CT0429 - Analisi Predittiva - aa 21/22 - Appello IV

Nome Cognome - matricola

- Istruzioni
- Esercizio 1
  - Es. 1.a
  - Es. 1.b
  - Es. 1.c
  - Es. 1.d
- Esercizio 2
  - Es. 2.a
  - Es. 2.b
  - Es. 2.c
  - Es. 2.d

Istruzioni

Salvate questo file con il nome `matricola.Rmd` . Questo sarà il file che dovrete consegnare. Il file deve compilare senza problemi: files che non possono essere compilati correttamente saranno penalizzati.

Per essere sicuri di poter trovare il file al momento della consegna potete controllare dove è il file quando compilate il file la prima volta usando il comando `getwd()` .

```
getwd()
```

```
## [1] "C:/Users/Dav/Downloads"
```

```
## da cancellare quando siete sicuri di dove è il file
```

Attenzione - per tutto l'esame, se non specificato esplicitamente, il livello di significatività da usare è $\alpha = 0.02$

Esercizio 1

Un app di smart-mobility desidera indagare quali fattori influenzino l'utilizzo degli utenti dei mezzi messi a disposizione dalla app. Per un campione di 70 giorni in diverse città nei mesi di Aprile e Maggio vengono misurate le seguenti informazioni:

- `n_mezzi` : il numero minimo di mezzi funzionanti e operativi nella giornata
- `temp` : la temperatura media della giornata
- `weekend` : una variabile che indica se la giornata era un sabato o domenica
- `usage` : una variabile che indica il numero di chilometri (in migliaia) coperto dagli utenti nella giornata. Questa è la variabile risposta

Le informazioni sulle variabili sono disponibili nel dataset `dex1` che si può caricare usando il seguente codice:

```
dex1 <- read.csv("ex1_data.csv", header = TRUE)
dex1
```

```
##    n_mezzi  temp weekend usage
## 1      201 20.04      no  78.1
## 2      196 29.74      no  83.5
## 3      162 25.49      no  59.7
## 4      186 27.91      si  74.1
## 5      181 22.38      si  73.7
## 6      196 28.25      si  81.3
## 7      152 29.77      no  63.7
## 8      194 15.96      no  82.1
## 9      171 27.26      si  58.9
## 10     152 28.63      si  57.9
## 11     194 20.42      no  69.6
## 12     153 28.92      no  64.5
## 13     192 16.34      si  82.7
## 14     150 22.77      no  42.1
## 15     167 29.39      no  69.1
## 16     190 18.86      no  66.9
## 17     160 15.70      no  47.6
## 18     213 24.25      no  95.5
## 19     151 26.86      no  49.9
## 20     179 24.87      no  66.9
## 21     189 21.02      si  68.1
## 22     188 16.52      si  64.3
## 23     209 29.33      no  99.3
## 24     199 25.50      si  82.4
## 25     209 19.27      si  85.6
## 26     174 23.59      no  57.2
## 27     199 17.16      si  73.5
## 28     190 18.54      no  69.1
## 29     202 29.19      no  81.9
## 30     199 16.44      no  68.3
## 31     181 25.64      no  72.5
## 32     166 25.88      no  70.8
## 33     150 25.63      si  51.1
## 34     214 23.71      no  84.7
## 35     175 28.66      si  68.8
## 36     208 28.56      no  92.8
## 37     194 26.08      si  69.4
## 38     194 24.03      no  72.9
## 39     193 22.39      no  80.0
## 40     168 29.59      no  66.7
## 41     178 24.84      si  65.5
## 42     176 15.37      no  63.7
## 43     210 18.93      si  76.3
## 44     173 15.53      no  57.2
## 45     212 22.95      no  96.0
## 46     165 25.73      no  50.9
## 47     162 26.65      si  56.9
## 48     161 27.03      no  61.3
## 49     158 24.27      no  49.8
## 50     176 15.44      si  54.2
## 51     164 20.35      no  54.9
## 52     154 17.72      no  50.9
## 53     195 17.04      no  62.7
```

```
## 54     202 29.59      si  81.8
## 55     203 19.30      no  74.4
## 56     159 26.39      no  62.5
## 57     157 22.06      no  53.2
## 58     214 18.16      no  88.5
## 59     199 15.35      si  82.3
## 60     205 15.11      no  76.3
## 61     160 15.57      si  50.6
## 62     182 17.12      no  70.2
## 63     190 24.24      si  72.6
## 64     184 26.85      no  75.4
## 65     196 28.85      no  84.6
## 66     166 16.11      si  44.3
## 67     171 18.55      no  49.9
## 68     167 26.72      no  62.0
## 69     202 18.17      no  66.9
## 70     201 27.50      no  88.7
```

Es. 1.a

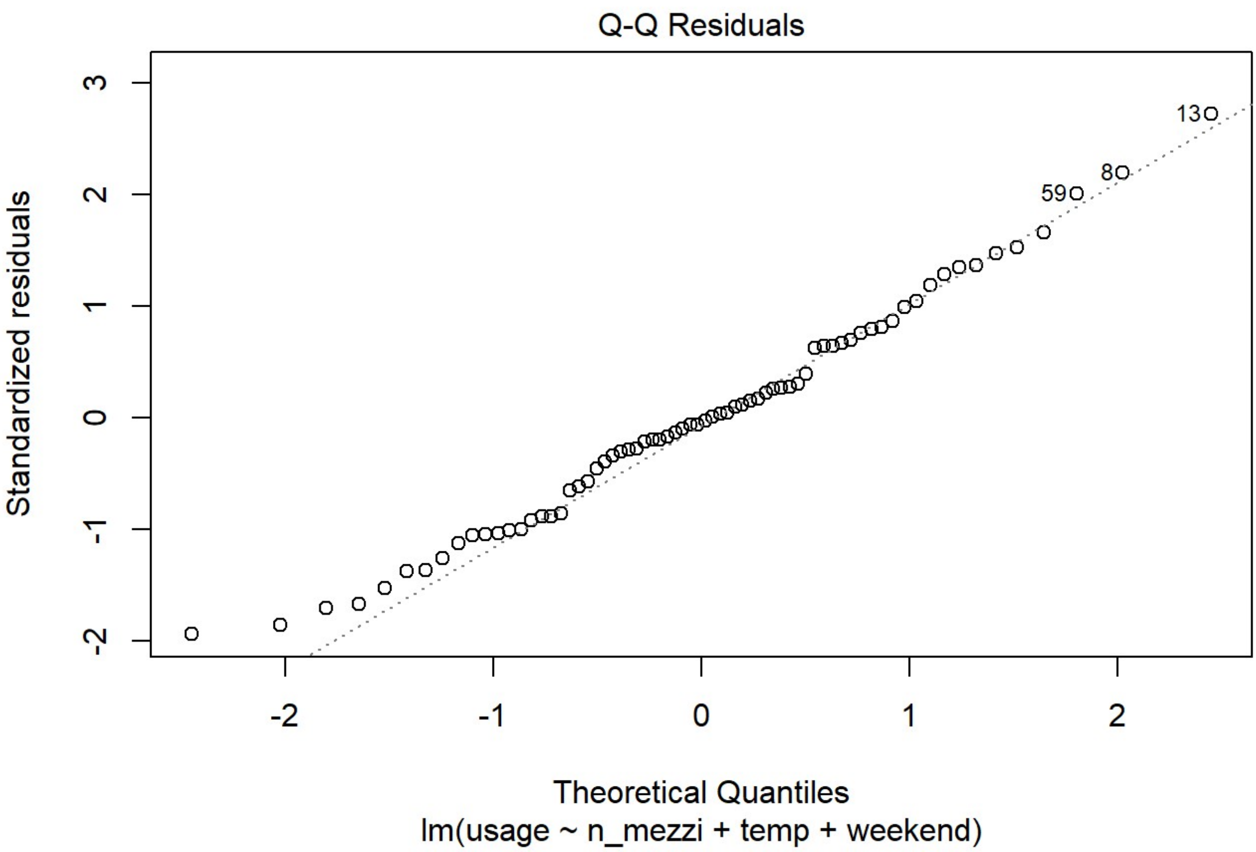
Si costruisca un primo modello lineare multiplo `fit1` in cui tutte le variabili esplicative vengono utilizzate. Si commenti la significatività del modello e dei singoli predittori. Si verifichi l'opportunità di proporre un modello più parsimonioso di `fit1`.

