

R Notebook

- Master II livello “Data science and big data analytics” - Progetto finale del modulo *Statistica*
 - Report Analisi statistica Dataset “Buildings1”
 - Agostino Fontana
 - **Analisi del dataset Buildings.**
 - **Descrizione sommaria dei dati:**
 - **Il problema Statistico**
 - **Individuazione di eventuali outliers**
 - Stima della Correlazione tra le variabili
 - Analisi delle componenti Principali
 - Analisi Regressione Lineare
 - Validazione del modello lineare
 - Indagini statistiche varie.
 - Conclusioni
-

Master II livello “Data science and big data analytics” - Progetto finale del modulo *Statistica*

Report Analisi statistica Dataset “Buildings1”

Agostino Fontana

N.B. Installare i package relativi ed importare le librerie necessarie per l'esecuzione del codice R sottostante

```
library(MLANP)
```

```
## ***** MLANP Version 1.7.0 *****
```

```
## Loading required package: rpanel
```

```
## Loading required package: tcltk
```

```
## Package `rpanel', version 1.1-5: type help(rpanel) for summary information
```

```
## Loading required package: rgl
```

```
## Loading required package: ks
```

```
## Loading required package: lattice
```

```
## Warning: package 'lattice' was built under R version 4.3.1
```

```
## Loading required package: misc3d
```

```
## Loading required package: MASS
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: ellipse
```

```
##  
## Attaching package: 'ellipse'
```

```
## The following object is masked from 'package:graphics':  
##  
## pairs
```

```
library(summarytools)  
library(knitr)  
library(heatmap3)  
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(scatterplot3d)
library(heatmap3)
library(factoextra)
```

```
## Loading required package: ggplot2
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(ggplot2)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg      ggplot2
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##   last_plot
```

```
## The following object is masked from 'package:MASS':
##
##   select
```

```
## The following object is masked from 'package:stats':
##
##   filter
```

```
## The following object is masked from 'package:graphics':
##
##   layout
```

```
library(caret)
```

Analisi del dataset Buildings.

Il dataset "Buildings1" oggetto di studio è contenuto nel package **MLANP**, quindi si procede all'importazione del dataset nell'ambiente di lavoro ed all'analisi delle variabili e delle prime righe, per capire la struttura dei dati del dataset oggetto dell'analisi.

```
data("buildings1")
dati=buildings1
attach(dati)
```

```
head(dati)
```

```
##      timestamp air_temperature cloud_coverage dew_temperature precip_depth_1_hr
## 1:           0              3.8         1.672312             2.4         0.6784231
## 2:           0              3.8         1.672312             2.4         0.6784231
## 3:           0              3.8         1.672312             2.4         0.6784231
## 4:           0              3.8         1.672312             2.4         0.6784231
## 5:           0              3.8         1.672312             2.4         0.6784231
## 6:           0              3.8         1.672312             2.4         0.6784231
##      sea_level_pressure wind_direction wind_speed building_id meter meter_reading
## 1:          1020.9         240          3.1          105      0         23.3036
## 2:          1020.9         240          3.1          106      0          0.3746
## 3:          1020.9         240          3.1          107      0        175.1840
## 4:          1020.9         240          3.1          108      0         91.2653
## 5:          1020.9         240          3.1          109      0         80.9300
## 6:          1020.9         240          3.1          110      0         86.2283
##      primary_use year_built log.meter floor.count sq.feet day hour
## 1: Education    1960.921 3.1906245         6    50624  0  0
## 2: Education    1960.172 0.3181628         5     5375  0  0
## 3: Education    2005.000 5.1715289        11    97533  0  0
## 4: Education    1913.000 4.5246681         6     81581  0  0
## 5: Education    1953.000 4.4058652         7     56996  0  0
## 6: Education    2006.000 4.4685288         9     27815  0  0
```

```
booktabs = TRUE
```

```
dim(dati)
```

```
## [1] 512872      18
```

Il dataset è composto da 512.872 righe e 18 colonne, così strutturate :

```
str(dati)
```

```
## Classes 'data.table' and 'data.frame':  512872 obs. of  18 variables:
## $ timestamp      : int  0 0 0 0 0 0 0 0 0 0 ...
## $ air_temperature : num  3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 ...
## $ cloud_coverage  : num  1.67 1.67 1.67 1.67 1.67 ...
## $ dew_temperature : num  2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 ...
## $ precip_depth_1_hr : num  0.678 0.678 0.678 0.678 0.678 ...
## $ sea_level_pressure: num  1021 1021 1021 1021 1021 ...
## $ wind_direction  : num  240 240 240 240 240 240 240 240 240 ...
## $ wind_speed      : num  3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 ...
## $ building_id     : int  105 106 107 108 109 110 111 112 113 ...
## $ meter           : int  0 0 0 0 0 0 0 0 3 0 ...
## $ meter_reading    : num  23.304 0.375 175.184 91.265 80.93 ...
## $ primary_use      : chr  "Education" "Education" "Education" "Education" ...
## $ year_built       : num  1961 1960 2005 1913 1953 ...
## $ log.meter        : num  3.191 0.318 5.172 4.525 4.406 ...
## $ floor.count      : num  6 5 11 6 7 9 8 7 7 10 ...
## $ sq.feet          : num  50624 5375 97533 81581 56996 ...
## $ day             : num  0 0 0 0 0 0 0 0 0 ...
## $ hour            : num  0 0 0 0 0 0 0 0 0 ...
## - attr(*, ".internal.selfref")=<externalptr>
```

Descrizione sommaria dei dati:

Si tratta di un campione relativo all'osservazione dei consumi orari di energia elettrica in diversi edifici.

Ogni rilevazione, contiene diverse variabili (18) che possono influenzare il consumo di energia.

Ecco una breve spiegazione delle variabili presenti nei dati:

Dati e Variabili:

512.872 rilevazioni orarie.

1. **"timestamp"**: Rappresenta l'orario espresso in ore trascorse dal tempo 0 della prima osservazione. Questo può essere utile per analizzare le variazioni di consumo nel corso del tempo.

Variabili atmosferiche:

2. **"air_temperature"**: La temperatura dell'aria nell'area circostante all'edificio.
3. **"cloud_coverage"**: La copertura nuvolosa.
4. **"dew_temperature"**: La temperatura di rugiada.
5. **"precip_depth_1_hr"**: La profondità delle precipitazioni nell'ultima ora.
6. **"sea_level_pressure"**: La pressione atmosferica al livello del mare.
7. **"wind_direction"**: La direzione del vento.
8. **"wind_speed"**: La velocità del vento.

Variabili caratteristiche dell'immobile

9. **"building_id"**: L'identificativo univoco dell'edificio in cui è stata effettuata la rilevazione.
10. **"meter"**: Indica il tipo di consumo di energia, con due tipi possibili: 0 e 3.
11. **"meter_reading"**: Rappresenta il consumo orario di energia elettrica.
12. **"primary_use"**: Specifica l'uso principale dell'edificio (ad esempio: commerciale, residenziale, industriale, ecc.).
13. **"year_built"**: L'anno in cui l'edificio è stato costruito.
14. **"log.meter"**: Il logaritmo naturale del consumo di energia (può essere utilizzato al posto dell'originale "meter_reading" per analisi statistiche).
15. **"floor.count"**: Il numero di piani dell'edificio.
16. **"sq.feet"**: La superficie dell'edificio espressa in piedi quadrati.
17. **"hour"**: L'ora della rilevazione.
18. **"day"**: Il giorno della rilevazione.

