*UNIVERSITÀ DEGLI STUDI DI MILANO*

*FACOLTÀ DI MATEMATICA*

# PROGETTO PER L'ESAME DI CPSM 2

*Analisi di dati e inferenza statistica*

Files: chicago_bikes.csv *(dataset)*
ChicagoBikes.R *(R script)*
*funzioni.R (R Script)*

Fabio Caironi

14/02/2019

# PROGETTO PER L'ESAME DI CPSM 2

## INTRODUZIONE

Si è deciso di analizzare dati riguardanti i noleggi di biciclette urbane pubbliche nella città di Chicago. Il dataset utilizzato è stato scaricato da https://www.kaggle.com/ ed ha il nome di chicago_bikes.csv: è un file di tipo Commar Separated Values contentente i seguenti campi:

year, month, week, hour, is_weekend, mean_temperature, median_temperature, tstorms, unknown, cloudy, rain_or_snow, not_clear, clear, rentals

I primi quattro campi (year, month, week, hour) contengono l'informazione temporale del noleggio di biciclette: più precisamente, ogni riga del dataset ha un'ora diversa della giornata x, della settimana y e del mese z e non ci sono ripetizioni.

Per ogni ora della giornata, ovvero per ogni riga, nel campo rentals è riportato il numero di noleggi effettuati in quell'ora dagli utenti del serivizio di bike-sharing pubblico.

I campi is_weekend, tstorms, unknown, cloudy, rain_or_snow, not_clear, clear contengono campioni di variabili dicotomiche; esse valgono 1 se, rispettivamente, quell'osservazione cade del weekend, è stata rilevata in presenza di temporali, di meteo sconosciuto, di tempo nuvoloso, di pioggia o neve, di cielo non sereno e di cielo sereno, 0 altrimenti.

Infine, i campi mean_temperature e median_temperature contengono le temperature medie e mediane dell'ora in cui è stata registrata ogni osservazione.

## LAVORO E IMPLEMENTAZIONE

Per prima cosa sono stati installati ed applicati i pacchetti che si utilizzeranno a seguire nel progetto:

"weathermetrics", "car", "nortest", "goftest", "ggplot2", "labstatR", "boot", "asympTest", "plotrix"

Si è importato poi il dataset in questione nel workspace di R e si è cominciata ad analizzare la struttura dei dati. Ci sono 34617 osservazioni, distribuite su 4 anni (2014 – 2017) e ciascuna di esse ha valori definiti su tutti i campi, ovvero non ci sono NA. Gli unici casi di mancanza di informazioni sono quelli in cui unknown == 1 (meteo sconosciuto), ma sono solo due osservazioni.

Si è rilevato che le colonne relative al meteo (tstorms, unknown, cloudy, rain_or_snow, not_clear, clear) sono mutuamente esclusive, ovvero per ogni osservazione una e una sola colonna tra queste ha valore 1 (il che non è scontato, perché, per esempio, si potrebbe pensare che possa valere rain_or_snow == 1 e not_clear == 1 per una certa osservazione). Si è deciso allora di assegnare ad ogni osservazione una stringa contenente la condizione meteorologica corrispondente, tramite l'introduzione di una nuova colonna di nome "weather".

Si è osservato che le colonne mean_temperature e median_temperature contengono le stesse informazioni: evidentemente, è stata registrata solo una temperatura per ora, per cui media e mediana delle temperature in quell'ora coincidono. Si decide così di

eliminare il campo mediana. Successivamente, si convertono le temperature di mean_temperature da Fahrenheit a Celsius, mediante la funzione fahrenheit.to.celsius del pacchetto {weathermetrics}.