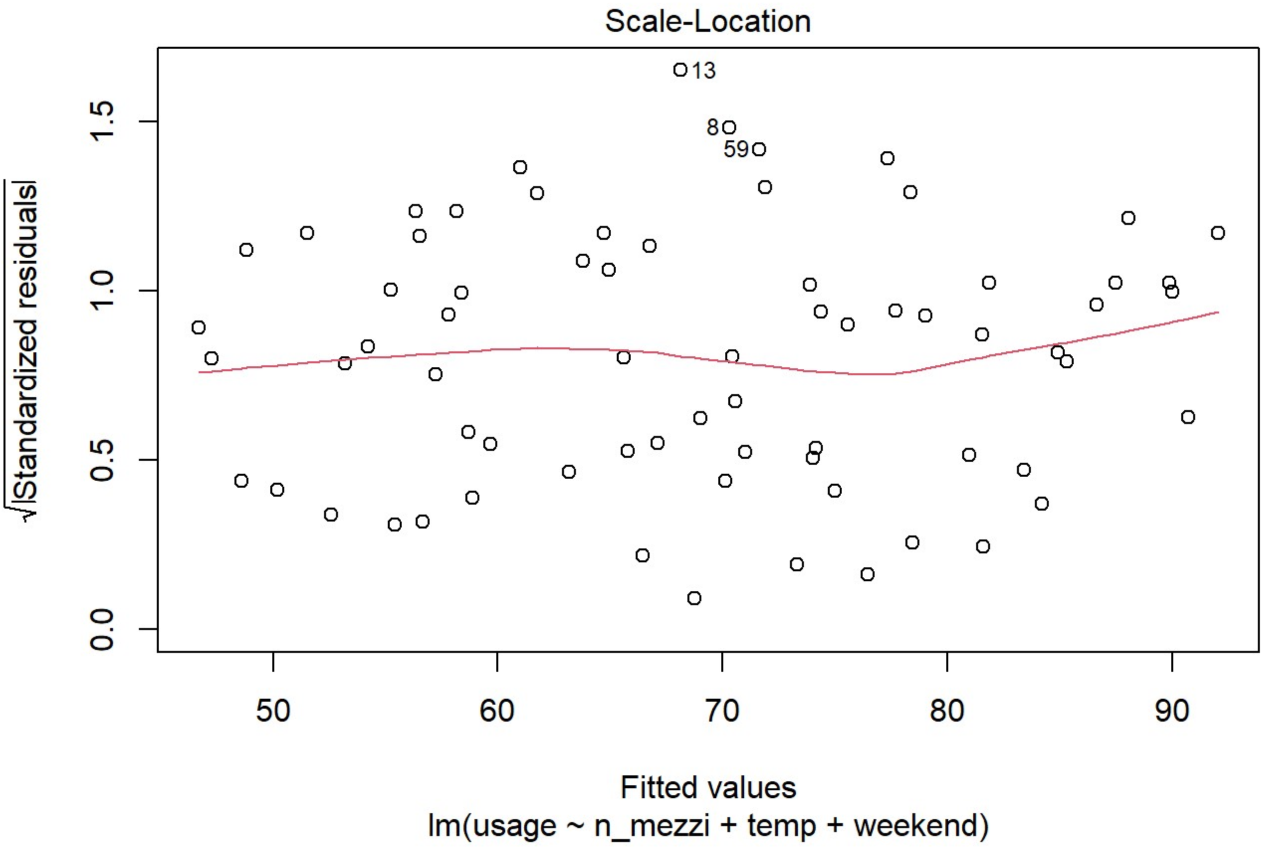
```
fit1 <- lm(usage~n_mezzi+temp+weekend, data = dex1)
summary(fit1)
```

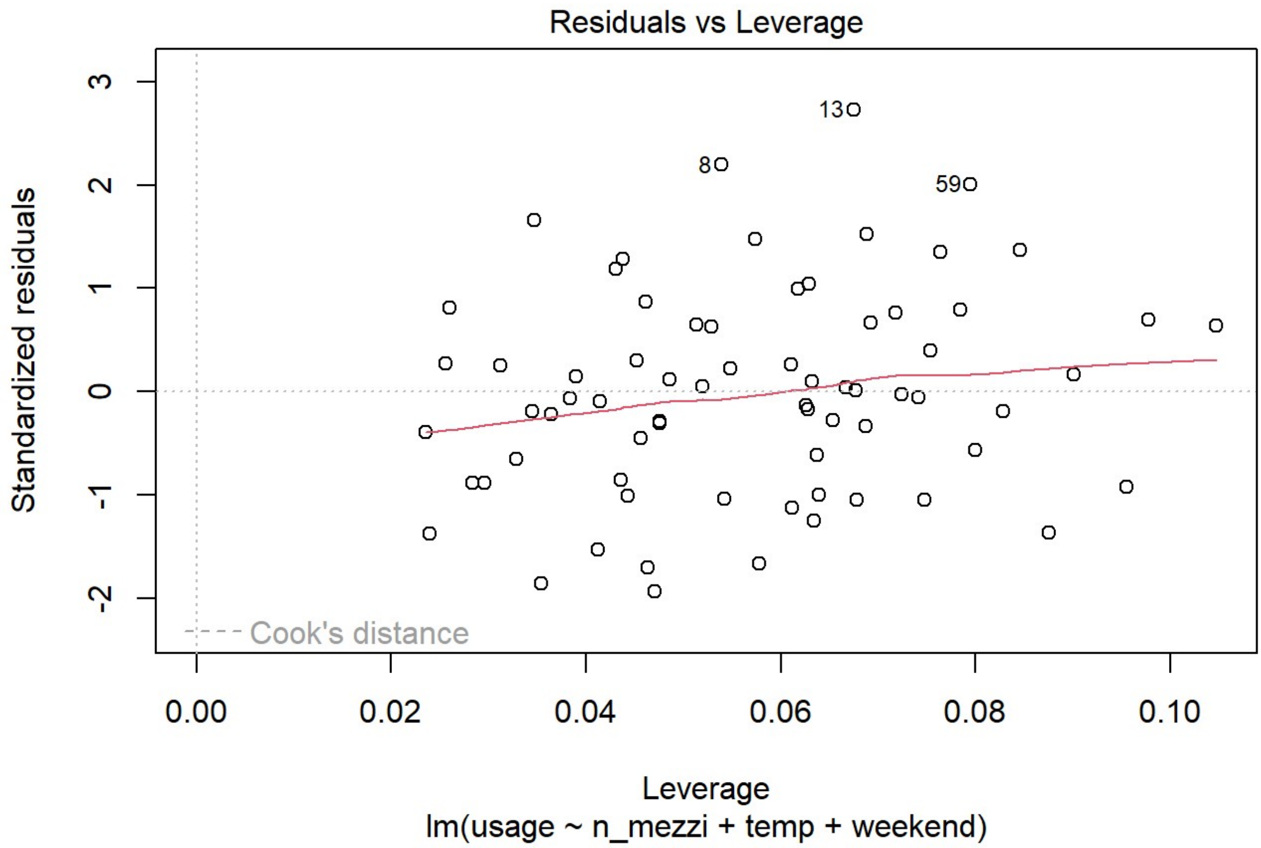
```
##
## Call:
## lm(formula = usage ~ n_mezzi + temp + weekend, data = dex1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.4661  -4.3613  -0.2295   3.5517  14.5649
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -66.7031     7.6498  -8.720 1.37e-12 ***
## n_mezzi       0.6304     0.0352  17.908  < 2e-16 ***
## temp          0.9203     0.1388   6.630 7.34e-09 ***
## weekendsi    -1.2425     1.4099  -0.881    0.381
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.531 on 66 degrees of freedom
## Multiple R-squared:  0.8356, Adjusted R-squared:  0.8281
## F-statistic: 111.8 on 3 and 66 DF,  p-value: < 2.2e-16
```

```
plot(fit1)
```