Nella tabella a seguire si osserva un riepilogo delle statistiche descrittive di base per ciascuna variabile.


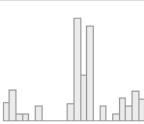
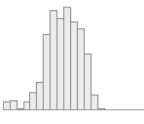
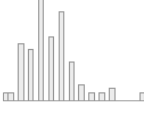
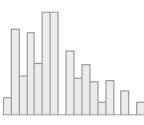
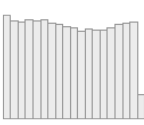

```
print(dfSummary(dati,
                varnumbers = FALSE,
                valid.col   = FALSE,
                graph.magnif = 0.90),
      max.tbl.height = 600,
      method = 'render'
)
```

Data Frame Summary

dati

Dimensions: 512872 x 18

Duplicates: 0

	4. Office 5. Public services	130174 (25.3%) 26022 (5.1%)		
year_built [numeric]	Mean (sd) : 1962.2 (28.5) min ≤ med ≤ max: 1900 ≤ 1960.7 ≤ 2007 IQR (CV) : 18.7 (0)	38 distinct values		0 (0.0%)
log.meter [numeric]	Mean (sd) : 4.4 (1.3) min ≤ med ≤ max: 0 ≤ 4.5 ≤ 10 IQR (CV) : 1.8 (0.3)	66129 distinct values		0 (0.0%)
floor.count [numeric]	Mean (sd) : 8 (2.5) min ≤ med ≤ max: 3 ≤ 8 ≤ 17 IQR (CV) : 3 (0.3)	13 distinct values		0 (0.0%)
sq.feet [numeric]	Mean (sd) : 66514 (40647.1) min ≤ med ≤ max: 5375 ≤ 57674 ≤ 174602 IQR (CV) : 56694 (0.6)	51 distinct values		0 (0.0%)
day [numeric]	Mean (sd) : 180.2 (106.7) min ≤ med ≤ max: 0 ≤ 177 ≤ 365 IQR (CV) : 187 (0.6)	366 distinct values		0 (0.0%)
hour [numeric]	Mean (sd) : 11.5 (6.9)			0

Generated by summarytools (<https://github.com/dcomtois/summarytools>) 1.0.1 (R (<https://www.r-project.org/>)
version 4.3.0)
2023-10-02

Il problema Statistico

Il problema statistico che si vuole affrontare riguarda la comprensione di come la variabile “log.meter” dipende dalle altre variabili.

Ovvero si vuole indagare sul livello d'interdipendenza della variabile $y=log.meter$ dal resto delle altre variabili, quindi si eseguirà un'analisi di regressione lineare al fine di definire un modello di stima del consumo elettrico in funzione delle altre variabili note (predittori)



Considerando che la regressione lineare è sensibile agli outliers il che può portare ad un overfitting del modello, si cercherà quindi la combinazione ottimale di variabili ed in fine si stimerà la bontà del modello individuato intesa in termini di errore quadratico medio. L'RMSE misura l'errore medio delle previsioni del modello rispetto ai dati reali.

Per quanto riguarda gli outliers si valuterà se escluderli o normalizzarli.

1. Accertato che il dataset non contenga valori mancanti si passerà all'analisi delle variabili.

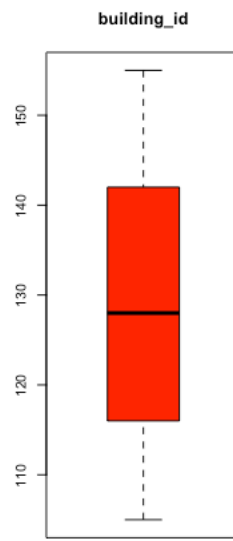
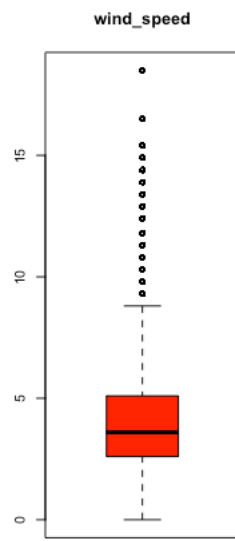
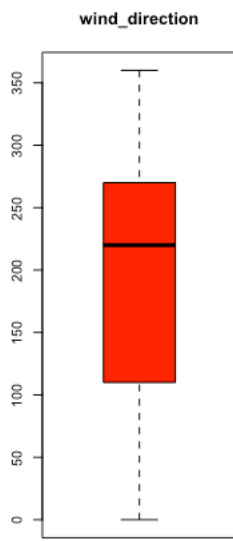
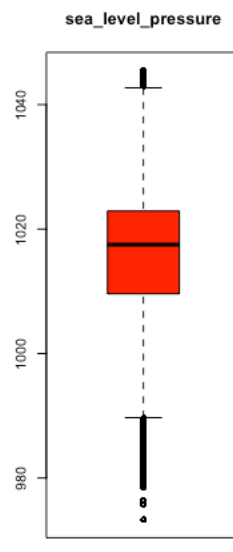
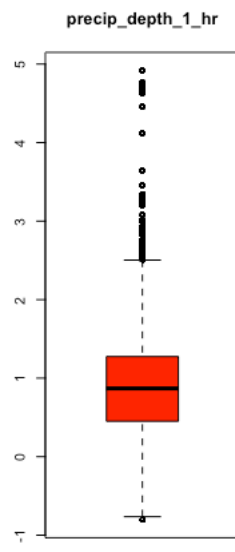
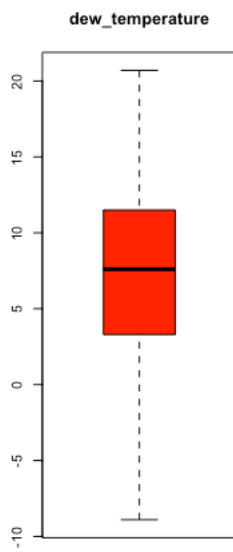
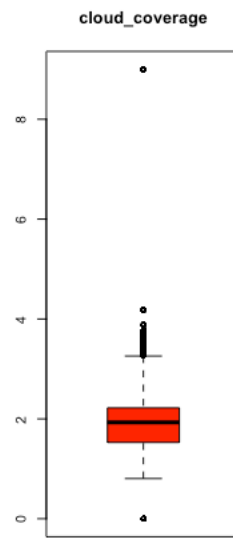
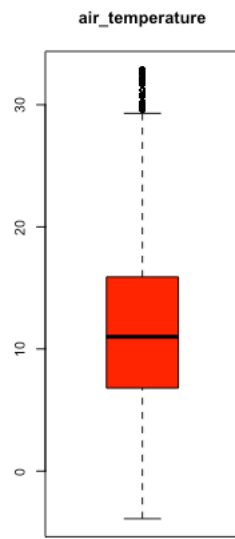
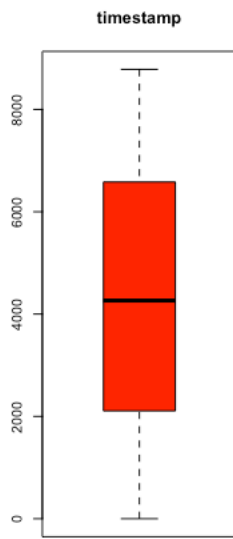
```
missing <- length(which(is.na(dati)==T))  
print(missing)
```