Si è infine notato che i dati forniti non completano tutte le ore dei quattro anni: ci sono alcuni orari, nonché righe, mancanti. Guardando meglio le colonne delle osservazioni adiacenti alle ore mancanti, si può dedurre il criterio con il quale queste siano state omesse: la temperatura in quelle ore è tipicamente molto bassa (fino a -15°F = -26°C !) o si tratta di ore notturne, quindi si può dedurre che siano state omesse perché con numero di noleggi pari a 0. Difatti, non sono presenti nel dataset osservazioni con rentals == 0. Allora, tramite diversi cicli for, si scandiscono le ore mancanti e si completa il foglio di dati con queste righe, dove è stato assegnato "0" alla colonna rentals e NA a tutte le altre colonne, di cui non abbiamo informazioni, escluse quelle dell'indicazione temporale. `chicago_bikes` ha ora lunghezza 34992 (sono state aggiunte 305 oss.).
Il completamento dei dati è stato effettuato perché può portare a risultati più precisi nello studio delle distribuzioni. Tuttavia ciò non avrà effetto sullo studio della dipendenza lineare dei noleggi dalla temperatura media, perché essa è NA nelle osservazioni aggiunte.

Sistemati così i dati, si sono fattorizzate opportunamente tutte le colonne eccetto `mean_temperature` e `rentals` e si sono ordinate le righe cronologicamente.

# ANALISI DEI DATI

## Analisi 1 – Distribuzione dei noleggi

Si prende in considerazione la variabile aleatoria X che conta i noleggi in una generica ora h e se ne vuole studiare la distribuzione sulla base del campione dato. Si noti che in questa prima analisi si esclude il parametro ore, o detto in altri termini si considera la distribuzione complessiva dei noleggi supponendo che l'ora specifica non influenzi la probabilità di avere un numero n di noleggi, ovvero le variabili $X_{1, ..., X_{34992}}$ del campione siano identicamente distribuite. Ciò verrà poi smentito dalla seconda analisi, ma l'indagine in questione non perde significato dal momento che, avendo completato il dataset con tutte le ore mancanti, ogni orario (0-23) ha la stessa probabilità di essere estratto (p = 1/1458). Inoltre, si suppone ragionevolmente che le v.a. del campione siano indipendenti, ovvero che i noleggi di un'ora non influenzino i noleggi di un'altra.

Si procede allora con la creazione di un istogramma delle frequenze relative dei noleggi (**Figura 1**), a cui si sovrappone la curva di densità empirica calcolata con il comando

```
density(chicago_bikes$rentals, from = 0, to = 2000)
```

Si osserva che la curva ha un andamento simil-esponenziale: sovrapponiamo per un confronto la densità di una v.a. esponenziale di parametro λ pari a:

```
1/mean(chicago_bikes$rentals)
```

A questo punto notiamo una certa affinità tra le due curve ma solo a partire da un certo numero di noleggi in poi (c.a 400). Prima di quel numero, la densità empirica ha una convessità maggiore. Difatti, due test asintotici per il confronto di distribuzioni danno come esito il rigetto dell'ipotesi nulla (a livello α = 0.05) di uguaglianza della ecdf del campione con la cdf di un'esponenziale con parametro λ come sopra.

Test 1: ad.test dal pacchetto {goftest}

```
Anderson-Darling test of goodness-of-fit
        Null hypothesis: exponential distribution
        with parameter rate = 0.0036852168482402

data:  chicago_bikes$rentals
An = Inf, p-value = 1.715e-08
```

Test 2: ks.test dal pacchetto {stats}

```
One-sample Kolmogorov-Smirnov test

data:  abs(chicago_bikes$rentals + runif(n = n, min = -0.5, max =
0.5))
D = 0.13286, p-value < 2.2e-16
alternative hypothesis: two-sided
```

Sono stati utilizzati questi test, Anderson-Darling e Kolmogorov-Smirnov, perché siamo in presenza di un grande campione, anche se nel nostro caso esso è estratto da una v.a. discreta e non continua. Per il test di K-S sono stati perturbati leggermente i dati per evitare la presenza di ties.
Concludiamo che non c'è evidenza statistica che i noleggi seguano una distribuzione esponenziale.

