Esercizi Unità 5

Analisi dei dati 2024/25

Cristiano Varin

1. Una società vuole valutare se il compenso dei suoi manager sia legato al budget che gestiscono. A tal fine, la società raccoglie informazioni su un campione di 20 manager. Con i dati campionari si trovano le seguenti quantità: la media del compenso (in migliaia di dollari) è 76.98 con deviazione standard 22.403, la media del budget (in centinaia di migliaia di dollari) è 5.825 con deviazione standard 2.462. La covarianza fra compenso e budget è 46.859.

   (a) Si calcoli la retta di regressione che lega compenso e budget gestito con il metodo dei minimi quadrati. Si commenti il risultato ottenuto.

   (b) Si calcoli l'indice di determinazione e si interpreti il risultato ottenuto.

   Soluzione

   (a) Indichiamo con Y il compenso e con x il budget. Dobbiamo stimare i coefficienti del modello di regressione lineare semplice $Y = \beta_0 + \beta_1 x + \epsilon$. Con i dati a disposizione otteniamo

   $$\hat{\beta}_1 = \frac{46.859}{2.462^2} = 7.731 \quad \text{e} \quad \hat{\beta}_0 = 76.98 - 7.731(5.825) = 31.947.$$

   Quindi l'equazione di regressione stimata è $Y = 31.947 + 7.731x$. I coefficienti stimati indicano che il compenso base è 31 947 dollari e aumenta di 7 731 dollari per ogni 100 000 dollari di budget gestito.

   (b) L'indice di correlazione è pari a

   $$r = \frac{46.859}{(22.403)(2.462)} = 0.85,$$

   quindi l'indice di determinazione vale $R^2 = 0.85^2 = 0.723$ e indica che il modello di regressione descrive approssimativamente il 72% della variabilità del compenso.

2. Il file **manager.csv** contiene tutte le 20 osservazioni delle variabili salario e budget dell'esercizio precedente. Si risponda ai quesiti dell'esercizio precedente usando la funzione **lm** di **R**. Inoltre, si calcoli un intervallo di previsione di livello 99% per il compenso di un manager che gestisce un budget di 1 milione di dollari usando la funzione **predict** di **R**.

   Soluzione Leggiamo i dati:

```
R> manager <- read.csv("manager.csv")
```

   Stimiamo il modello di regressione con la funzione **lm**:

```
R> mod <- lm(salario ~ budget, data = manager)
R> summary(mod)

##
## Call:
## lm(formula = salario ~ budget, data = manager)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.333  -6.684  -2.034   6.984  20.869
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   31.937      7.125   4.482 0.000288 ***
## budget         7.733      1.131   6.837 2.13e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.14 on 18 degrees of freedom
## Multiple R-squared:  0.722,Adjusted R-squared:  0.7065
## F-statistic: 46.74 on 1 and 18 DF,  p-value: 2.126e-06
```

Si noti che le quantità stimate da **lm** sono leggermente diverse da quelle calcolate nell'esercizio precedente per via degli arrotondamenti che erano stati usati nel calcolo delle statistiche.

Per ottenere l'intervallo di previsione di livello 99% con la funzione **predict** bisogna digitare:

```
R> predict <- predict(mod, newdata = data.frame(budget = 10), level = .99,
+         interval = "prediction")
R> predict

##        fit      lwr      upr
## 1 109.2642 70.97487 147.5536
```

L'intervallo di confidenza di livello 99% indica che lo stipendio atteso è approssimativamente di 109 mila dollari con un margine d'errore di poco superiore a 38 mila dollari.

3. Di seguito viene riportata la sintesi di una retta di regressione che descrive l'andamento dell'importo annuale investito da una software house per produrre un certo software negli ultimi 12 anni. L'importo annuale è espresso in migliaia di dollari. Sulla base della retta di regressione stimata, possiamo affermare che l'importo investito è aumentato mediamente più di 2 000 dollari ogni anno?

```
##
## Call:
## lm(formula = importo ~ anno)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2489 -2.2249  0.0663  1.5821  4.9112
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4895.0525   483.8447  -10.12 1.43e-06 ***
## anno            2.4543     0.2409   10.19 1.34e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.881 on 10 degrees of freedom
## Multiple R-squared:  0.9121,Adjusted R-squared:  0.9033
## F-statistic: 103.8 on 1 and 10 DF,  p-value: 1.34e-06
```

Soluzione

L'esercizio chiede di valutare il sistema d'ipotesi

$$\mathrm{H}_0 : \beta_1 = 2 \quad \text{contro} \quad \mathrm{H}_A : \beta_1 > 2.$$

Se le ipotesi di normalità, indipendenza e omoschedasticità sono corrette, allora possiamo affrontare la verifica d'ipotesi con il test T. Il valore osservato della statistica T è

$$t = (2.45 - 2)/0.24 = 1.88.$$

Sotto l'ipotesi nulla, la statistica T ha distribuzione T di Student con 10 gradi di libertà. Il valore osservato della statistica T conduce ad un livello di significatività osservato pari a

$$\text{p-value} = \Pr(T \geq 1.88) = 0.04,$$

che indica una debole evidenza che l'importo investito sia aumentato mediamente più di 2 000 dollari ogni anno.