Residuals vs Fitted
Residuals
Fitted values
lm(usage ~ n_mezzi + temp + weekend)
13
8
59
15
10
5
0
-5
-10
50
60
70
80
90


Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(usage ~ n_mezzi + temp + weekend)
13
8
59
3
2
1
0
-1
-2
-2
-1
0
1
2





I predittori di questo modello sono tutti molto significativi, escluso il parametro booleano “weekend”: l’utilizzo aumenta all’aumentare del numero di mezzi disponibili e della temperatura media della giornata. Il weekend invece ha correlazione negativa, anche se non è significativo e potrebbe aver senso eliminarlo per avere un modello più parsimonioso

Es. 1.b

Si derivino intervalli di confidenza (a livello di confidenza 98%) per il coefficiente angolare relativo alla variabile `temp` nel modello `fit11`. Si verifichi inoltre il sistema di ipotesi $H_0 : \beta_{temp} = 1$ VS $H_1 : \beta_{temp} \neq 1$

```
confint(fit1, level = 0.98)
```

```
##                     1 %        99 %
## (Intercept) -84.9417857 -48.4644880
## n_mezzi       0.5464965   0.7143635
## temp          0.5893622   1.2512911
## weekendsi    -4.6038733   2.1189356
```

l'intervallo di confidenza del relativo a temp è [0.5893622, 1.2512911]. siccome 1 è compreso nell'intervallo di confidenza, non possiamo rifiutare quest'ipotesi

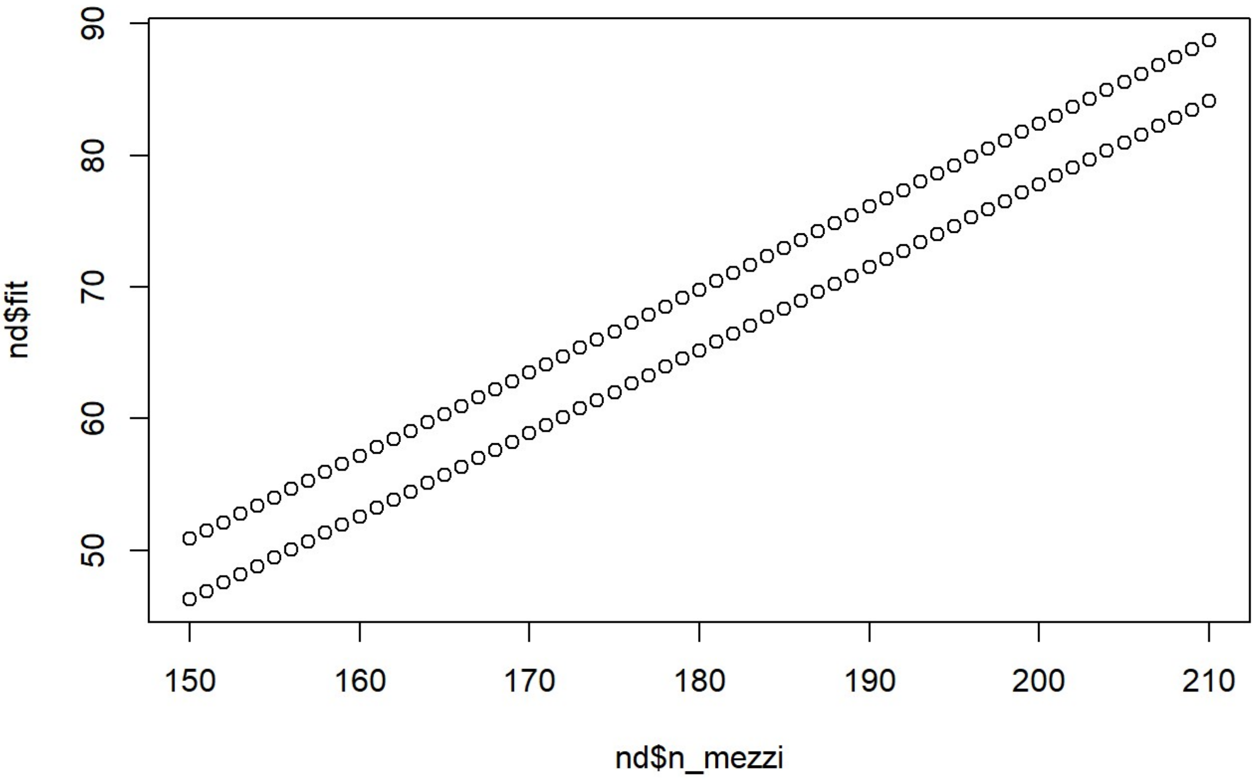
Es. 1.c

Si produca una visualizzazione che mostri i valori stimati dal modello prescelto per giornate feriali con 20 e 25 gradi e tra i 150 e i 210 mezzi disponibili.

```
nd <- data.frame(
  temp = rep(c(20, 25), each = 61),     # 20 ripetuto 61 volte, poi 25 ripetuto 61 volte
  n_mezzi = rep(150:210, times = 2),    # sequenza 150–210 ripetuta per i due blocchi
  weekend = "no"
)

# calcolo valori stimati
nd$fit <- predict(fit1, newdata = nd)

a <- predict(fit1, newdata = nd)
plot (nd$fit~nd$n_mezzi)
```



Es. 1.d

Il CEO dell'azienda desidera valutare l'opportunità di aumentare il numero di mezzi nelle città per il prossimo inverno: è possibile utilizzare il modello selezionato per predirre il numero di chilometri che saranno coperti dagli utenti nei mesi di Dicembre e Gennaio?

```
summary(dex1)
```

```
##     n_mezzi          temp          weekend              usage      
##  Min.   :150.0   Min.   :15.11   Length:70          Min.   :42.10  
##  1st Qu.:166.0   1st Qu.:18.26   Class :character   1st Qu.:59.10  
##  Median :185.0   Median :23.87   Mode  :character   Median :68.95  
##  Mean   :182.6   Mean   :22.83                      Mean   :69.02  
##  3rd Qu.:199.0   3rd Qu.:26.86                      3rd Qu.:79.53  
##  Max.   :214.0   Max.   :29.77                      Max.   :99.30  
```

no perché la temperatura minima è 15 gradi, quindi i dati "invernali" sono troppo fuori dal range e la stima perde di significato

Esercizio 2

Un ristoratore monitora il numero di ordini fatti tramite un app di food-delivery e desidera indagare quali siano i fattori che inducono gli utenti ad ordinare presso il suo ristorante. Le variabili che prende in considerazione sono

- `domenica` : una variabile che indica se la giornata è una Domenica

- `temp` : la temperatura media giornaliera
- `nOrd` : il numero di ordini ricevuti in una serata. Questa è la variabile risposta.