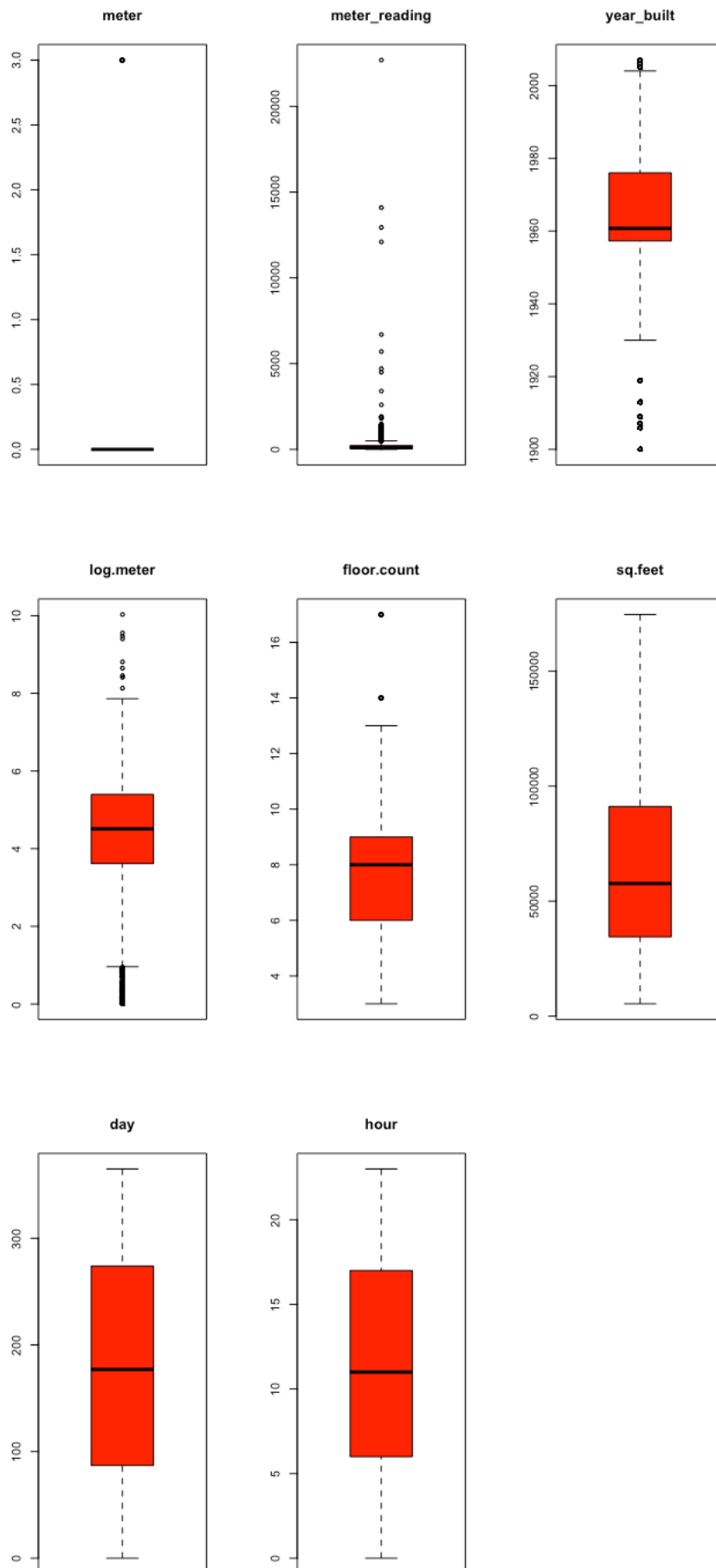
```
## [1] 0
```

Individuazione di eventuali outliers

Per verificare la presenza di outlier risulta utile analizzare la seguente rappresentazione.

```
options(repr.plot.width = 10, repr.plot.height = 5)  
par(mfrow = c(1, 3))  
  
# Creo un ciclo for per eseguire un boxplot di ogni variabile numerica.  
  
for (col_name in colnames(dati)) {  
  if (is.numeric(dati[[col_name]])) {  
    boxplot(dati[[col_name]],  
            main = col_name, col = "red")  
  }  
}
```





Si noti la presenza di outliers per le variabili : air_temperature, cloud_coverage, precip_depth_1_hr, sea_level_pressure, wind_speed, year_built, log.meter, pertanto è necessario tenerne conto nelle successive fasi di analisi.

Trasformazione di Variabili in fattore.

La variabile "**primary_use**" essendo un classificatore del edificio, è necessario trattarla da fattore, analizzeremo successivamente l'andamento dei consumi nel tempo per tipologia di immobile.

Le variabili ora del giorno e giorno della settimana sono entrambi tipi di dati categorici, non hanno un valore numerico intrinseco in quanto non ha senso dire che le 14:00 è “maggiore” delle 13:00 o che il mercoledì è “maggiore” del martedì pertanto, trattarli come fattori riflette meglio la loro natura. L'andamento delle temperature ed il consumo di energia elettrica può variare in modo regolare in base all'ora del giorno o al giorno della settimana, trattarle come fattori consente di catturare questo comportamento in modo più preciso rispetto a una rappresentazione numerica continua, si valuterà infine il consumo medio di energia per giorno dell'anno.

La variabile **meter** essendo un indicatore del tipo di consumo, anch'essa verrà trattata come fattore. Anche le variabili **year_built** (anno di costruzione) e **building_id** (identifica univocamente l'edificio) devono essere trattati come fattore, pertanto si effettuerà un'analisi dell'andamento dei consumi in relazione all'anno di costruzione dell'edificio.

```
dati$primary_use <- as.factor(dati$primary_use)
dati$hour<-as.factor(dati$hour)
dati$day<-as.factor(day%7)
dati$meter<-as.factor(meter)
dati$year_built<-as.factor(year_built)
dati$building_id<-as.factor(building_id)
```