Prima di passare alla prossima analisi, però, ci si chiede se si possono estrarre gruppi di osservazioni per i quali il test non-parametrico precedente abbia esito positivo. L'idea è quella di selezionare dati per cui la concavità della densità empirica nei punti < 400 sia maggiore. Si prova allora a cercare un anno e un mese in cui le occorrenze di noleggi inferiori a 400 siano maggiori. A tal fine si usa la funzione getmode:

```
getmode(subset(chicago_bikes, rentals < 400)$year)
```

```
                                # esito = "2014"

getmode(subset(chicago_bikes, rentals < 400 & year == 2014)$month)
                                # esito = "3"
```

Plottando così la densità empirica del sotto-campione relativo a Marzo del 2014 (**Figura 2**) si riscontra effettivamente a livello grafico una maggiore vicinanza alla densità esponenziale. Tuttavia, eseguendo nuovamente i test precedenti, non si ottengono ancora p-value sufficientemente alti (`8.065e-07` per A-D e `4.332e-08` per K-S) per accettare l'ipotesi nulla, però essi sono cresciuti considerevolmente, come conseguenza della diminuzione delle statistiche test che misurano la distanza dei grafici delle due cumulative (un visualizzazione grafica analoga avrebbe potuto essere appunto il plot delle due cumulative sovrapposte).