Si carichi il dataset usando il codice seguente:

```
dex2 <- read.csv("ex2_data.csv", header = TRUE)
dex2$domenica <- factor(dex2$domenica)
head(dex2)
```

```
##   domenica      temp nOrd
## 1        0 16.283386   15
## 2        0 15.633810    9
## 3        0  9.045714   18
## 4        1  9.317354   40
## 5        0 16.231297    9
## 6        0 20.347525   12
```

Si desidera costruire un modello predittivo per la variabile `nOrd` , un modello cioè che predica il numero di ordini, usando un modello lineare generalizzato usando una distribuzione di Poisson con funzione legame canonica in cui la variabile `nOrd` è la variabile risposta.

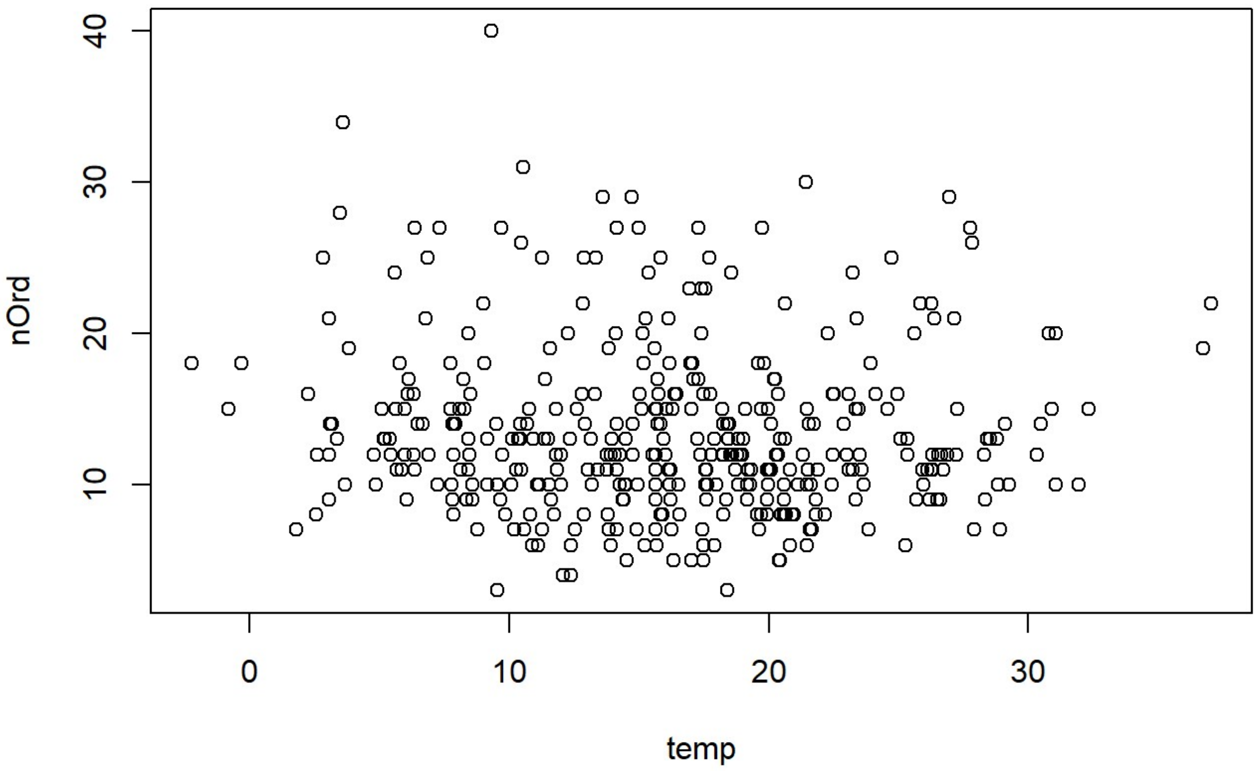
Es. 2.a

Si verifichi se la temperatura è un predittore significativo, verificando inoltre se è conveniente usare termini polinomiali di ordine superiore ad uno (questo si può fare usando la funzione `I` o la funzione `poly`).

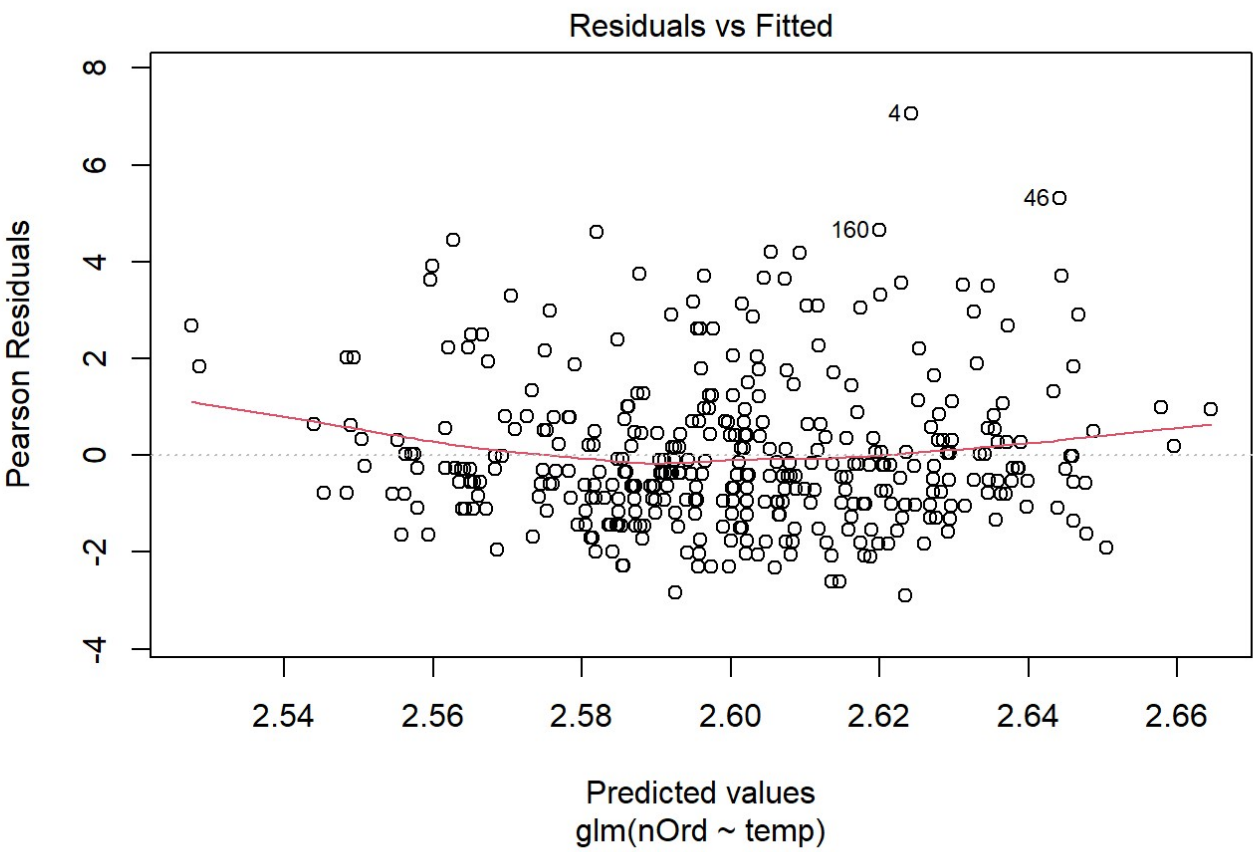
```
fit <- glm(nOrd~temp, data = dex2, family = "poisson")
summary(fit)
```

