cum că fumătorii folosesc mai mult bomboanele mentolate, iar fumatul este legat de scăderea capacității pulmonare
  - pentru a putea convinge, ar trebui să vedem dacă fumătorii ce folosesc bomboane mentolate au capacitate pulmonară redusă, precum și ne-fumătorii ce folosesc bomboane mentolate au de asemenea capacitate pulmonară redusă - în acest context am putea verifica ipoteza că doar bomboanele în sine au efect asupra capacității
  - ar trebui să găsim aceeași corelație indiferent de status-ul fumător/nefumător
  - prin regresie, menținem constant acest status și investigăm cealaltă variabilă

Regresia multi-variabilă: exemplu

- multivariable regression este un instrument puternic pentru predicție
- presupunem o companie de asigurări vrea să prezică, pe baza cererilor de despăgubire din anii precedenți, câte zile de spitalizare va necesita o persoană pentru anul curent
- există mulți predictori (features) conținute în aceste cereri; regresia liniară simplă nu e potrivită
- cum se încorporează mai mulți predictori în regresie?
- care e consecința adăugării mai multor variabile de regresie? overfitting?
- dar dacă omitem predictori?

Modelul liniar

- general linear model (GLM) extinde o regresie simplă prin adăugarea de termeni:

$$Y_i = \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_p X_{pi} + \epsilon_i = \sum_{k=1}^{p} \beta_k X_{ki} + \epsilon_i$$

- în mod tipic, $X_{1i} = 1$, astfel încât β_1 este intercept-ul[4]
- metoda celor mai mici pătrate (adică estimarea maximum likelihood[5] sub ipoteza Gaussiană iid a erorilor) minimizează:

$$\sum_{i=1}^{n} \left(Y_i - \sum_{k=1}^{p} \beta_k X_{ki} \right)^2$$

[4] v. add_constant în modelul OLS din statmodels

[5] maximum likelihood estimator se aplică la căutarea parametrilor care se potrivesc cel mai bine datelor

Modelul liniar

- de observat că modelul liniar presupune liniaritate în coeficienți; astfel:

$$Y_i = \beta_1 X_{1i}^2 + \beta_2 X_{2i}^2 + \ldots + \beta_p X_{pi}^2 + \epsilon_i$$

este tot un model liniar, deși am ridicat la pătrat predictorii

Coeficienții regresiei multi-variabilă

- considerăm modelul regresiei prin origine - Y_i și X_i sunt shiftate în origine (medie zero)
- din cursul anterior:

$$\beta_1 = Cor(Y,X)\frac{Sd(Y)}{Sd(X)} = \frac{Cov(Y,X)}{Sd(X)Sd(Y)}\frac{Sd(Y)}{Sd(X)} = \frac{Cov(Y,X)}{Var(X)}$$
$$= \frac{\sum_i (Y_i - \bar{Y})(X_i - \bar{X})}{\sum_i (X_i - \bar{X})^2}$$
$$= \frac{\sum_i Y_i X_i}{\sum_i X_i^2}$$

- considerăm două variabile regressor, $Y_i = \beta_1 X_{1i} + \beta_2 X_{2i} + \epsilon_i$, pentru care minimizăm:

$$\sum_{i=1}^{n} (Y_i - \beta_1 X_{1i} - \beta_2 X_{2i})^2$$

Coeficienții regresiei multi-variabilă (2)

- facem notația:

$$\sum_{i=1}^{n}(\underbrace{Y_i - \beta_1 X_{1i}}_{\tilde{Y}_i} - \beta_2 X_{2i})^2$$

- atunci:

$$\beta_2 = \frac{\sum_i \tilde{Y}_i X_{2i}}{\sum_i X_{2i}^2}$$

- introducem în prima relație:

$$\sum_{i=1}^{n}\left[Y_i - \beta_1 X_{1i} - \frac{\sum_j (Y_j - \beta_1 X_{1j}) X_{2j}}{\sum_j X_{2j}^2} X_{2i}\right]^2$$

Coeficienții regresiei multi-variabilă (3)

- după mai multe prelucrări:

$$\sum_{i=1}^{n}\left[Y_i - \underbrace{\frac{\sum_j Y_j X_{2j}}{\sum_j X_{2j}^2}}_{b} X_{2i} - \beta_1\left(X_{1i} - \underbrace{\frac{\sum_j X_{1j} X_{2j}}{\sum_j X_{2j}^2}}_{a} X_{2i}\right)\right]^2$$

$$= \sum_{i=1}^{n}\left[Y_i - bX_{2i} - \beta_1\left(X_{1i} - aX_{2i}\right)\right]^2$$

- coeficientul b are dimensiunea unui β dacă se face regresia lui Y_i funcție de X_{2i}
- idem pentru a, pentru regresia lui X_{1i} funcție de X_{2i}

Coeficienții regresiei multi-variabilă (4)