4. I dati sintetici di Anscombe mostrati nell'ultima pagina dell'Unità 5 sono disponibili nel dataset **anscombe** del pacchetto **datasets** che fa parte della distribuzione di base di **R**. Come spiegato nella guida in linea, **anscombe** contiene quattro coppie di risposte e predittori indicate come (**x1**, **y1**), (**x2**, **y2**), (**x3**, **y3**) e (**x4**, **y4**).

   Per ogni coppia di risposte e predittori:

   (a) Si stimi la retta di regressione.

   (b) Si discuta l'adattamento della retta di regressione ai dati.

   Soluzione

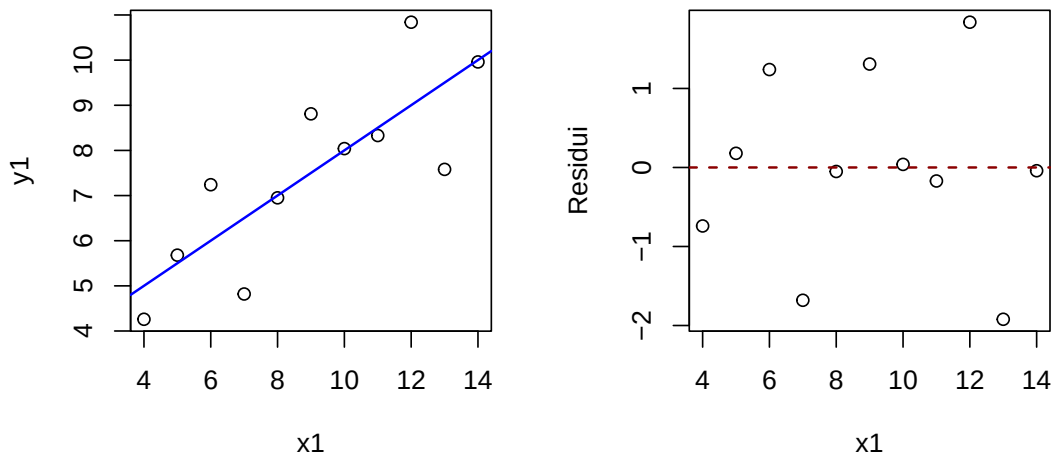
   (a) **Prima coppia**. La retta di regressione stimata è:

```
R> mod <- lm(y1 ~ x1, data = anscombe)
R> summary(mod)

##
## Call:
## lm(formula = y1 ~ x1, data = anscombe)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -1.92127 -0.45577 -0.04136  0.70941  1.83882
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0001     1.1247   2.667  0.02573 *
## x1            0.5001     0.1179   4.241  0.00217 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6665,Adjusted R-squared:  0.6295
## F-statistic: 17.99 on 1 and 9 DF,  p-value: 0.00217
```

Visualizziamo i dati con la retta stimata e i corrispondenti residui:

```
R> par(mfrow = c(1, 2))
R> plot(y1 ~ x1, data = anscombe)
R> abline(mod, col = "blue", lwd = 1.5)
R> plot(resid(mod) ~ x1, data = anscombe, ylab = "Residui")
R> abline(h = 0, col = "darkred", lwd = 1.5, lty = "dashed")
R> par(mfrow = c(1, 1))
```



Il grafico della retta di regressione stimata e i corrispondenti residui sono adeguati.

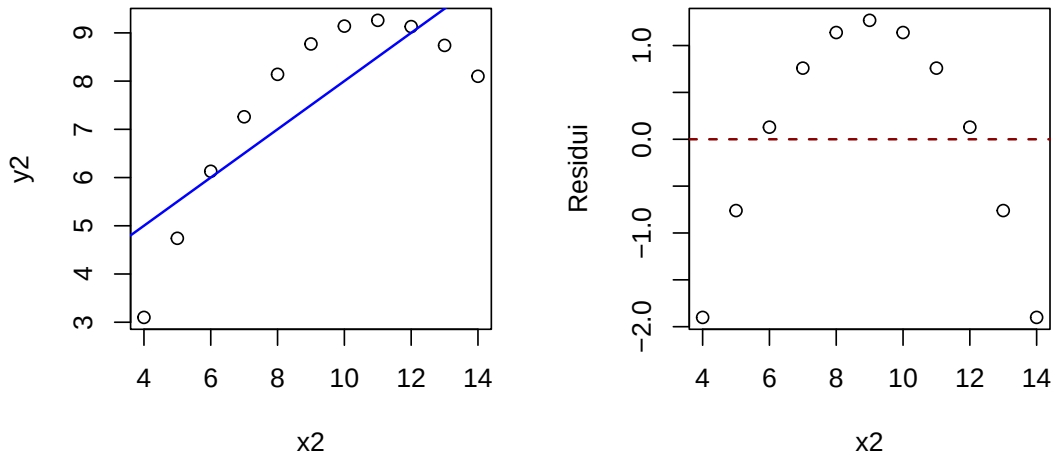
(b) **Seconda coppia**. La retta di regressione stimata è:

```
R> mod <- lm(y2 ~ x2, data = anscombe)
R> summary(mod)

##
## Call:
## lm(formula = y2 ~ x2, data = anscombe)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9009 -0.7609  0.1291  0.9491  1.2691
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.001      1.125   2.667  0.02576 *
## x2             0.500      0.118   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6662,Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002179
```