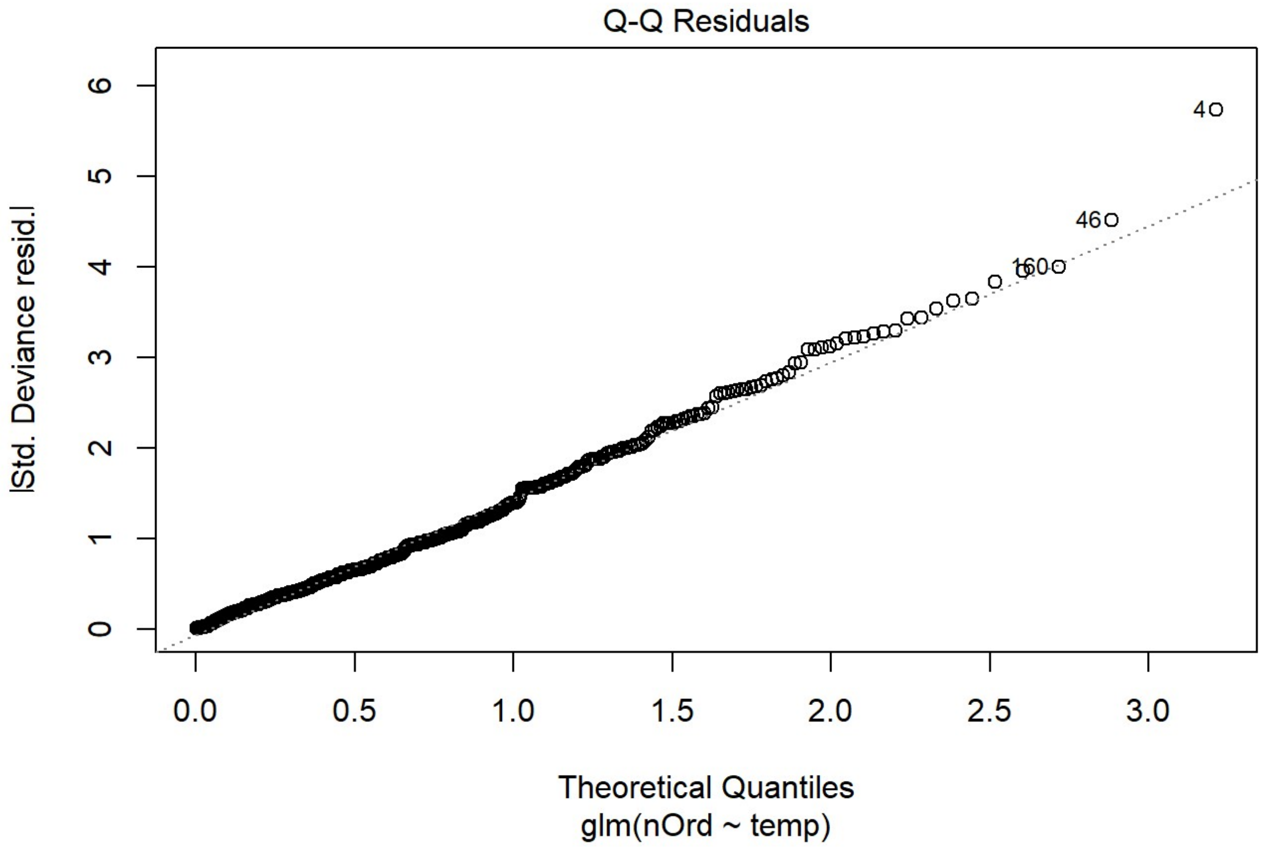
```
## 
## Call:
## glm(formula = nOrd ~ temp, family = "poisson", data = dex2)
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  2.656713   0.034339  77.367   <2e-16 ***
## temp        -0.003488   0.001965  -1.775   0.0759 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for poisson family taken to be 1)
## 
##     Null deviance: 838.53  on 379  degrees of freedom
## Residual deviance: 835.38  on 378  degrees of freedom
## AIC: 2500.2
## 
## Number of Fisher Scoring iterations: 4
```