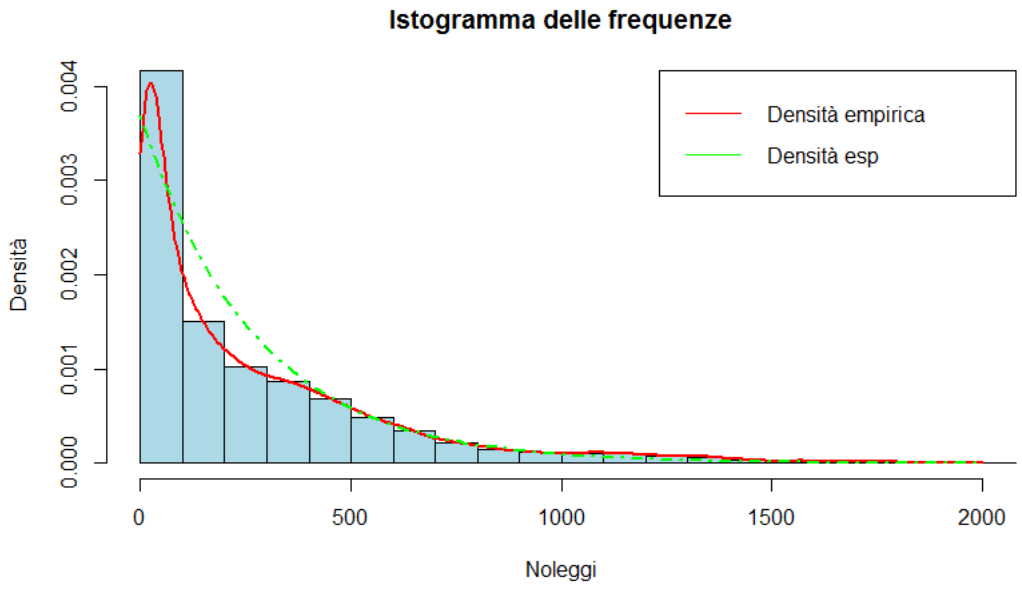


**Figura 1**

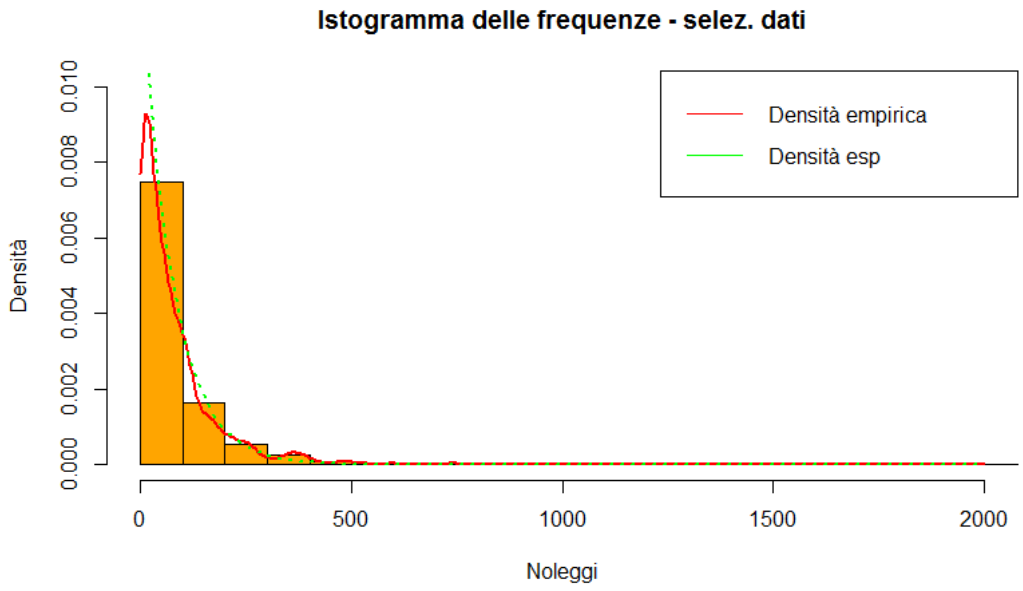


**Figura 2**

# Analisi 2 – Distribuzione dei noleggi per ora

Come seconda indagine, si prende in considerazione il parametro 'ora' e si vuole studiare, per ognuna di esse, caratteristiche significative della distribuzione dei noleggi corrispondenti. Pare infatti sensato che, trattandosi di dati relativi a comportamenti umani, ci sia una qualche dipendenza delle distribuzioni dei noleggi dall'ora in cui vengono effettuati. Per esempio, a priori si può immaginare che nelle ore notturne ci siano in media meno noleggi che nelle ore del pomeriggio. È proprio sulle medie che si farà inferenza.

Si divide il dataset in 24 sotto-campioni divisi per ora, da 0 a 23 e per ciascuno di essi si estrae la colonna dei noleggi. Si calcolano dunque le medie di noleggi per ogni ora e le si plottano con un grafico a barre (**Figura 3**). Sono evidenti dei picchi nei noleggi medi alle ore 8 e 17, mentre tra questi due orari le medie sono più vicine alla media totale, che risulta essere 271.35; inoltre, come ci i aspettava, i noleggi nelle ore notturne calano visibilmente.

Per confermare queste affermazioni, si costruiscono innanzi tutto degli intervalli di confidenza per la media dei noleggi di ciascuna ora, mediante il metodo asintotico che utilizza la statistica:

$$\frac{\bar{X} - \mu}{\sqrt{\frac{S_n^2}{n}}} \sim AN(0,1)$$

Tale metodo è implementato nella funzione asymp.test dell'omonimo pacchetto. È lecito utilizzare approssimazioni asintotiche dal momento che siamo in presenza di grandi campioni (taglia 1458). Utilizzando le funzioni di disegno grafico del pacchetto {ggplot2} si plottano gli IC così trovati e, solo osservando il grafico, si è certi (a livello $\alpha = 0.05$) del fatto che le medie di noleggi divisi per ora non sono tra loro tutte uguali: infatti l'intersezione tra tutti gli intervalli di confidenza è vuota.