A seguire si riporta la struttura del dataset.

```
str(dati)
```

```
## Classes 'data.table' and 'data.frame':  512872 obs. of  18 variables:
## $ timestamp      : int  0 0 0 0 0 0 0 0 0 0 ...
## $ air_temperature : num  3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 ...
## $ cloud_coverage  : num  1.67 1.67 1.67 1.67 1.67 ...
## $ dew_temperature : num  2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 ...
## $ precip_depth_1_hr : num  0.678 0.678 0.678 0.678 0.678 ...
## $ sea_level_pressure: num  1021 1021 1021 1021 1021 ...
## $ wind_direction  : num  240 240 240 240 240 240 240 240 240 240 ...
## $ wind_speed       : num  3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 ...
## $ building_id      : Factor w/ 51 levels "105","106","107",...: 1 2 3 4 5 6 7 8 9 ...
## $ meter            : Factor w/ 2 levels "0","3": 1 1 1 1 1 1 1 2 1 ...
## $ meter_reading     : num  23.304 0.375 175.184 91.265 80.93 ...
## $ primary_use      : Factor w/ 5 levels "Education","Entertainment/public assembly",...: 1 1 1 1 1 1 1 1 1 1 ...
##
## $ year_built       : Factor w/ 38 levels "1900","1906",...: 17 14 36 5 8 37 4 15 15 12 ...
## $ log.meter        : num  3.191 0.318 5.172 4.525 4.406 ...
## $ floor.count      : num  6 5 11 6 7 9 8 7 7 10 ...
## $ sq.feet          : num  50624 5375 97533 81581 56996 ...
## $ day              : Factor w/ 7 levels "0","1","2","3",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ hour             : Factor w/ 24 levels "0","1","2","3",...: 1 1 1 1 1 1 1 1 1 1 ...
## - attr(*, ".internal.selfref")=<externalptr>
```

Si effettua quindi un subset del dataframe per escludere : le variabili non numeriche, timestamp e meter_reading.

```
#datisub<-subset(dati,! (colnames(dati)%in%c("timestamp","building_id","primary_use","meter","meter_reading","day","hour")))
datisub<-subset(dati[,c(2:8,14:16)])
```

Stima della Correlazione tra le variabili

Si effettua adesso una stima delle correlazioni tra le variabili al fine di individuare le relazioni tra le variabili, in particolar modo si è interessati all'osservazione del grado di correlazione della variabile **log.meter** rispetto alle altre variabili.

```
matrice_correlazione <- round(cor(datisub),3)
kable(matrice_correlazione,
      format = "html",
      booktabs = TRUE,
      method = 'render'
)
```

	air_temperature	cloud_coverage	dew_temperature	precip_depth_1_hr	sea_level_pressure	wind_direction	wind_speed	log.meter	floor.count	sq.feet
air_temperature	1.000	0.198	0.815	0.237	0.011	0.035	0.171	-0.037	0.004	0.003
cloud_coverage	0.198	1.000	0.271	0.904	-0.347	0.145	0.451	0.002	0.000	-0.001
dew_temperature	0.815	0.271	1.000	0.281	-0.042	0.008	0.086	-0.062	0.005	0.003
precip_depth_1_hr	0.237	0.904	0.281	1.000	-0.252	0.206	0.747	0.006	0.000	0.000
sea_level_pressure	0.011	-0.347	-0.042	-0.252	1.000	-0.063	-0.333	-0.015	0.002	0.002
wind_direction	0.035	0.145	0.008	0.206	-0.063	1.000	0.091	-0.002	0.000	0.000
wind_speed	0.171	0.451	0.086	0.747	-0.333	0.091	1.000	0.026	-0.001	0.000
log.meter	-0.037	0.002	-0.062	0.006	-0.015	-0.002	0.026	1.000	0.290	0.546
floor.count	0.004	0.000	0.005	0.000	0.002	0.000	-0.001	0.290	1.000	0.553
sq.feet	0.003	-0.001	0.003	0.000	0.002	0.001	0.000	0.546	0.553	1.000

Dall'osservazione della matrice di correlazione notiamo che la nostra variabile **“log.meter”** è scarsamente correlata con il resto delle variabili eccetto che per le seguenti : **sq.feet ; floor.count**.

Mentre la variabile atmosferiche tra di loro risultano correlate nel seguente modo :

“air_temperature” è altamente correlata positivamente con **“dew_temperature”** (0.815) e moderatamente correlata positivamente con **“wind_speed”** (0.171).

La variabile **“cloud_coverage”** è altamente correlata positivamente con **“precip_depth_1_hr”** (0.904) e negativamente correlata con **“sea_level_pressure”** (-0.347).

La variabile **“dew_temperature”** è altamente correlata positivamente con **“air_temperature”** (0.815) e moderatamente correlata positivamente con **“wind_speed”** (0.086).

La variabile **"precip_depth_1_hr"** è altamente correlata positivamente con **"cloud_coverage"** (0.904) e moderatamente correlata positivamente con **"wind_speed"** (0.747).

La variabile **"sea_level_pressure"** è correlata negativamente con **"cloud_coverage"** (-0.347) e moderatamente correlata negativamente con **"wind_speed"** (-0.333). La variabile **"wind_direction"** ha una correlazione moderata positiva con **"wind_speed"** (0.091).

Le altre correlazioni mostrano le relazioni tra le variabili caratteristiche degli immobili ovvero **"year_built"**, **"log.meter"**, **"floor.count"**, e **"sq.feet"** rispetto alle altre variabili.

Nello specifico, considerando il focus oggetto di studio ovvero la comprensione di come la variabile **"log.meter"** dipende dalle altre variabili osserviamo le correlazioni in ordine decrescente di **"log.meter"** con le altre variabili :

- **"sq.feet"** con una correlazione di 0.546
- **"floor.count"** con una correlazione di 0.290
- **"wind_speed"** con una correlazione di 0.026
- **"precip_depth_1_hr"** con una correlazione di 0.006
- **"dew_temperature"** con una correlazione di -0.062
- **"air_temperature"** con una correlazione di -0.037
- **"sea_level_pressure"** con una correlazione di -0.015
- **"year_built"** con una correlazione di -0.102
- **"cloud_coverage"** con una correlazione di 0.002
- **"wind_direction"** con una correlazione di -0.002

Al fine di migliorare la comprensione dei dati, si effettua una conversione della variabile superficie: da piedi quadrati in metri quadrati.