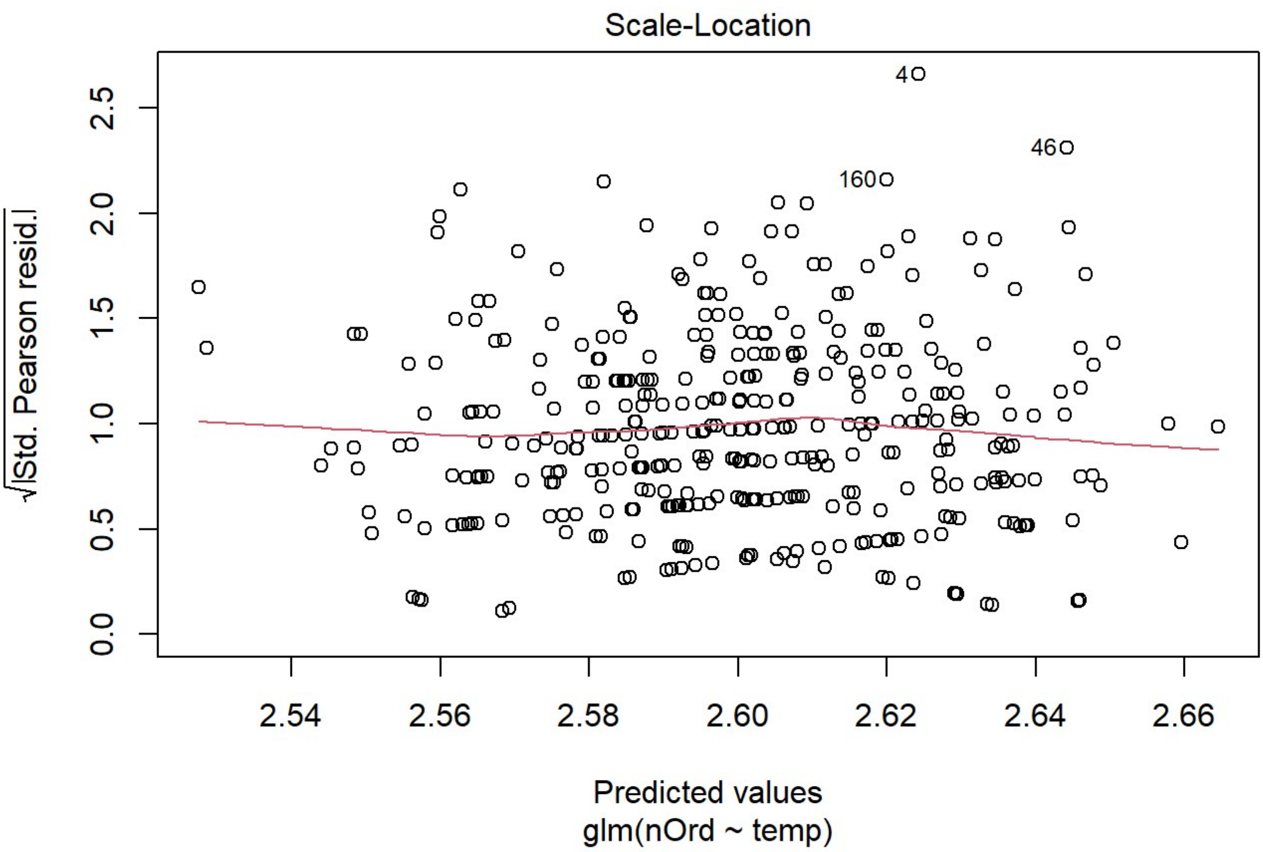
```
plot(nOrd~temp, data = dex2)
```

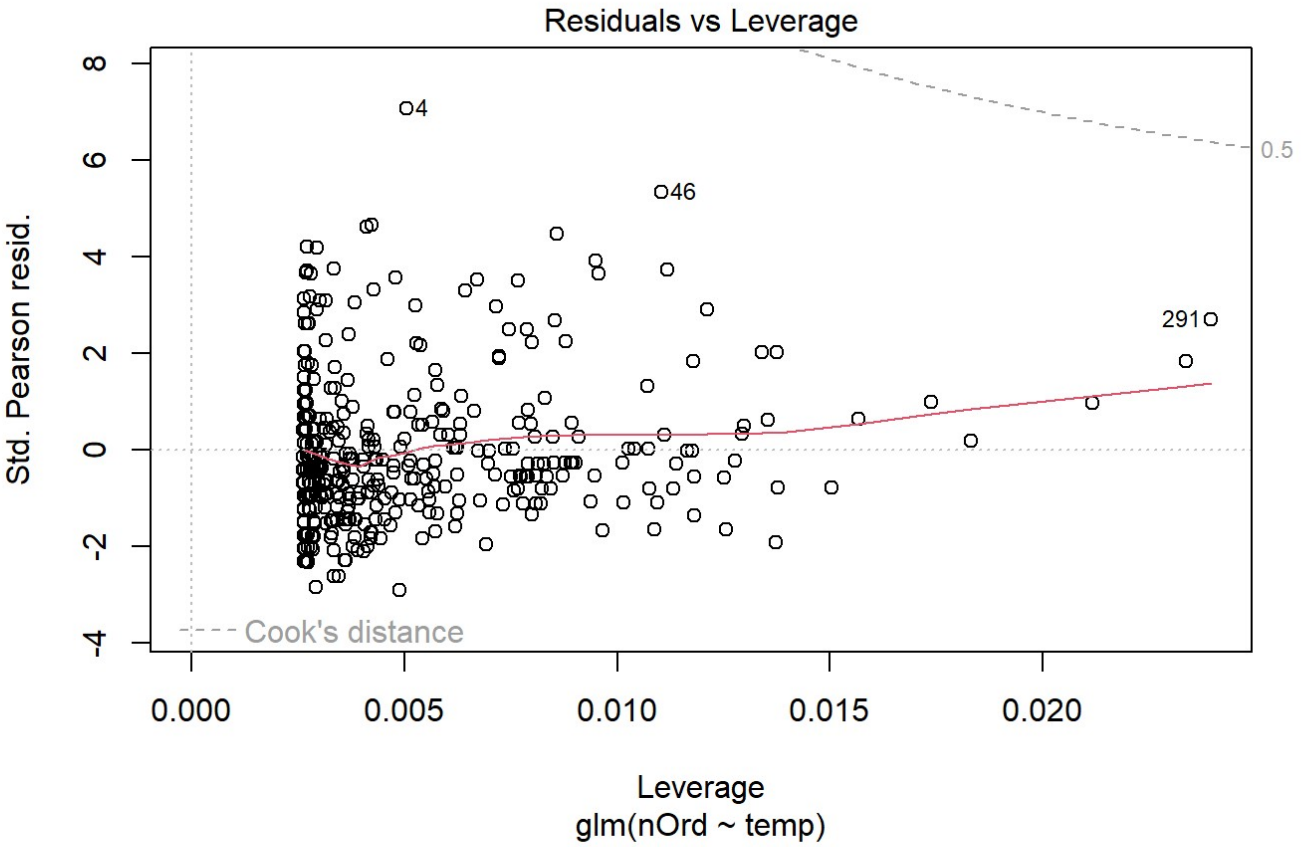

nOrd
temp
0
10
20
30
40

```
plot(fit)
```


Residuals vs Fitted
Pearson Residuals
Predicted values
glm(nOrd ~ temp)
40
46
160


Q-Q Residuals
|Std. Deviance resid.|
0
1
2
3
4
5
6
0.0
0.5
1.0
1.5
2.0
2.5
3.0
4
46
160
Theoretical Quantiles
glm(nOrd ~ temp)


Scale-Location
√|Std. Pearson resid.|
0.0
0.5
1.0
1.5
2.0
2.5
2.54
2.56
2.58
2.60
2.62
2.64
2.66
4
46
160
Predicted values
glm(nOrd ~ temp)