Si può notare anche che gli intervalli di confidenza non hanno tutti la stessa ampiezza: per esempio, nelle ore di punta (8 e 17), gli IC sembrano avere ampiezza maggiore a tutti gli altri. Mostriamo che ciò è dovuto ad una maggiore varianza dei noleggi in questi due gruppi. Ricordiamo infatti che l'ampiezza di un IC asintotico bilaterale per la media di livello $1 - \alpha = 0.95$ risulta essere:

$$\left|\left[\bar{X} - N_{1-\frac{\alpha}{2}}^{(0,1)}\sqrt{\frac{S_n^2}{n}}, \bar{X} + N_{1-\frac{\alpha}{2}}^{(0,1)}\sqrt{\frac{S_n^2}{n}}\right]\right| = 2N_{1-\frac{\alpha}{2}}^{(0,1)}\sqrt{\frac{S_n^2}{n}}$$

quantità che è proporzionale alla varianza campionaria $S_n^2$.

Si eseguono allora 22 test per l'ora 8 e 22 test per l'ora 17, ciascuno dei quali per verificare l'uguaglianza delle varianze dei noleggi all'ora i con le varianze dei noleggi all'ora j, per

$$i = 8, 17 \text{ e } j = 1, \dots, 7, 9, \dots 16, 18, \dots, 23$$

contro l'ipotesi alternativa che la varianza del primo gruppo (relativo ad i) sia maggiore di quella del secondo gruppo (relativo a j). Il comando utilizzato per ciascun test è:

```
print(asymp.test(subset(chicago_bikes, hour == j)$rentals,
                 subset(chicago_bikes, hour == i)$rentals,
                 parameter = "dVar", alternative = "greater"))
```

Si riscontrano effettivamente p-values molto bassi (< 2.2e-16), per cui si conclude che negli orari di punta c'è una maggiore varianza dei noleggi.

I risultati ottenuti si possono interpretare nel seguente modo: i cittadini effettuano numerosi noleggi nelle ore di punta dei giorni lavorativi, probabilmente per recarsi ai luoghi di lavoro/studio, e questi noleggi hanno un peso consistente sia sulla media totale sia su quella dei singoli orari di punta (8 e 17). Infine, poiché nei giorni non lavorativi non si registrano tali picchi, risulta in generale alta la variabilità dei noleggi negli orari di punta.