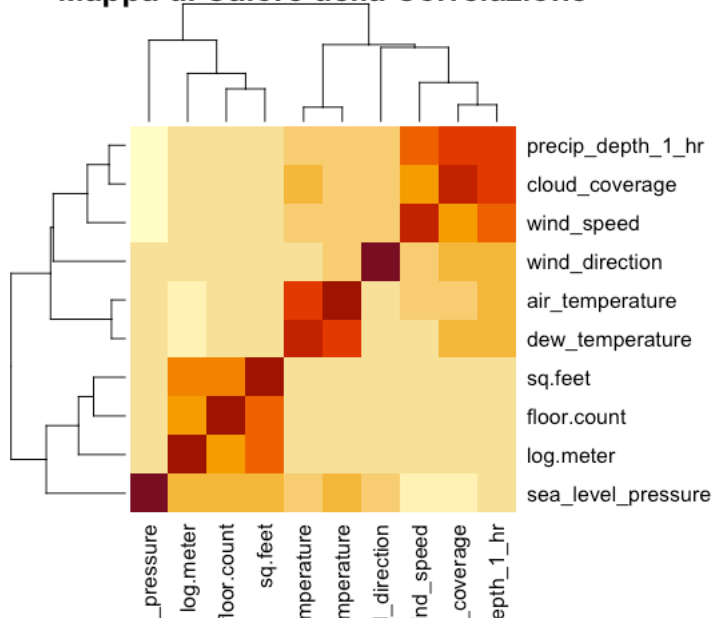
```
conversion_factor <- 0.092903
dati$sq.meters <- dati$sq.feet * conversion_factor
head(dati)
```

```
##      timestamp air_temperature cloud_coverage dew_temperature precip_depth_1_hr
## 1:           0              3.8       1.672312           2.4       0.6784231
## 2:           0              3.8       1.672312           2.4       0.6784231
## 3:           0              3.8       1.672312           2.4       0.6784231
## 4:           0              3.8       1.672312           2.4       0.6784231
## 5:           0              3.8       1.672312           2.4       0.6784231
## 6:           0              3.8       1.672312           2.4       0.6784231
##      sea_level_pressure wind_direction wind_speed building_id meter meter_reading
## 1:           1020.9           240           3.1           105           0       23.3036
## 2:           1020.9           240           3.1           106           0       0.3746
## 3:           1020.9           240           3.1           107           0       175.1840
## 4:           1020.9           240           3.1           108           0       91.2653
## 5:           1020.9           240           3.1           109           0       80.9300
## 6:           1020.9           240           3.1           110           0       86.2283
##      primary_use      year_built log.meter floor.count sq.feet day hour
## 1: Education 1960.92149019334 3.1906245           6   50624   0   0
## 2: Education 1960.17153742397 0.3181628           5   5375   0   0
## 3: Education           2005 5.1715289           11  97533   0   0
## 4: Education           1913 4.5246681           6   81581   0   0
## 5: Education           1953 4.4058652           7   56996   0   0
## 6: Education           2006 4.4685288           9   27815   0   0
##      sq.meters
## 1: 4703.1215
## 2: 499.3536
## 3: 9061.1083
## 4: 7579.1196
## 5: 5295.0994
## 6: 2584.0969
```

Dalla trasformazione in metriquadri di sq.feet si nota che gli edifici hanno delle dimensioni che variano da 500mq a 9000mq, pertanto è necessario tenerne conto nelle successive fasi di analisi.

```
heatmap(matrice_correlazione,
        # Scala dei colori
        main = "Mappa di Calore della Correlazione")
```


Mapa di Calore della Correlazione



Considerato il focus oggetto di studio ovvero la comprensione di come **log.meter** (variabile target) dipende dalle altre variabili (predittori), gli outlier rilevati nelle variabili :air_temperature, cloud_coverage, precip_depth_1_hr, sea_livel_pressure, wind_speed,year_built, log.meter, non dovrebbero penalizzarne lo studio.

Analisi delle componenti Principali

Effettuo adesso un'analisi delle componenti principali al fine di ridurre la dimensionalità e trasformare un insieme di variabili correlate in un nuovo insieme di variabili non correlate, definite appunto "componenti principali" combinazione lineare delle variabili originarie, ordinate in base all'importanza decrescente, in modo che le prime contengano la maggior parte delle informazioni.

Si effettuata la trasformazione delle variabili categoriche in numeriche al fine di includerle nell'analisi.

```
dati$building_id<-as.numeric(building_id)
dati$day<-as.numeric(day)
dati$hour<-as.numeric(hour)
dati$year_built<-as.numeric(year_built)
dati$meter<-as.numeric(meter)
str(dati)
```

```
## Classes 'data.table' and 'data.frame':  512872 obs. of  19 variables:
## $ timestamp      : int  0 0 0 0 0 0 0 0 0 0 ...
## $ air_temperature : num  3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 3.8 ...
## $ cloud_coverage  : num  1.67 1.67 1.67 1.67 1.67 1.67 ...
## $ dew_temperature : num  2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 2.4 ...
## $ precip_depth_1_hr : num  0.678 0.678 0.678 0.678 0.678 ...
## $ sea_level_pressure: num  1021 1021 1021 1021 1021 ...
## $ wind_direction  : num  240 240 240 240 240 240 240 240 240 240 ...
## $ wind_speed      : num  3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 3.1 ...
## $ building_id     : num  105 106 107 108 109 110 111 112 112 113 ...
## $ meter           : num  0 0 0 0 0 0 0 0 3 0 ...
## $ meter_reading    : num  23.304 0.375 175.184 91.265 80.93 ...
## $ primary_use      : Factor w/ 5 levels "Education","Entertainment/public assembly",...: 1 1 1 1 1 1 1 1 1 1
##
## $ year_built      : num  1961 1960 2005 1913 1953 ...
## $ log.meter        : num  3.191 0.318 5.172 4.525 4.406 ...
## $ floor.count      : num  6 5 11 6 7 9 8 7 7 10 ...
## $ sq.feet          : num  50624 5375 97533 81581 56996 ...
## $ day             : num  0 0 0 0 0 0 0 0 0 0 ...
## $ hour            : num  0 0 0 0 0 0 0 0 0 0 ...
## $ sq.meters        : num  4703 499 9061 7579 5295 ...
## - attr(*, ".internal.selfref")=<externalptr>
```

Standardizzazione dei dati, in modo da avere tutti i dati con media zero e deviazione standard unitaria

```
#autovalori
zdati <- datisub
zdati<- scale(zdati) # si effettua uno scale dei dati.
summary(zdati) # ci accertiamo che le variabili abbiano media 0 e varianza unitaria
```

```
## air_temperature cloud_coverage dew_temperature precip_depth_1_hr
## Min. :-2.52468 Min. :-1.7934 Min. :-3.1417 Min. :-2.36666
## 1st Qu.:-0.77229 1st Qu.:-0.1582 1st Qu.:-0.8010 1st Qu.:-0.57136
## Median :-0.08444 Median : 0.2727 Median : 0.0240 Median : 0.02705
## Mean : 0.00000 Mean : 0.0000 Mean : 0.0000 Mean : 0.00000
## 3rd Qu.: 0.71805 3rd Qu.: 0.5840 3rd Qu.: 0.7723 3rd Qu.: 0.60769
## Max. : 3.50222 Max. : 7.8437 Max. : 2.5374 Max. : 5.82562
## sea_level_pressure wind_direction wind_speed log.meter
## Min. :-3.9356 Min. :-1.9835 Min. :-1.7907 Min. :-3.36038
## 1st Qu.:-0.5823 1st Qu.:-0.8667 1st Qu.:-0.6201 1st Qu.:-0.61292
## Median : 0.1475 Median : 0.2502 Median : -0.1699 Median : 0.06642
## Mean : 0.0000 Mean : 0.0000 Mean : 0.0000 Mean : 0.00000
## 3rd Qu.: 0.6464 3rd Qu.: 0.7578 3rd Qu.: 0.5054 3rd Qu.: 0.74093
## Max. : 2.7341 Max. : 1.6716 Max. : 6.5383 Max. : 4.26423
## floor.count sq.feet
## Min. :-1.97549 Min. :-1.5041
## 1st Qu.:-0.78599 1st Qu.:-0.7887
## Median : 0.00701 Median :-0.2175
## Mean : 0.00000 Mean : 0.0000
## 3rd Qu.: 0.40351 3rd Qu.: 0.6061
## Max. : 3.57552 Max. : 2.6592
```