- după mai multe prelucrări:

$$\beta_1 = \frac{\sum_{i=1}^{n}(Y_i - bX_{2i})(X_{1i} - aX_{2i})}{\sum_{i=1}^{n}(X_{1i} - aX_{2i})^2}$$

- coeficientul β_1 este calculat ca și cum am scoate contribuția (reziduul) lui X_2 din X_1 respectiv reziduul lui X_2 din Y și apoi am face regresia prin origine
- regresia multi-variabilă calculează coeficientul β_1 ca pentru efectul lui X_2 (celălalt), eliminat atât din răspunsul Y cât și din predictorul X_1
- pentru β_2 se obține o formulă similară
- β_2 este coeficientul obținut dacă eliminăm X_1 atât din răspunsul Y cât și predictorul X_1
- **coeficientul regresiei multi-variabilă se calculează prin eliminarea efectului celorlalți predictori atât din răspuns cât și din predictorul vizat**

Cazul general

- metoda celor mai mici pătrate va trebui să minimizeze:

$$\sum_{i=1}^{n}(Y_i - \beta_1 X_{1i} - \beta_2 X_{2i} - \ldots - \beta_p X_{pi})$$

- pentru doar doi regresori (intercept și slope), vom minimiza suma pătratelor distanțelor față de o linie
- pentru trei regresori, minimizăm suma pătratelor distanțelor dintre puncte și plan
- pentru mai mult de trei regresori (4 și mai mulți), avem de-a face cu un hiperplan

Cazul general (2)

- în calculul lui β_1, contribuția celorlalți regresori, $X_2, X_3, \ldots X_p$ a fost înlăturată liniar atât din Y cât și din X_1:

$$\beta_1 = \frac{\sum_{i=1}^{n}(Y_i - \sum_{k=2}^{p} b_k X_{ki})(X_{1i} - \sum_{k=2}^{p} a_k X_{ki})}{\sum_{i=1}^{n}(X_{1i} - \sum_{k=2}^{p} a_k X_{ki})^2}$$

  unde a_k și b_k sunt generalizările lui a și b anteriori, nu pentru 2, ci pentru k

- regresia liniară 'ajustează' coeficientul pentru impactul liniar al celorlalte variabile
- atenție, fiecare din acești termeni de la numitor și de la numărător au dimensiunile unor reziduuri (ce rămâne după ce dependența liniară a fost înlăturată); ca și cum am face regresia fără această componentă

Interpretarea coeficienților

- considerăm media prezisă pentru un set de valori ai regresorilor:

$$E[Y|X_1 = x_1, \ldots X_p = x_p] = \sum_{k=1}^{p} \beta_k x_k$$

- dacă incrementăm doar X_1 cu 1 și restul rămân la fel:

$$E[Y|X_1 = x_1 + 1, \ldots X_p = x_p] = \beta_1(x_1 + 1) + \sum_{k=2}^{p} \beta_k x_k$$

- scădem cele două ecuații:

$$E[Y|X_1 = x_1 + 1, \ldots X_p = x_p] - E[Y|X_1 = x_1, \ldots X_p = x_p]$$
$$= \beta_1(x_1 + 1) + \sum_{k=2}^{p} \beta_k x_k - \sum_{k=1}^{p} \beta_k x_k = \beta_1$$

- coeficientul unui regresor reprezintă schimbarea așteptată în răspunsul Y pe unitatea de regresor X dacă ceilalți regresori nu se modifică

Reziduuri și variația lor

- toate raționamentele pentru Simple Linear Regression se extind pentru regresia liniară multi-variabilă
- modelul este $Y_i = \sum_{k=1}^{p} \beta_k X_{ik} + \epsilon_i$, unde $\epsilon_i \sim N(0, \sigma^2)$
- răspunsul estimat $\hat{Y}_i = \sum_{k=1}^{p} \hat{\beta}_k X_{ik}$
- reziduurile $e_i = Y_i - \hat{Y}_i$
- estimatorul dispersiei $\hat{\sigma}^2 = \frac{1}{n-p} \sum_{i-1}^{k} e_i^2$

Statistici și intervale de confidență

- coeficienții au erori standard, anume $\hat{\sigma}_{\hat{\beta}_k}$, iar

$$\frac{\hat{\beta}_k - \beta_k}{\hat{\sigma}_{\hat{\beta}_k}}$$

urmează o distribuție Student T ci $n - p$ grade de libertate

- răspunsurile prezise (estimările) au erori standard, putem calcula intervalele de confidență pentru predicție și pentru dreapta de regresie

Note

- regresia liniară (modelul liniar) este cel mai aplicat model de ML (conduce detașat)
- modelul liniar este primul model de încercat pentru un set nou de date, deoarece oferă relații ușor de explicat între predictori și răspuns
- exemplu celebru: seriile de timp precum sunetul sunt descompuse în armonici - Transformata Fourier Discretă este un model liniar
- putem aproxima satisfăcător funcții complicate
- putem folosi variabile de tip categorie ca predictori (ANOVA, ANCOVA)