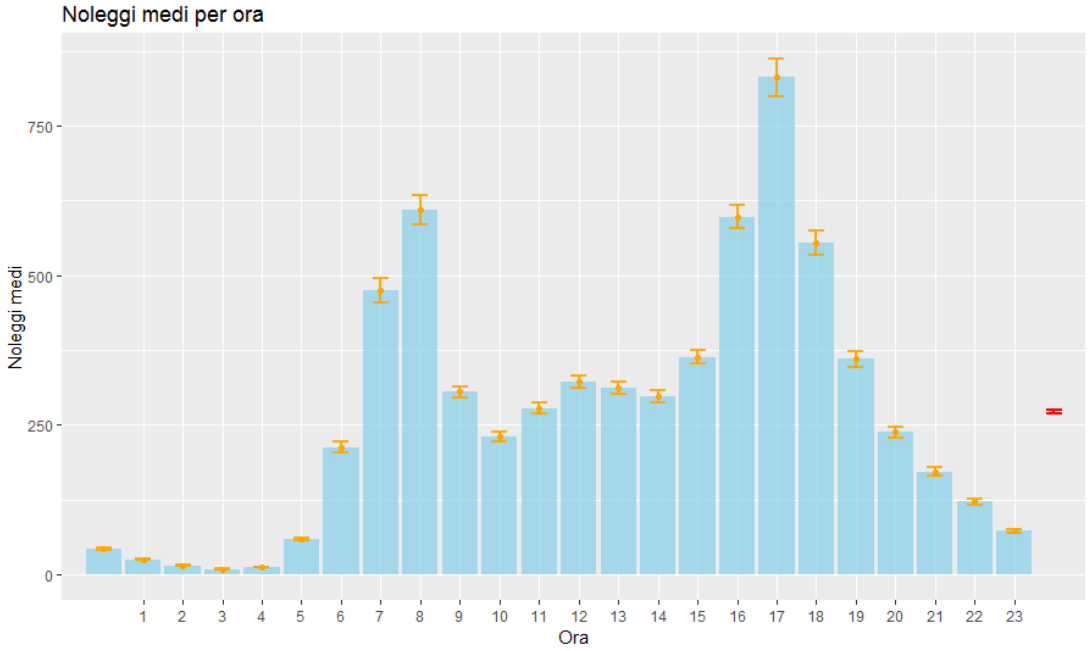


**Figura 3** — intervalli di confidenza per il valor medio di noleggi di ciascun ora; intervallo di confidenza per il valor medio di noleggi complessivo.

# Analisi 3 – Modello lineare tra noleggi e condizioni climatiche

In questa terza e ultima parte si studierà la dipendenza lineare del numero dei noleggi dalla temperatura. Premettiamo che le temperature registrate si distribuiscono su un range di

(-26.11, 35.0) °C

e si concentrano tra 2°C e 20°C, che sono rispettivamente il primo e il terzo quartile, come si vede con un boxplot, oppure con il comando

```
summary(chicago_bikes$mean_temperature)
Min.    1st Qu.  Median    Mean 3rd Qu.    Max.
-26.11    2.22   11.72   10.67   20.61   35.00
```

Cominciamo con il modello lineare semplice:

```
result <- lm(rentals ~ mean_temperature, data = chicago_bikes)
summary(result)
```

Esito:

```
Residuals:
    Min      1Q  Median      3Q     Max
-485.25 -171.88  -49.24   87.67 1914.83

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      129.3123     2.1167   61.09   <2e-16 ***
mean_temperature  13.5829     0.1349  100.68   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 288.6 on 34615 degrees of freedom
Multiple R-squared:  0.2265,   Adjusted R-squared:  0.2265
F-statistic: 1.014e+04 on 1 and 34615 DF,  p-value: < 2.2e-16
```

Si osserva subito che la regressione sembra essere molto significativa (p-values bassi per tutti i tests), tuttavia dai quantili dei residui possiamo vedere che la loro distribuzione è poco simmetrica. Con ciò potrebbe non essere soddisfatta l'ipotesi di normalità degli errori. Si procede comunque con la nostra analisi sul modello, per poi tornare sui dati e applicare una trasformazione per rendere normali i residui.

Si vuole espandere il modello esistente ad un modello che tenga in considerazione anche le variabili dicotomiche

tstorms, cloudy, clear, not_clear, rain_or_snow, rush_hour

ovvero, sostanzialmente, che preveda una traslazione, per ogni gruppo tra i sopraccitati, della retta di regressione già esistente. I primi cinque sono, come già visto, parametri relativi al meteo. La colonna rush_hour, invece, è stata aggiunta selezionando le ore 8 e 17, che, dall'analisi precedente, sono risultate le ore di punta dei noleggi. Quindi, in una generica osservazione, la colonna rush_hour vale 1 se e solo se l'ora è 8 o 17. Ciò è stato fatto perché si vuole andare a completare l'analisi precedente, mostrando che la disuguaglianza

$$\mathbb{E}[R_{rush\_hour}] > \mathbb{E}[R_{not\_rush\_hour}]$$

dove $R_{rush\_hour}$ conta il numero di noleggi nelle ore di punta e $R_{not\_rush\_hour}$ nelle ore non di punta, già dimostrata, è riscontrabile in un valore fortemente positivo del coefficiente relativo alla variabile dummy "rush_hour" nel modello di regressione completo.