Visualizziamo i dati con la retta stimata e i corrispondenti residui:

```
R> par(mfrow = c(1, 2))
R> plot(y2 ~ x2, data = anscombe)
R> abline(mod, col = "blue", lwd = 1.5)
R> plot(resid(mod) ~ x2, data = anscombe, ylab = "Residui")
R> abline(h = 0, col = "darkred", lwd = 1.5, lty = "dashed")
R> par(mfrow = c(1, 1))
```



I dati sono perfettamenti disposti lungo una parabola, quindi il modello di regressione lineare è del tutto inappropriato. Ovviamente, stimando un modello polinomiale di ordine due otteniamo una perfetta descrizione dei dati:

```
R> summary(update(mod, . ~ . + I(x2 ^ 2)))

##
## Call:
## lm(formula = y2 ~ x2 + I(x2^2), data = anscombe)
```

```
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0013287 -0.0011888 -0.0006294  0.0008741  0.0023776
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.9957343  0.0043299   -1385   <2e-16 ***
## x2           2.7808392  0.0010401    2674   <2e-16 ***
## I(x2^2)     -0.1267133  0.0000571   -2219   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.001672 on 8 degrees of freedom
## Multiple R-squared:      1,Adjusted R-squared:      1
## F-statistic: 7.378e+06 on 2 and 8 DF,  p-value: < 2.2e-16
```

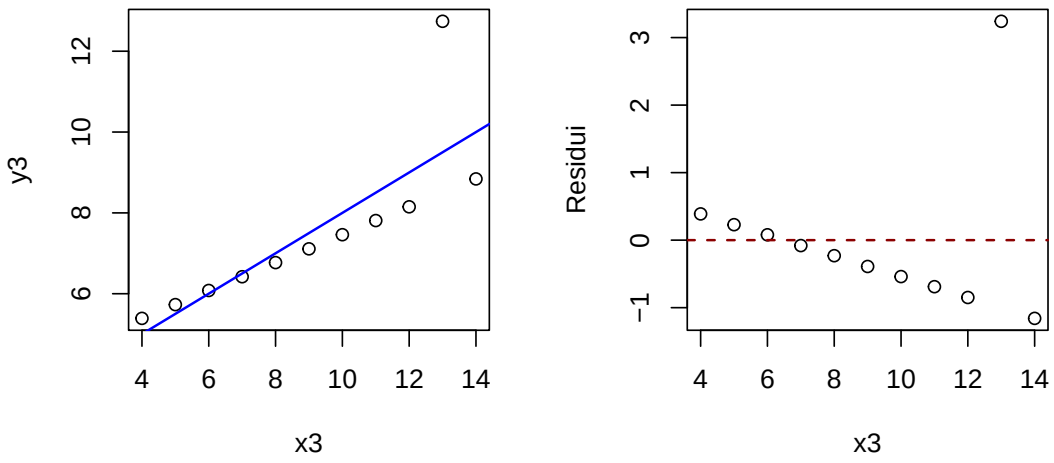
(c) **Terza coppia**. La retta di regressione stimata è:

```
R> mod <- lm(y3 ~ x3, data = anscombe)
R> summary(mod)
```

```
##
## Call:
## lm(formula = y3 ~ x3, data = anscombe)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1586 -0.6146 -0.2303  0.1540  3.2411
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0025     1.1245   2.670  0.02562 *
## x3            0.4997     0.1179   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6663,Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002176
```

Visualizziamo i dati con la retta stimata e i corrispondenti residui:

```
R> par(mfrow = c(1, 2))
R> plot(y3 ~ x3, data = anscombe)
R> abline(mod, col = "blue", lwd = 1.5)
R> plot(resid(mod) ~ x3, data = anscombe, ylab = "Residui")
R> abline(h = 0, col = "darkred", lwd = 1.5, lty = "dashed")
R> par(mfrow = c(1, 1))
```



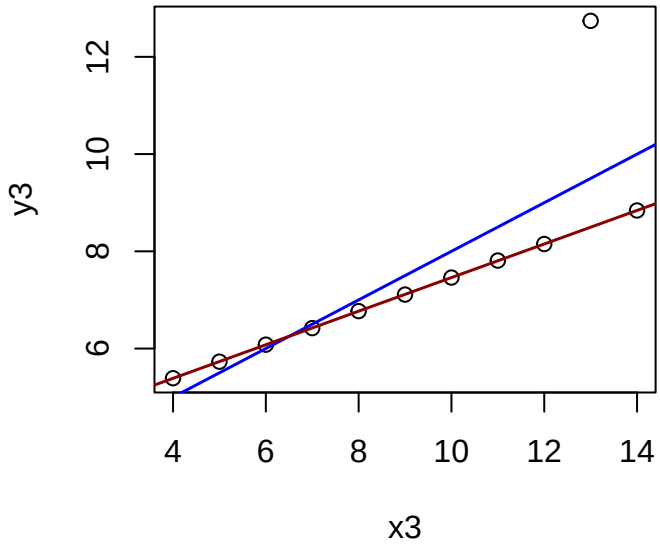
I grafici mostrano un'osservazione influente che determina una retta di regressione gsbagliata" per le rimanenti osservazioni che sono perfettamente allineate su un'altra retta. L'osservazione influente è la terza:

```
R> which.max(anscombe$y3)

## [1] 3
```