Stima degli autovalori, indicatori chiave che aiutano a capire quanto contribuisce alla varianza totale ciascuna componente principale.

```
autov=eigen(cor(zdati))
autov$values
```

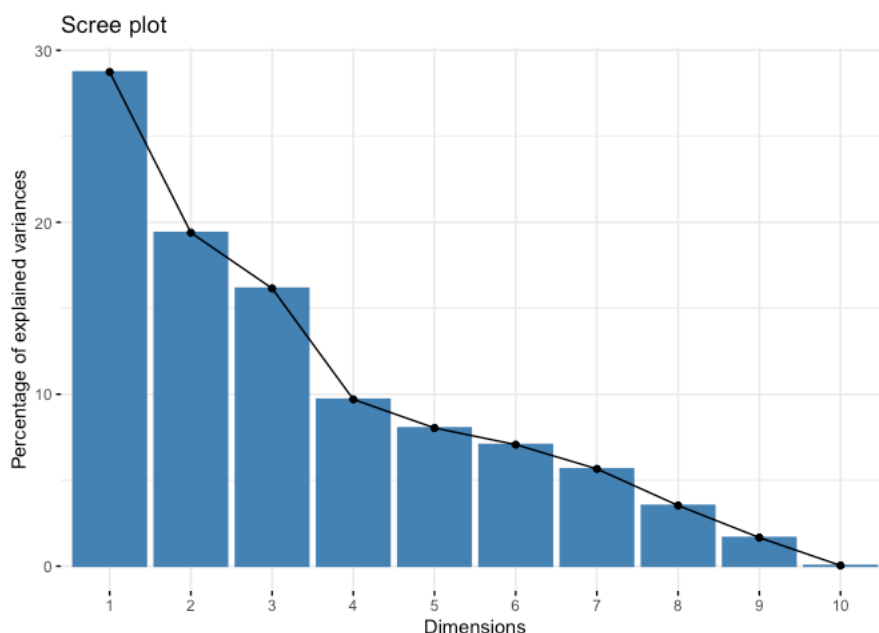
```
## [1] 2.873838767 1.939605499 1.615771626 0.970282851 0.803906425 0.707090631
## [7] 0.566042259 0.353124965 0.166409500 0.003927477
```

```
#utilizzo il package factoextra
res.pca<-prcomp(datisub,scale. = TRUE) #utilizzo il subset precedente ed imposto lo scale a TRUE anche se i dati
sono stati già normalizzati
eig.val<-get_eigenvalue(res.pca)
eig.val
```

```
## eigenvalue variance.percent cumulative.variance.percent
## Dim.1 2.873838767 28.73838767 28.73839
## Dim.2 1.939605499 19.39605499 48.13444
## Dim.3 1.615771626 16.15771626 64.29216
## Dim.4 0.970282851 9.70282851 73.99499
## Dim.5 0.803906425 8.03906425 82.03405
## Dim.6 0.707090631 7.07090631 89.10496
## Dim.7 0.566042259 5.66042259 94.76538
## Dim.8 0.353124965 3.53124965 98.29663
## Dim.9 0.166409500 1.66409500 99.96073
## Dim.10 0.003927477 0.03927477 100.00000
```

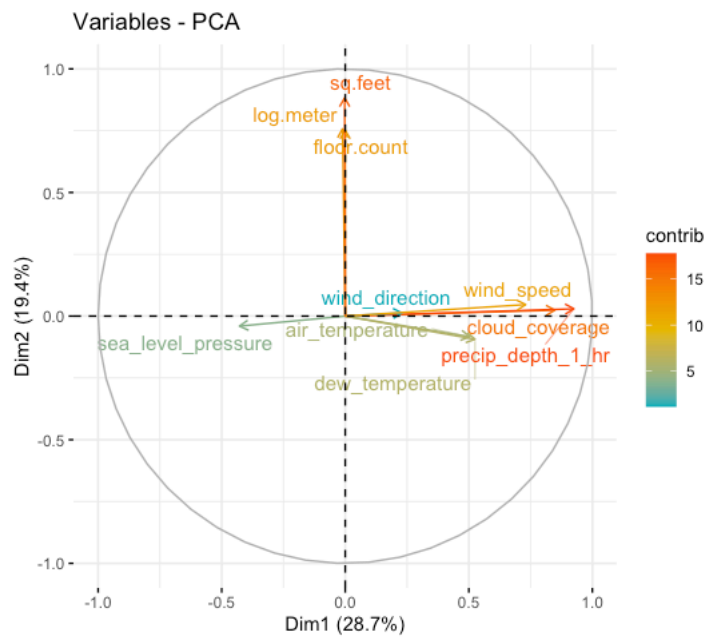
Si osservi come la percentuale di varianza espressa da ciascun autovalore decresce.

```
fviz_eig(res.pca) #grafico dei residui
```



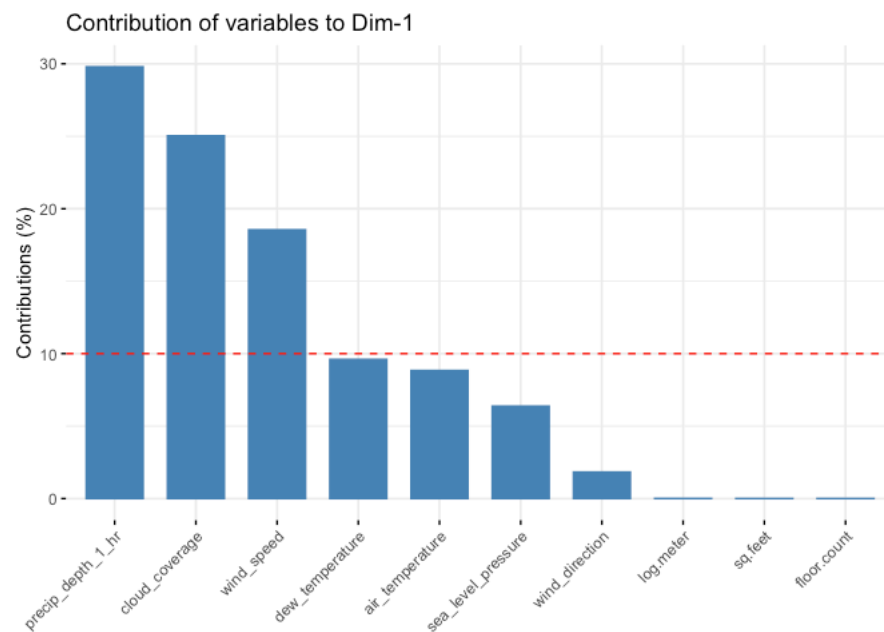
si osservino quindi le variabili contenute nelle componenti principali ed il relativo contributo.

```
fviz_pca_var(res.pca,col.var = "contrib", gradient.cols= c("#00AFBB","#E7B800","#FC4E07"),repel = TRUE) #grafico
delle variabili
```