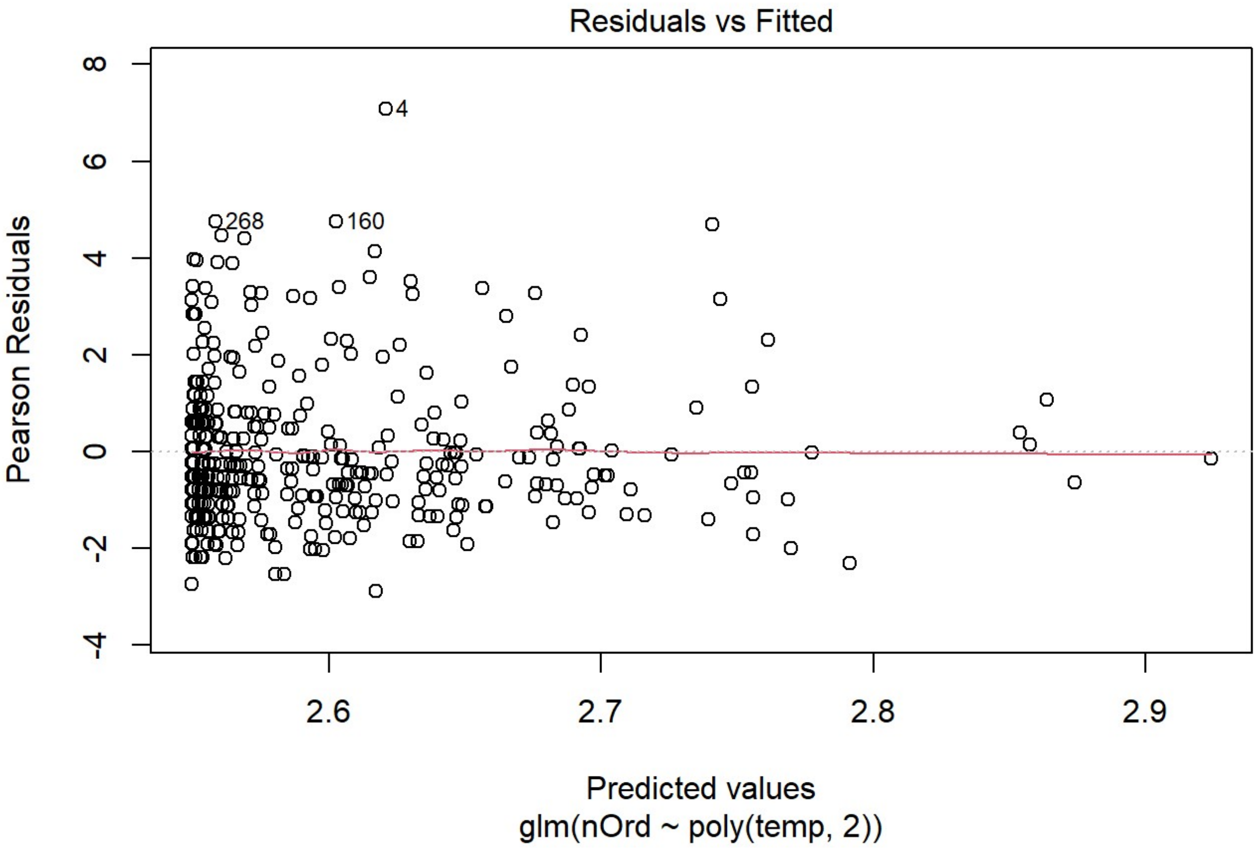


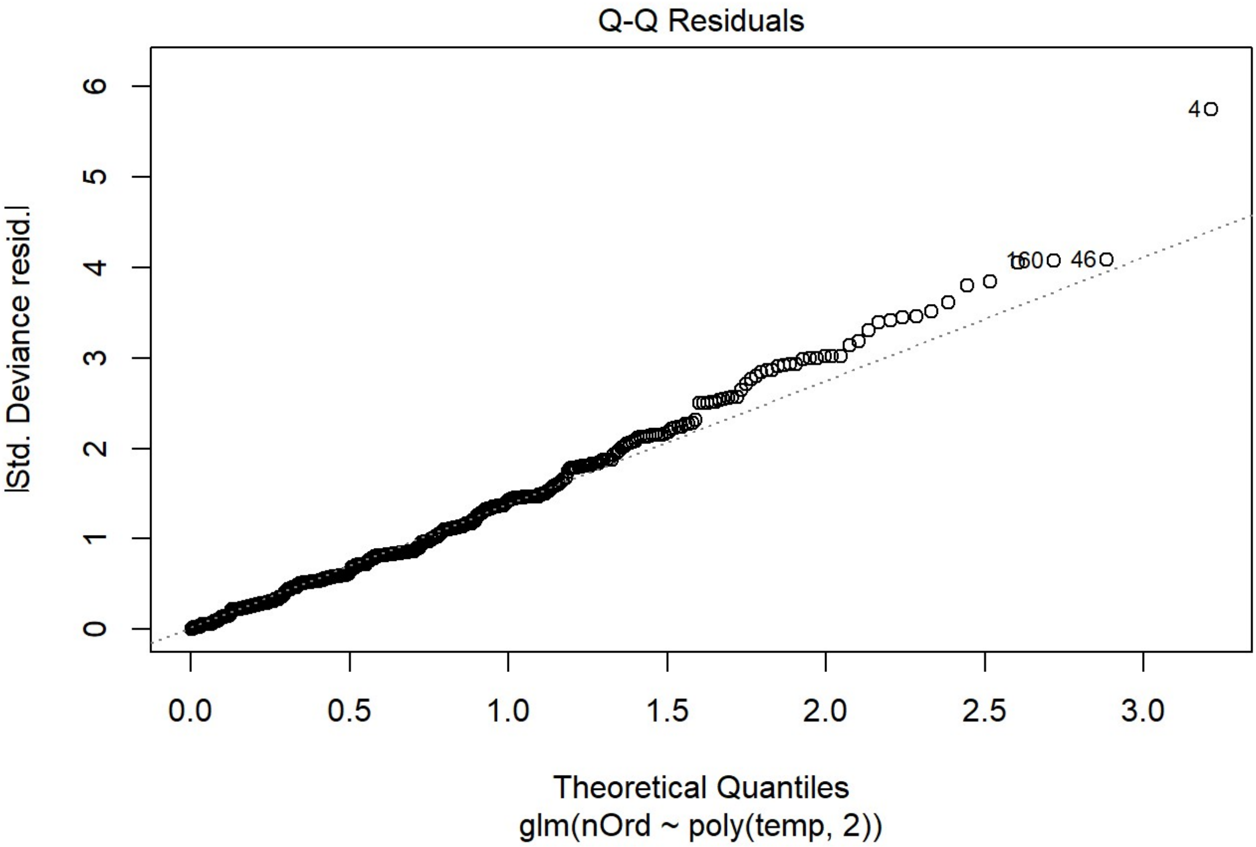
la temperatura è blandamente significativa, i punti hanno una forma vagamente parabolica, conviene quindi indagare i polinomi di ordine superiore all'1

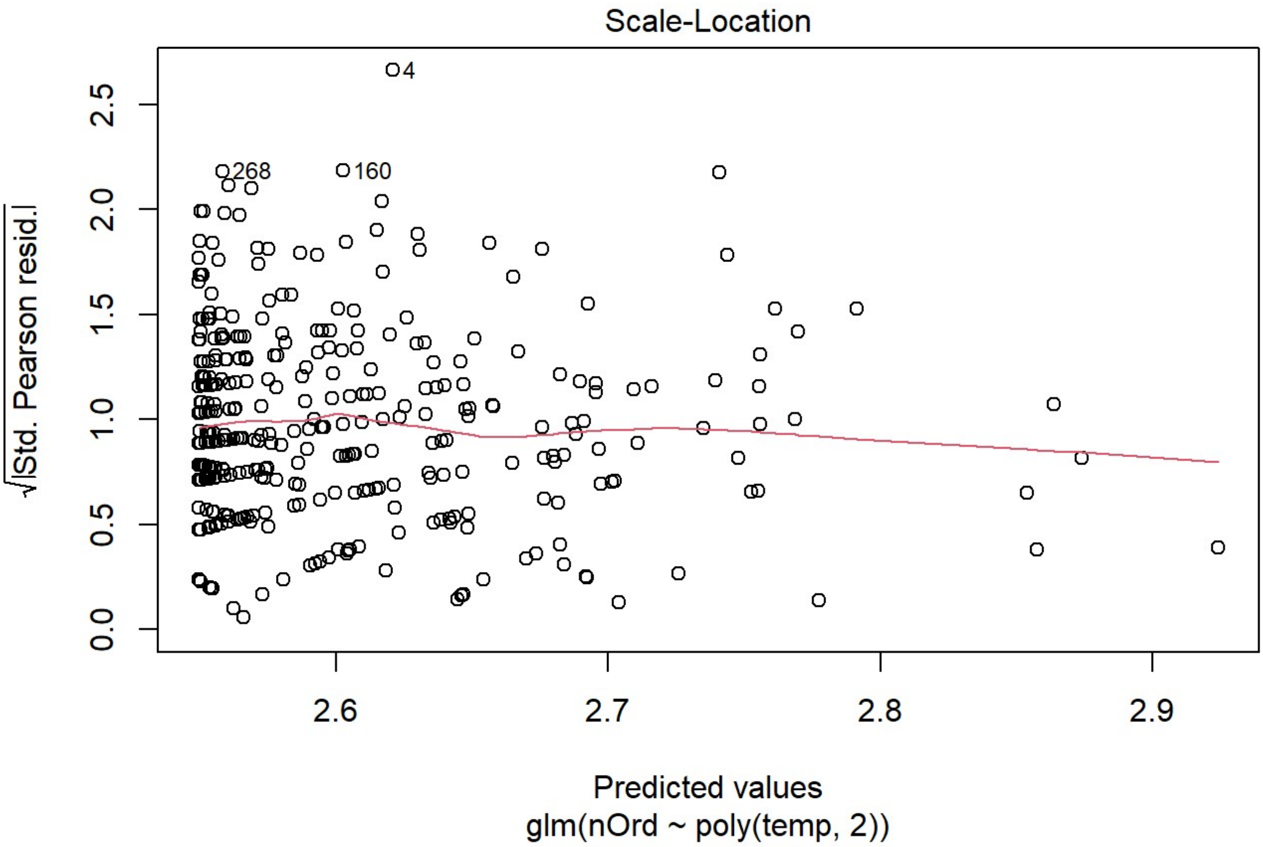
```
fit2 <- glm(nOrd~poly(temp, 2), data = dex2, family = "poisson")
summary(fit2)
```