Ristimando il modello senza la terza osservazione, otteniamo, ovviamente, una nuova retta di regressione che descrive perfettamente i dati. Per stimare la retta di regressione escludendo la terza osservazione usiamo l'opzione **subset** di **lm** come segue:

```
R> mod2 <- lm(y3 ~ x3, data = anscombe, subset = -3)
R> summary(mod2)

##
## Call:
## lm(formula = y3 ~ x3, data = anscombe, subset = -3)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0041558 -0.0022240  0.0000649  0.0018182  0.0050649
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.0056494  0.0029242    1370   <2e-16 ***
## x3          0.3453896  0.0003206    1077   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003082 on 8 degrees of freedom
## Multiple R-squared:      1,Adjusted R-squared:      1
## F-statistic: 1.161e+06 on 1 and 8 DF,  p-value: < 2.2e-16
```

```
R> plot(y3 ~ x3, data = anscombe)
R> abline(mod, col = "blue", lwd = 1.5)
R> abline(mod2, col = "darkred", lwd = 1.5)
```

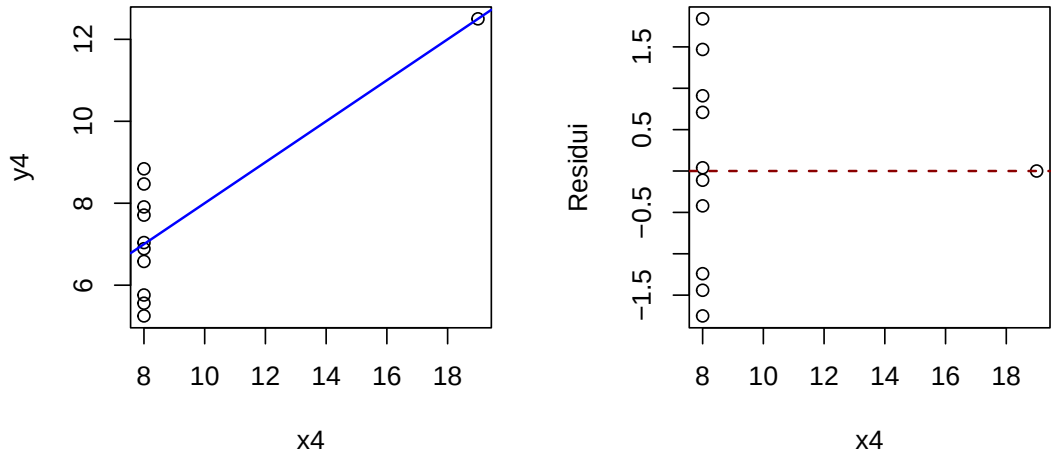


(d) **Quarta coppia**. La retta di regressione stimata è:

```
R> mod <- lm(y4 ~ x4, data = anscombe)
R> summary(mod)

##
## Call:
## lm(formula = y4 ~ x4, data = anscombe)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.751 -0.831  0.000  0.809  1.839
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0017     1.1239   2.671  0.02559 *
## x4            0.4999     0.1178   4.243  0.00216 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6667,Adjusted R-squared:  0.6297
## F-statistic:    18 on 1 and 9 DF,  p-value: 0.002165
```

Visualizziamo i dati con la retta stimata e i corrispondenti residui:

```
R> par(mfrow = c(1, 2))
R> plot(y4 ~ x4, data = anscombe)
R> abline(mod, col = "blue", lwd = 1.5)
R> plot(resid(mod) ~ x4, data = anscombe, ylab = "Residui")
R> abline(h = 0, col = "darkred", lwd = 1.5, lty = "dashed")
R> par(mfrow = c(1, 1))
```



Anche in questo caso vi è la presenza di un'osservazione influente che porta a stimare una retta di regressione che non descrive le restanti osservazioni. Si tratta dell'ottava osservazione:

```
R> which.max(anscombe$y4)

## [1] 8
```

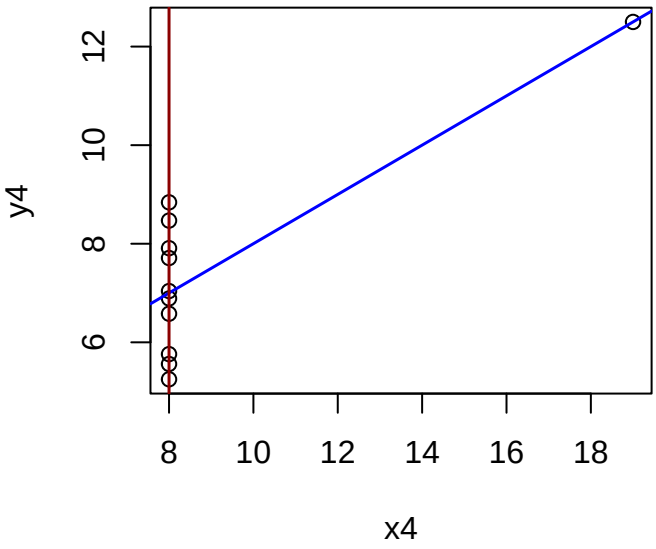
Ristimando la retta di regressione senza l'ottava osservazione otteniamo

```
R> mod2 <- lm(y4 ~ x4, data = anscombe, subset = -8)
R> summary(mod2)

##
## Call:
## lm(formula = y4 ~ x4, data = anscombe, subset = -8)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.751 -1.036 -0.036  0.859  1.839
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.0010     0.3908   17.92 2.39e-08 ***
## x4                NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.236 on 9 degrees of freedom
```