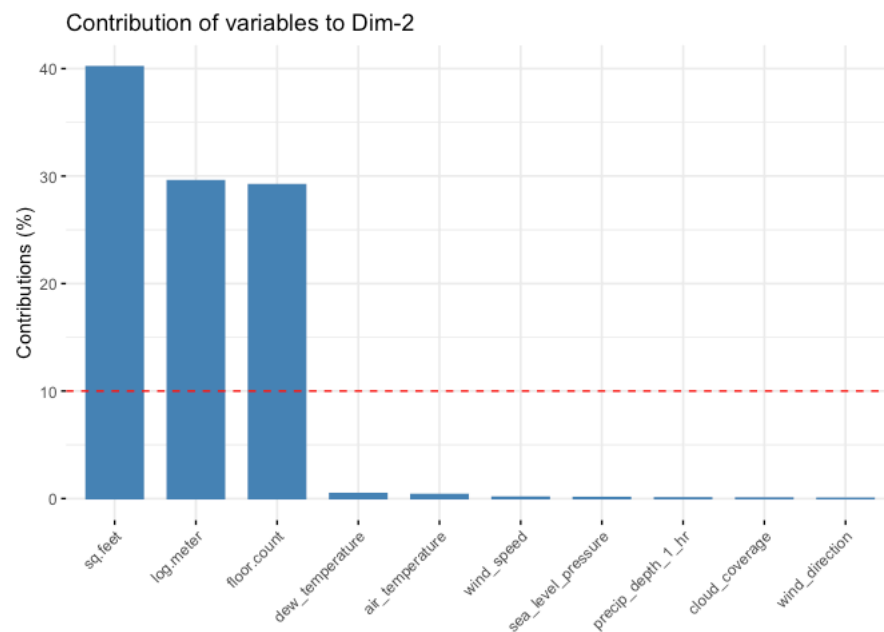


I successivi plot esplicitano il contributo di ciascuna variabile nelle due componenti principali.

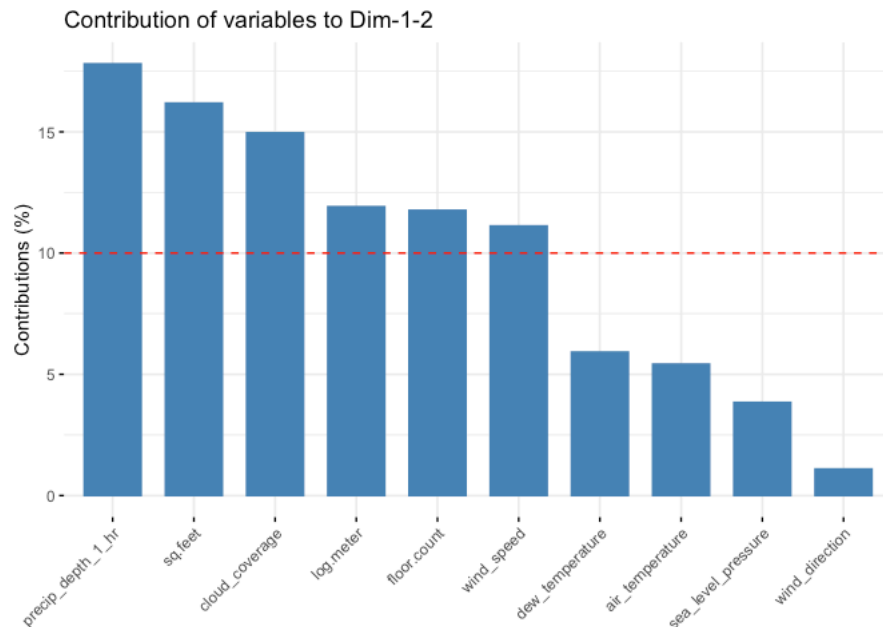
```
fviz_contrib(res.pca, choice = "var", axes = 1, top=10)
```



```
fviz_contrib(res.pca, choice = "var", axes = 2, top=10)
```



```
fviz_contrib(res.pca, choice = "var", axes = 1:2, top=10)
```



Si osservi che le variabili che danno il maggior contributo sono quelle che stanno sopra la linea rossa, che indica il contributo medio, ovvero **:sq.feet -log.meter-floor.count- precip_depth_1_hr - cloud_coverage - wind_speed.**

Si individua adesso il contributo espresso da ogni autovalore e la relativa percentuale di varianza espressa.

```
eig.val<-get_eigenvalue(res.pca)
eig.val
```

##	eigenvalue	variance.percent	cumulative.variance.percent
## Dim.1	2.873838767	28.73838767	28.73839
## Dim.2	1.939605499	19.39605499	48.13444
## Dim.3	1.615771626	16.15771626	64.29216
## Dim.4	0.970282851	9.70282851	73.99499
## Dim.5	0.803906425	8.03906425	82.03405
## Dim.6	0.707090631	7.07090631	89.10496
## Dim.7	0.566042259	5.66042259	94.76538
## Dim.8	0.353124965	3.53124965	98.29663
## Dim.9	0.166409500	1.66409500	99.96073
## Dim.10	0.003927477	0.03927477	100.00000

Individuate le variabili che danno il maggior contributo alla spiegazione della varianza in un modello lineare, passiamo all'individuazione

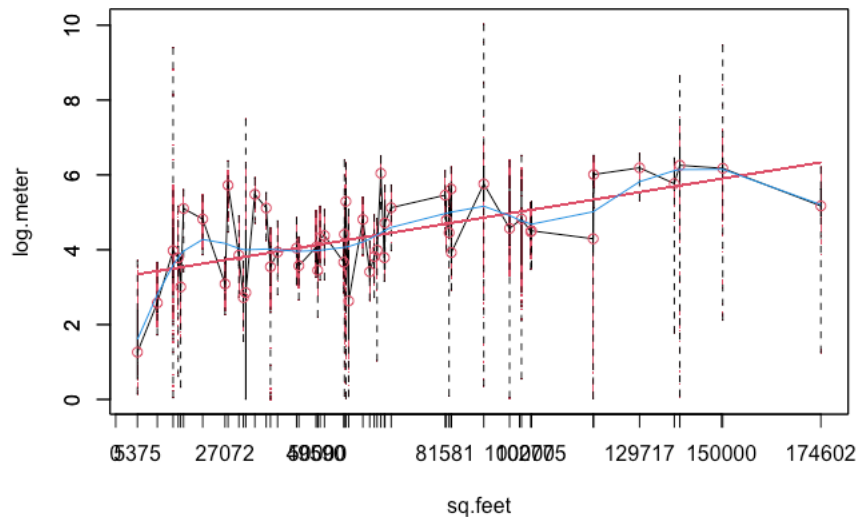
Analisi Regressione Lineare

Considerato il focus dell'analisi assegnata ovvero : **la comprensione di come log.meter (variabile target) dipende dalle altre variabili (predittori)**, gli studi precedenti, sia in termini di correlazione che sulle componenti principali si inizia con l'analisi della la relazione tra più evidente tra **log.meter** e **sq.feet**