Si procede dunque con lo studio di questo modello completo: si utilizza la funzione step del pacchetto {stats}, con direzione forward, che aggiunge di volta in volta variabili, tra quelle specificate, al modello, selezionando quelle per cui l'AIC è più basso.

```
step(result, scope = (rentals ~ mean_temperature + tstorms + cloudy
           + clear + not_clear + rain_or_snow + rush_hour),
           direction = 'forward')
```

Esito:

```
Call:
lm(formula = rentals ~ mean_temperature + rush_hour + cloudy +
    tstorms + not_clear + clear, data = chicago_bikes)

Coefficients:
     (Intercept)  mean_temperature        rush_hour1         cloudy1

           41.20             13.21            362.46           95.31

        tstorms1       not_clear1            clear1

        -98.75          62.60                 19.02
```

Il metodo AIC ha selezionato tutte le variabili proposte eccetto rain_or_snow. Il modello risultante è perciò:

$$R = \beta_0 + \beta_1 T + \delta_1 D_1 + \delta_2 D_2 + \delta_3 D_3 + \delta_4 D_4 + \delta_5 D_5$$

con:

- $R$ il numero di noleggi
- $T$ la temperatura
- $D_i,\ i = 1, \dots, 5$ le dummy variables rush_hour, cloudy, tstorms, not_clear e clear

Dai coefficienti individuati leggiamo importanti informazioni:

- Il numero dei noleggi cresce, secondo il modello, di $\beta_1 = 13.21$ al crescere della temperatura di 1°C
- Come previsto, il coefficiente $\delta_1 = 362.46$ relativo a rush_hour ha un valore molto alto
- In caso di temporali, il numero di noleggi è ridotto di $\delta_3 = -98.75$ , pur mantenendo una dipendenza lineare dalla temperatura.

Si plottano poi le rette (segmenti) di regressione utilizzando colori selettivi (**Figura 4**). Si sono evitati gli scatterplot dei dati a causa delle dimensioni estese dei campioni.

Infine, si vuole applicare una trasformazione ai dati di partenza, come si diceva prima, per ottenere possibilmente un modello in cui è soddisfatta l'ipotesi di normalità degli errori. Dato che i residui nel modello precedente si distribuiscono con un'ampia coda a destra, potremmo pensare di trasformare i dati di rentals con il logaritmo:

```
result <- lm(log(rentals) ~ mean_temperature, data = chicago_bikes)
summary(result)
```

E successivamente riapplichiamo il metodo step, ottenendo la stessa selezione delle variabili di prima. Questa volta, però, i residui sono effettivamente "più normali", come si evince dall'osservazione del grafico creato con la funzione qqPlot del pacchetto {car} (**Figura 5**).