Non troviamo alcun valore per il coefficiente angolare stimato (**NA = not available**) poiché le osservazioni sono perfettamente allineate verticalmente:

```
R> plot(y4 ~ x4, data = anscombe)
R> abline(mod, col = "blue", lwd = 1.5)
R> abline(v = 8, col = "darkred", lwd = 1.5)
```



5. L'efficienza di un programma dipende dalla dimensione dei dati che riceve come input. In generale, dataset di dimensioni più grandi richiedono tempi di elaborazione più elevati, riducendo il numero di processi elaborati in una data unità di tempo. Il dataset **efficienza.csv** contiene il numero di processi elaborati per ora per un campione casuale di 29 dataset di varie dimensioni misurate in Gigabyte.

   (a) Si costruisca un modello per prevedere quanti processi vengono elaborati in un'ora in funzione della dimensione dei dati ricevuti dal programma.

   (b) Si illustri il modello calcolando un intervallo di previsione al 95% per il numero di processi elaborati in un'ora nel caso di (i) un dataset di dimensione 10 Gigabyte e (ii) nel caso di un dataset di dimensione 15 Gigabyte.

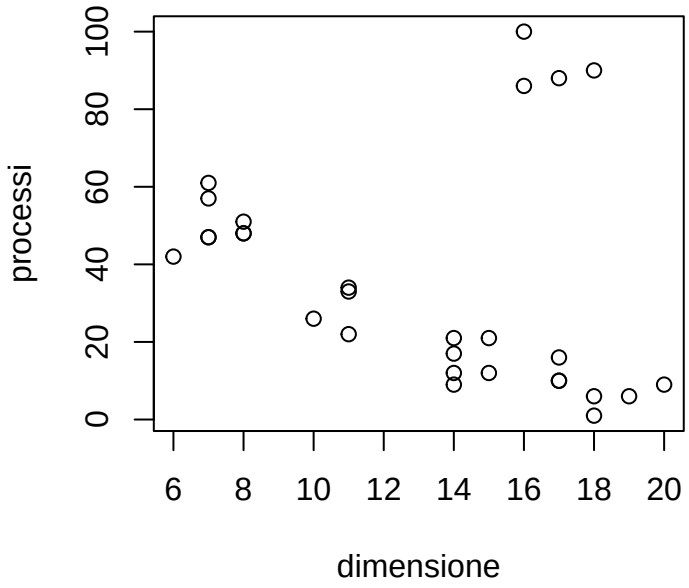
Soluzione

(a) Leggiamo i dati:

```
R> efficienza <- read.csv("efficienza.csv")
```

Visualizziamo i dati:

```
R> plot(processi ~ dimensione, data = efficienza)
```



Il grafico mostra un andamento decrescente non lineare del numero di processi in funzione della dimensione per la gran parte delle osservazioni a meno di un grappolo di quattro osservazioni con un numero di processi elaborati particolarmente grande nonostante la dimensione dei corrispondenti datasets.

Le osservazioni anomale sono:

```
R> outliers <- which(efficienza$processi > 80)
R> outliers
## [1]  8  9 18 29
```

Stimiamo un modello quadratico su tutti i dati:

```
R> mod <- lm(processi ~ dimensione + I(dimensione ^ 2), data = efficienza)
R> summary(mod)
##
## Call:
## lm(formula = processi ~ dimensione + I(dimensione^2), data = efficienza)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.930 -19.101  -9.357   3.668  70.423
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      78.5531    54.4679   1.442    0.161
```

```
## dimensione       -5.4942     9.4102  -0.584    0.564
## I(dimensione^2)   0.1521     0.3700   0.411    0.684
##
## Residual standard error: 28.38 on 26 degrees of freedom
## Multiple R-squared:  0.07223,Adjusted R-squared:  0.0008619
## F-statistic: 1.012 on 2 and 26 DF,  p-value: 0.3773
```

Nel modello calcolato su tutti i dati, nessun coefficiente di regressione è significativo e l'indice R^2 è pressoché nullo

Vediamo ora lo stesso modello togliendo i dati anomali:

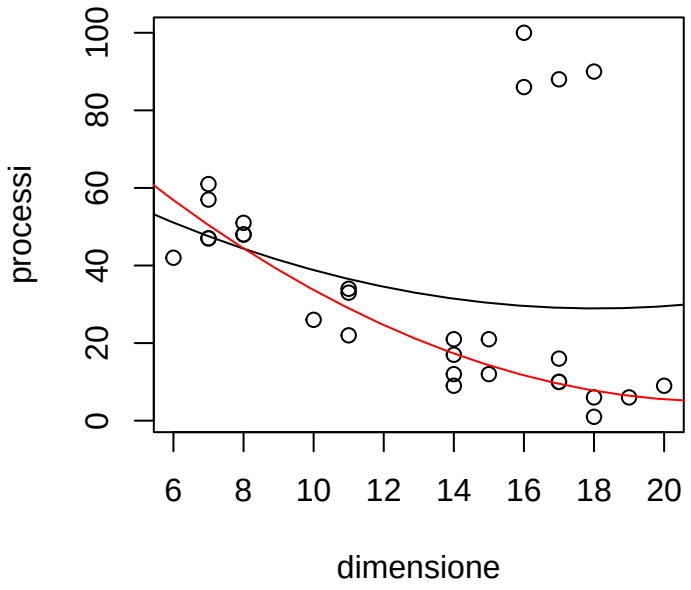
```
R> mod2 <- lm(processi ~ dimensione + I(dimensione ^ 2), data = efficienza,
+ subset = -outliers)
R> summary(mod2)
```

```
##
## Call:
## lm(formula = processi ~ dimensione + I(dimensione^2), data = efficienza,
##     subset = -outliers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.8440  -3.4158   0.5015   4.0487  10.5842
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     104.33360   12.36306   8.439  2.4e-08 ***
## dimensione       -9.18934    2.13998  -4.294 0.000294 ***
## I(dimensione^2)   0.21240    0.08412   2.525 0.019283 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.333 on 22 degrees of freedom
## Multiple R-squared:  0.892,Adjusted R-squared:  0.8822
## F-statistic:  90.9 on 2 and 22 DF,  p-value: 2.321e-11
```

In questo modello, tutti i coefficienti di regressione sono significativi e l'indice R^2 è pari a 0.89.

Visualizziamo i due modelli stimati:

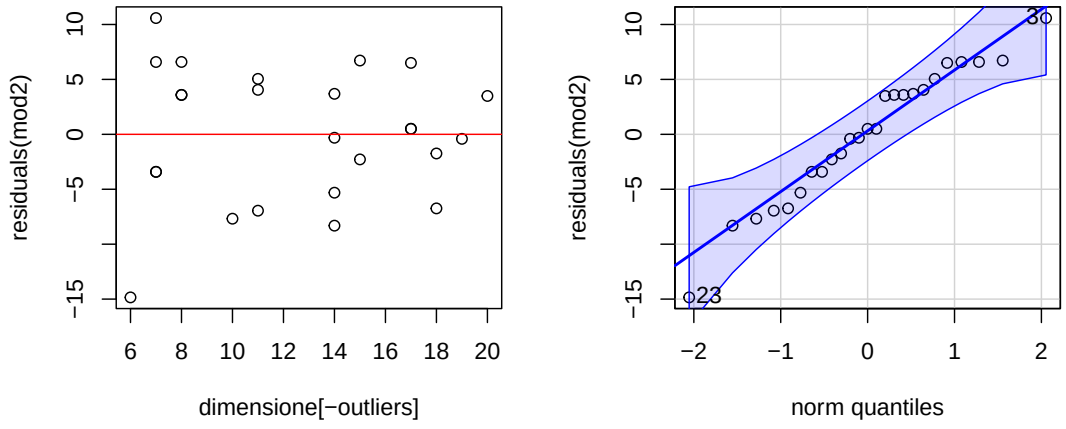
```
R> plot(processi ~ dimensione, data = efficienza)
R> curve(coef(mod)[1] + coef(mod)[2] * x + coef(mod)[3] * x ^ 2,
+ from = min(efficienza$processi), to = max(efficienza$processi),
+ add = TRUE)
R> curve(coef(mod2)[1] + coef(mod2)[2] * x + coef(mod2)[3] * x ^ 2,
+ from = min(efficienza$processi), to = max(efficienza$processi),
+ add = TRUE, col = "red")
```



Il modello stimato togliendo i quattro outliers si adatta decisamente meglio alle osservazioni a confermare che i punti anomali andavano rimossi.
Vediamo l'analisi dei residui del modello con gli outliers rimossi:

```
R> library(car)
R> par(mfrow = c(1, 2))
R> plot(residuals(mod2) ~ dimensione[-outliers], data = efficienza)
R> abline(h = 0, col = "red")
R> qqPlot(residuals(mod2))
```

```
## 23  3
## 20  3
```



L'analisi dei residui è soddisfacente. Procediamo, comunque, con il modello stimato senza i valori anomali che

- spiega quasi il 90% della variabilità del numero di processi elaborati;
- conferma la relazione decrescente non lineare fra numero di processi elaborati e dimensione dei datasets.

(b) Calcoliamo l'intervallo di previsione al 95% per il numero di processi elaborati con un dataset di dimensione 10 Gigabyte:

```
R> predict(mod2, newdata = data.frame(dimensione = 10), interval =
+ "prediction")
```

```
##        fit      lwr      upr
## 1 33.68022 20.02937 47.33106
```

Se invece la dimensione fosse pari a 15 Gigabyte allora il numero di processi elaborati previsto dal modello scende a:

```
R> predict(mod2, newdata = data.frame(dimensione = 15), interval =
+ "prediction")
```

```
##        fit       lwr      upr
## 1 14.28353 0.6057741 27.96129
```

6. Il dataset **iq.csv** contiene il reddito (in migliaia di dollari) e il quantile del quoziente intellettivo IQ per un campione casuale di 94 persone. Per esempio, se un individuo ha un quantile pari a 0.9 vuol dire che solo il 10% della popolazione ha un quoziente intellettivo più alto di quel individuo.