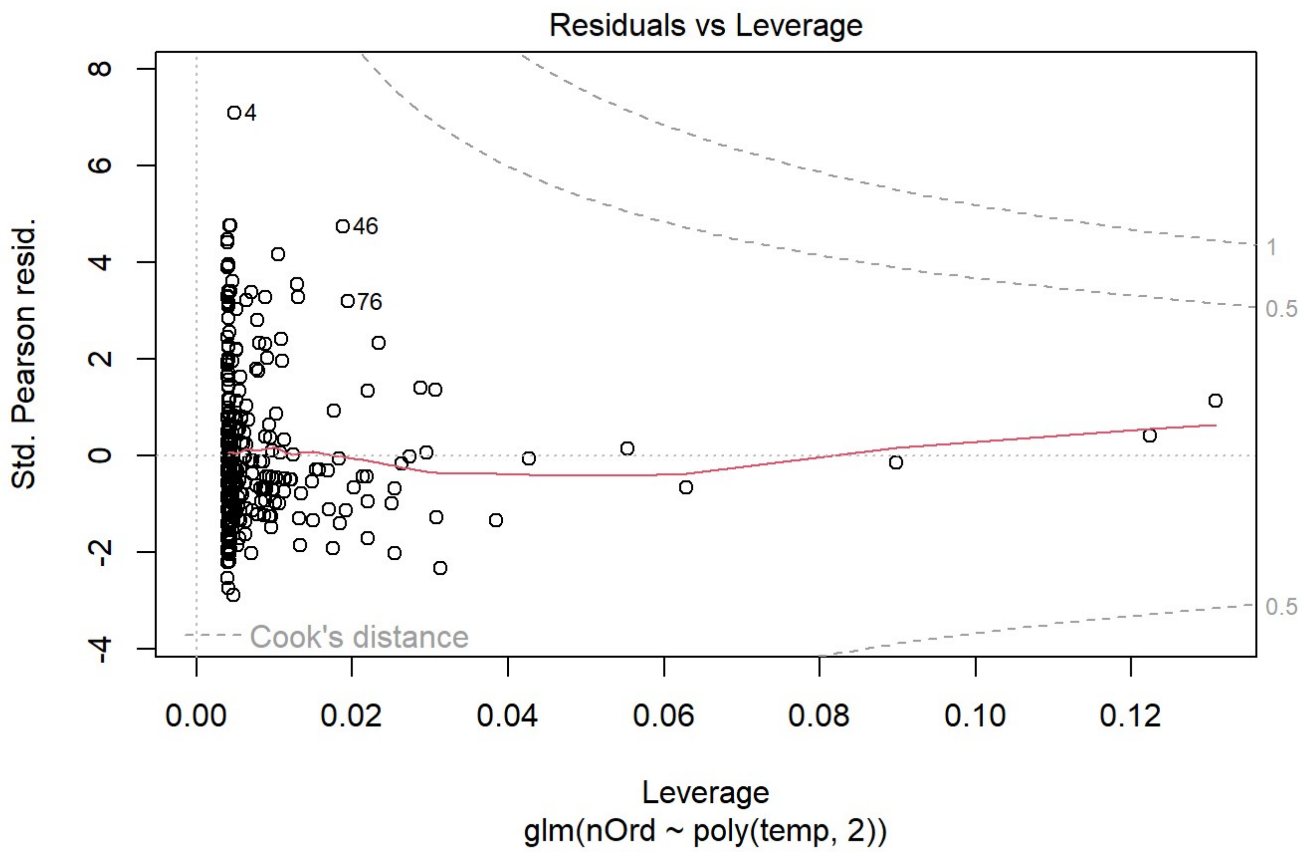
```
##
## Call:
## glm(formula = nOrd ~ poly(temp, 2), family = "poisson", data = dex2)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)      2.5987     0.0140 185.583   <2e-16 ***
## poly(temp, 2)1  -0.4582     0.2625  -1.745   0.0809 .
## poly(temp, 2)2   1.1327     0.2564   4.417    1e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 838.53  on 379  degrees of freedom
## Residual deviance: 816.69  on 377  degrees of freedom
## AIC: 2483.5
##
## Number of Fisher Scoring iterations: 4
```

```r
plot(fit2)
```


Residuals vs Fitted
Pearson Residuals
Predicted values
glm(nOrd ~ poly(temp, 2))
4
268
160


Q-Q Residuals
|Std. Deviance resid.|
Theoretical Quantiles
glm(nOrd ~ poly(temp, 2))
4
160
46





È significativo il monomio del secondo ordine. polinomi di ordine superiore non risultano necessari per predire nOrd

Es. 2.b

Usando il modello migliore che si è scelto al punto a) si verifichi se, a parità di temperatura, vi è una qualche differenza nel numero di ordini effettuati la domenica o nelle altre giornate.

```
fit_dom <- glm(nOrd ~ poly(temp, 2) + domenica, data = dex2, family = poisson)
summary(fit_dom)
```

```
##
## Call:
## glm(formula = nOrd ~ poly(temp, 2) + domenica, family = poisson,
##     data = dex2)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      2.44734    0.01653 148.081  < 2e-16 ***
## poly(temp, 2)1 -0.52848    0.26261  -2.012   0.0442 *
## poly(temp, 2)2  1.05599    0.25928   4.073 4.65e-05 ***
## domenica1        0.68594    0.03103  22.109  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 838.53  on 379  degrees of freedom
## Residual deviance: 378.93  on 376  degrees of freedom
## AIC: 2047.8
##
## Number of Fisher Scoring iterations: 4
```

il predittore Domenica è molto significativo, la correlazione è positiva. Anche considerando il criterio AIC, il nuovo modello è decisamente migliore del secondo

Es. 2.c

Usando il modello che si ritiene migliore si produca una stima del numero medio di ordini attesi per le giornate nel dataset `nd`. Si produca anche una stima intervallare usando un livello di confidenza pari al 98%.

```
nd <- data.frame(temp = c(16, 16, 26, 26), domenica = factor(c(0,1,0,1)))
rownames(nd) <- c("g16","d16","g26","d26")

pred <- predict(fit_dom, newdata = nd, type = "link", se.fit = TRUE)
z <- qnorm(1 - 0.02/2)  # circa 2.33

nd$fit    <- exp(pred$fit)                        # stima attesa
nd$lower  <- exp(pred$fit - z * pred$se.fit)     # limite inferiore CI
nd$upper  <- exp(pred$fit + z * pred$se.fit)     # limite superiore CI

nd
```

```
##     temp domenica      fit    lower    upper
## g16   16        0 11.08720 10.58708 11.61095
## d16   16        1 22.01508 20.59444 23.53372
## g26   26        0 11.51832 10.82841 12.25218
## d26   26        1 22.87112 21.17462 24.70354
```

Es. 2.d

Quale è la funzione legame usata quando si usa la funzione legame canonica per la distribuzione Poisson? Per il modello utilizzato al punto c) si provi ad usare la funzione legame radice quadrata (`link = sqrt`) e si verifichi se i valori puntuali stimati del numero di ordini differiscono quando si usa una diversa funzione legame.

la funzione legame canonica per la distribuzione poisson è la logaritmica.

```
fit_rad <- glm(nOrd ~ poly(temp, 2) + domenica, data = dex2, family = poisson(), link = "s
qrt")
```

```
## Error in glm.control(link = "sqrt"): unused argument (link = "sqrt")
```