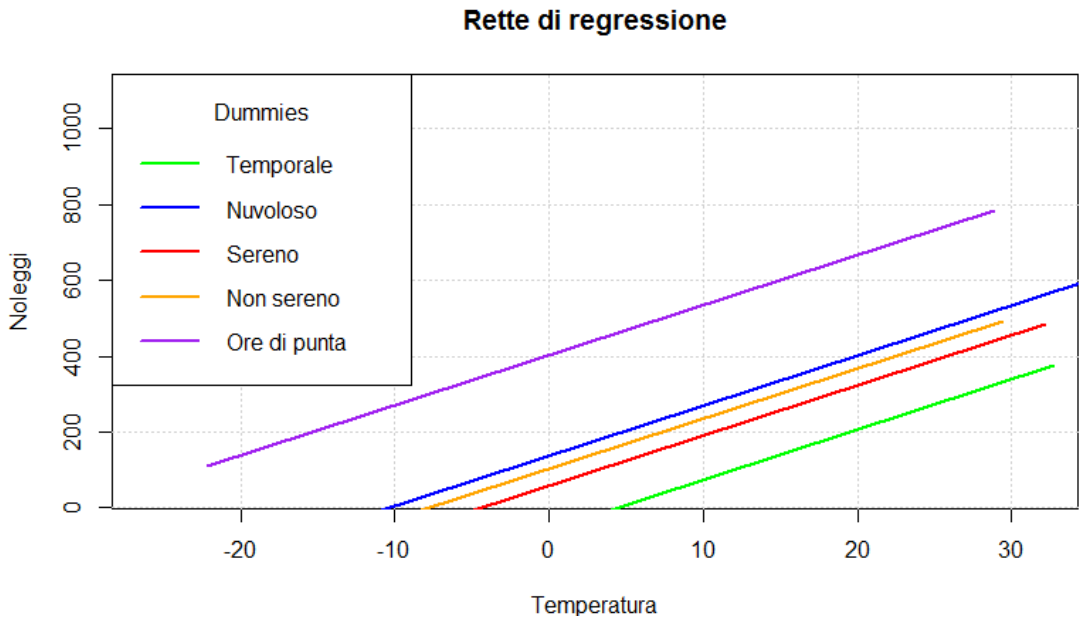


Figura 4

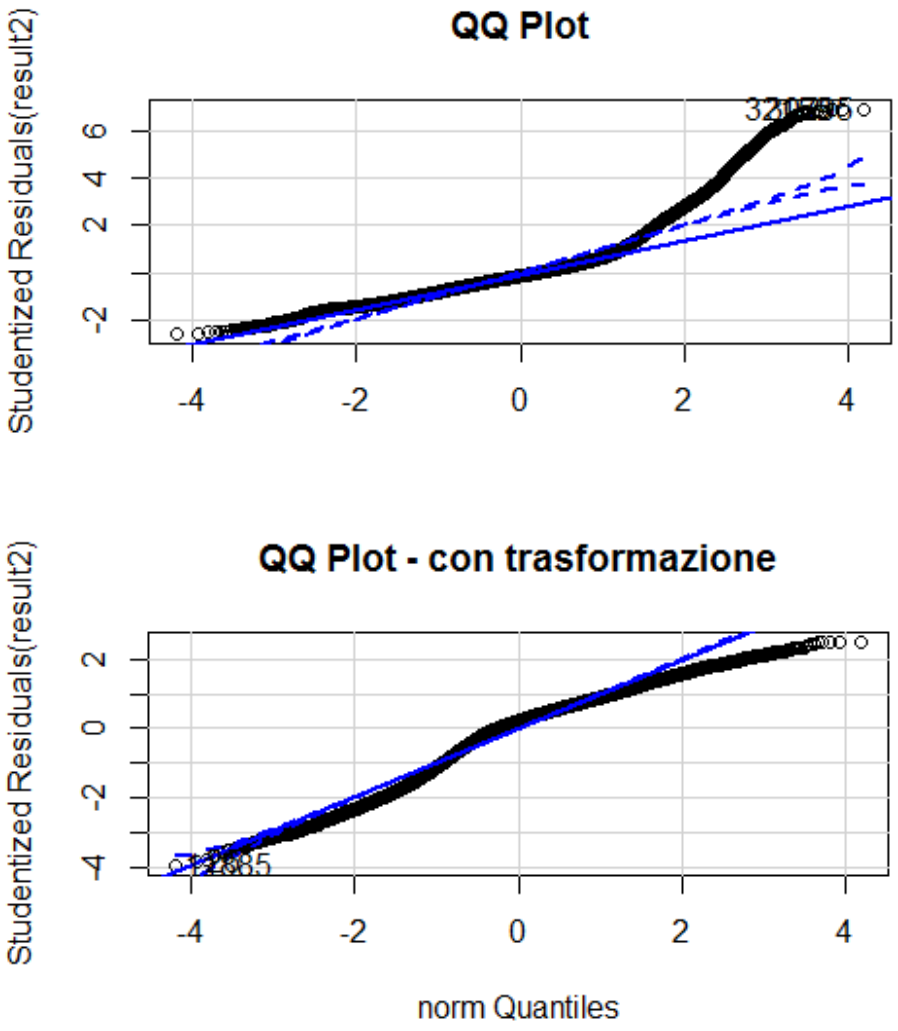


Figura 5