   (a) Si costruisca e commenti un modello per prevedere il reddito in funzione del quoziente intellettivo.

   (b) Si illustri il modello calcolando un intervallo di previsione al 95% per il reddito di un individuo il cui quantile del quoziente intellettivo sia pari a (i) 0.5 e (ii) 0.9.

Soluzione

(a) Leggiamo i dati:

```
R> iq <- read.csv("iq.csv")
```

Stimiamo la retta di regressione per prevedere il reddito con il quoziente intellettivo:

```
R> mod <- lm(reddito ~ IQ, data = iq)
R> summary(mod)
```

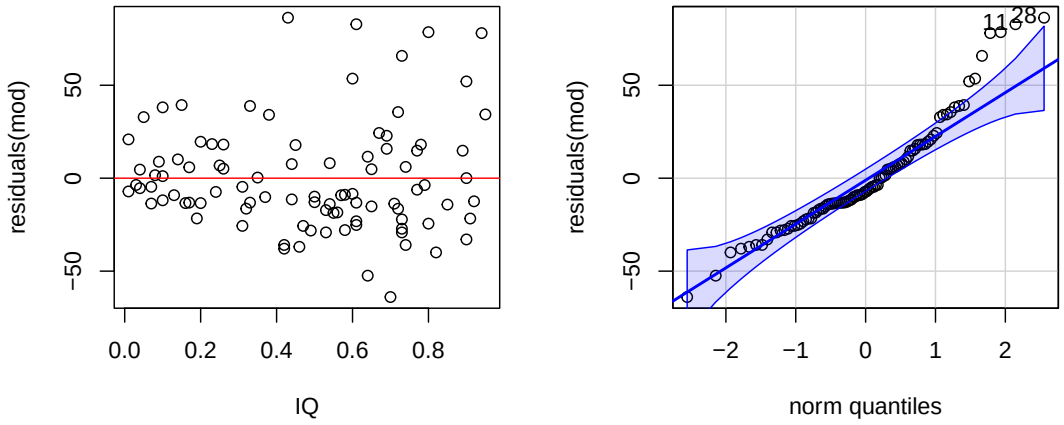
```
##
## Call:
## lm(formula = reddito ~ IQ, data = iq)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -63.934 -16.987  -7.265  14.808  86.336
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   45.381      6.149   7.380 6.85e-11 ***
## IQ            75.076     11.087   6.771 1.18e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.2 on 92 degrees of freedom
## Multiple R-squared:  0.3326,Adjusted R-squared:  0.3254
## F-statistic: 45.85 on 1 and 92 DF,  p-value: 1.177e-09
```

Controlliamo i residui:

```
R> library(car)
R> par(mfrow = c(1, 2))
R> plot(residuals(mod) ~ IQ, data = iq)
R> abline(h = 0, col = "red")
R> qqPlot(residuals(mod))
```

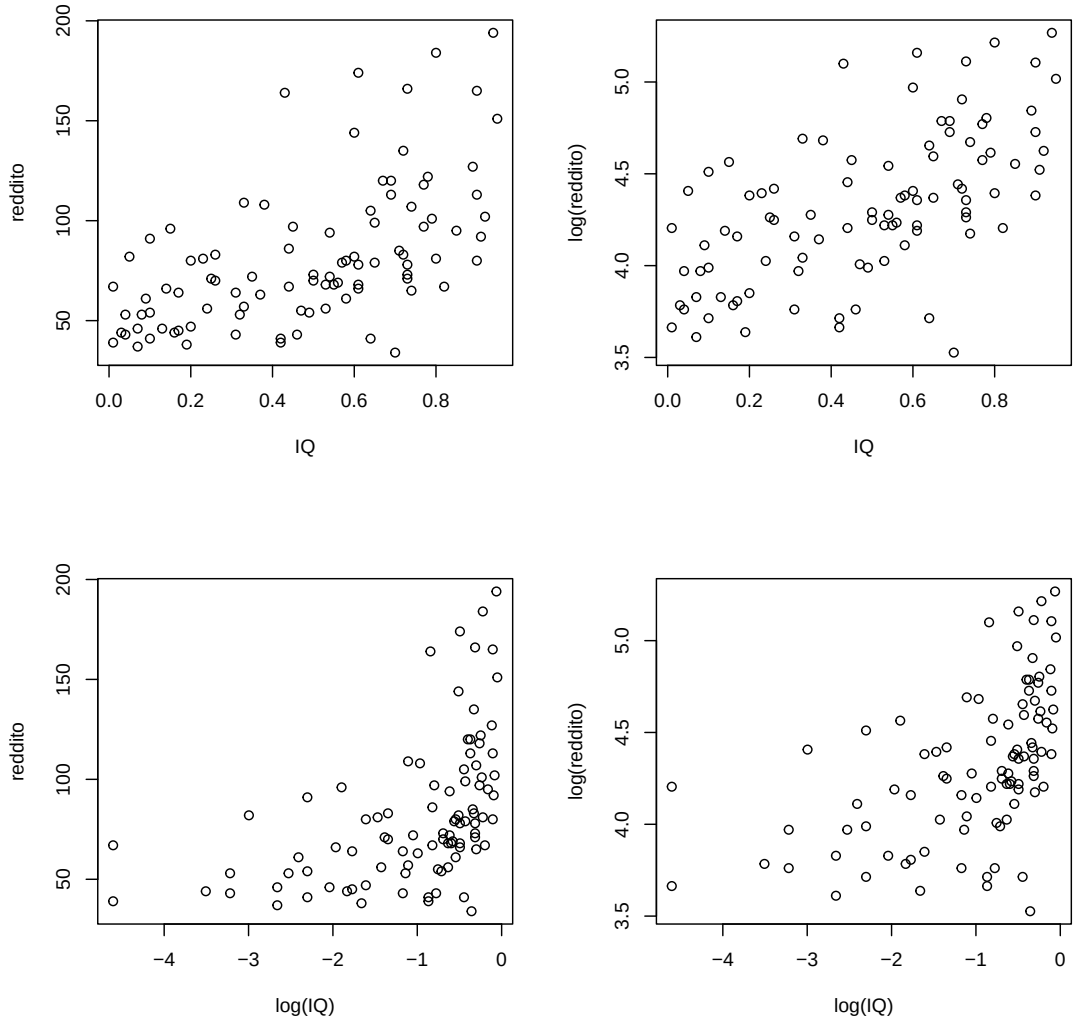
```
## [1] 28 11
```



L'analisi dei residui però non è soddisfacente perché mostra evidenza di eterogeneità e vi è deviazione dall'assunzione di normalità sulla coda di destra.

Proviamo a trasformare la risposta o il predittore su scala logaritmica:

```
R> par(mfrow = c(2,2))
R> plot(reddito ~ IQ, data = iq)
R> plot(log(reddito) ~ IQ, data = iq)
R> plot(reddito ~ log(IQ), data = iq)
R> plot(log(reddito) ~ log(IQ), data = iq)
```



I grafici a dispersione suggeriscono di trasformare solo la risposta. Stimiamo, quindi, il modello su scala logaritmica:

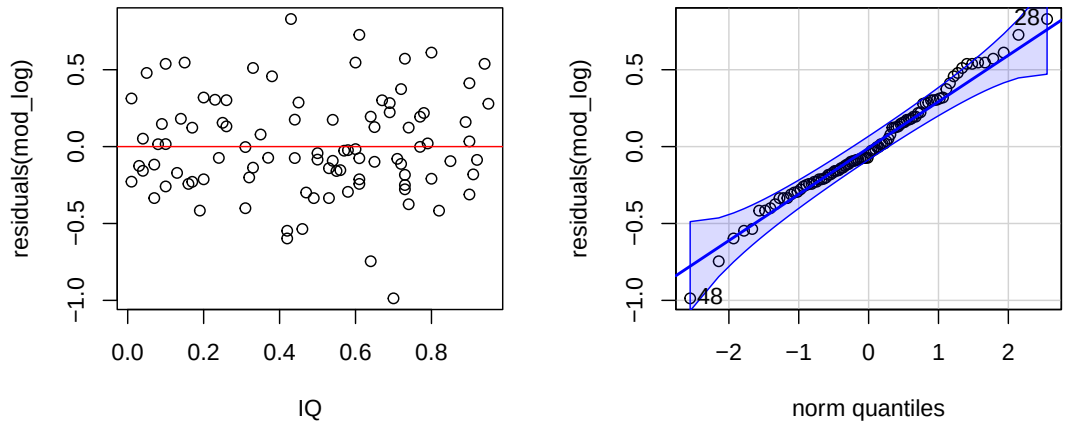
```
R> mod_log <- lm(log(reddito) ~ IQ, data = iq)
R> summary(mod_log)

##
## Call:
## lm(formula = log(reddito) ~ IQ, data = iq)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98693 -0.21171 -0.05772  0.19463  0.82985
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.88257    0.06905  56.230  < 2e-16 ***
## IQ           0.90102    0.12451   7.237 1.35e-10 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3279 on 92 degrees of freedom
## Multiple R-squared:  0.3627,Adjusted R-squared:  0.3558
## F-statistic: 52.37 on 1 and 92 DF,  p-value: 1.348e-10
```

Il predittore IQ è fortemente significativo e il modello di regressione spiega il 36% della variabilità del reddito su scala logaritmica. Vediamo i residui:

```
R> library(car)
R> par(mfrow = c(1, 2))
R> plot(residuals(mod_log) ~ IQ, data = iq)
R> abline(h = 0, col = "red")
R> qqPlot(residuals(mod_log))
```

```
## [1] 48 28
```



L'analisi dei residui del modello su scala logaritmica è soddisfacente.

(b) Calcoliamo gli intervalli di previsione richiesti:

```
R> pr <- predict(mod_log, newdata = data.frame(IQ = c(0.5, 0.9)),
+ interval = "prediction", level = 0.95)
```

e li trasformiamo sulla scala originale:

```
R> exp(pr)
```

```
##         fit      lwr      upr
## 1  76.17902 39.58456 146.6037
## 2 109.23430 56.30644 211.9142
```

Le previsioni mostrano che avere un alto quoziente intellettivo da un chiaro vantaggio in termini di reddito atteso per quanto la variabilità di previsione è molto ampia ad indicare che il quoziente intellettivo non è sufficiente per prevedere il reddito con buona precisione.