Mod.(l1)

```
MLA.boxplot(log.meter,sq.feet,pch=".")
```

Boxplot, regression line, linear regression



```
## $m
##      5375      10302      14220      15490      16093      16803      21540      27072
## 1.263335 2.580988 3.977365 3.783154 3.005569 5.102512 4.820365 3.087130
##      27815      30496      31615      32207      34456      37266      38320      40086
## 5.725053 3.854279 2.722056 2.852572 5.481212 5.116749 3.544692 3.937213
##      44790      45349      49590      50021      50624      51689      56468      56630
## 4.043773 3.573796 4.064769 3.450486 4.317062 4.379765 3.666463 4.413278
##      56996      57674      61205      62894      64025      64724      65658      66533
## 5.292544 2.632409 4.809576 3.412103 3.816861 3.989176 6.044086 3.788356
##      66662      68212      81581      81882      82549      83044      83109      91150
## 4.715679 5.123604 5.458566 4.793024 4.455845 5.624716 3.924849 5.758838
##      97533      100482      102775      102958      118232      118339      129717      138317
## 4.570665 4.822962 4.482725 4.506635 4.293645 6.011138 6.190022 5.767139
##      139684      150319      174602
## 6.257289 6.176059 5.171879
##
## $x
## [1] 5375 10302 14220 15490 16093 16803 21540 27072 27815 30496
## [11] 31615 32207 34456 37266 38320 40086 44790 45349 49590 50021
## [21] 50624 51689 56468 56630 56996 57674 61205 62894 64025 64724
## [31] 65658 66533 66662 68212 81581 81882 82549 83044 83109 91150
## [41] 97533 100482 102775 102958 118232 118339 129717 138317 139684 150319
## [51] 174602
```

Com'era normale attendersi si nota una relazione lineare positiva tra il consumo di energia e l'aumento della superficie, la retta blu rappresenta la retta di regressione, mentre quella rossa la curva di regressione non parametrica ovvero quel modello non analitico che cerca di seguire al meglio le osservazioni, più queste due sono vicine e più c'è una relazione lineare tra le variabili.

Si osservino i dati riepilogativi del modello.

```
lm1=lm(log.meter~sq.feet,data=dati)
summary(lm1)
```

```
##
## Call:
## lm(formula = log.meter ~ sq.feet, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.6407 -0.5991  0.0435  0.7468  5.9017
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.248e+00  2.949e-03 1101.4  <2e-16 ***
## sq.feet      1.767e-05  3.783e-08  467.2  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.101 on 512870 degrees of freedom
## Multiple R-squared:  0.2985, Adjusted R-squared:  0.2985
## F-statistic: 2.183e+05 on 1 and 512870 DF, p-value: < 2.2e-16
```

```
AIC(lm1)
```

```
## [1] 1554354
```

Il modello lineare appena rappresentato, esprime la relazione tra la variabile Y= log.meter e X=sq.feet

$$Y = X\beta + \varepsilon$$

$$\log.\text{meter} = 3.248 + \text{sq.feet} \cdot 1,767^{(-5)}$$

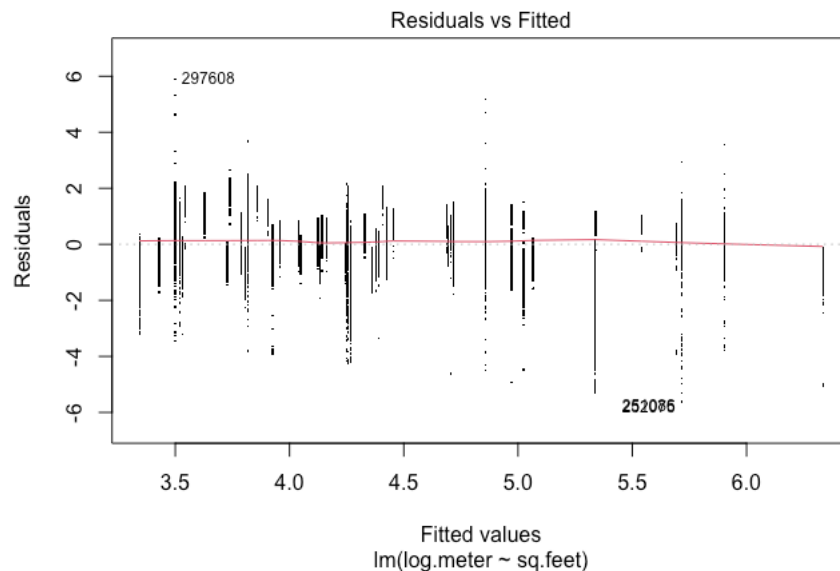
Questo modello esprime il **29,85%** di varianza, inoltre il *t value* dello stimatore, calcolato rispetto alla variabile *sq.feet* è significativamente diverso da zero pertanto la variabile è statisticamente significativa.

Quando si considerano diversi modelli ed si vuole valutare il modello ottimale in termini di complessità e livello di adattamento (fitting), l'AIC score, rappresenta una misura della bontà di adattamento del modello ed è utile per confrontare diversi modelli, poichè in breve meglio rappresenta la distribuzione degli errori residui.

Si preferisce solitamente il modello con l'AIC più basso, poichè offre una buona adattabilità con una complessità minima. In questo caso l'AIC è pari a : **1.554.354**

Il grafico sottostante rappresenta la distribuzione dei residui, i quali dovrebbero equidistribuirsi sopra e sotto la retta $y=0$

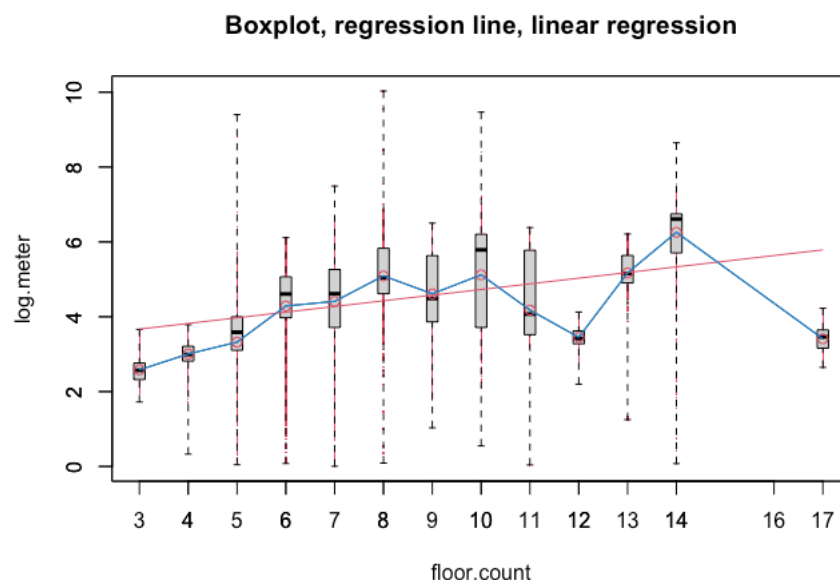
```
plot(lm1,1, pch='.')
```



Si osservi adesso la relazione tra **log.meter** e **floor.count**

Mod.(l2)

```
MLA.boxplot(log.meter, floor.count, pch=".")
```



```
## $m
##      3      4      5      6      7      8      9     10
## 2.580988 3.005569 3.321632 4.288096 4.411492 5.092433 4.613219 5.120680
##      11     12     13     14     17
## 4.179510 3.450486 5.171879 6.257289 3.412103
##
## $x
## [1] 3 4 5 6 7 8 9 10 11 12 13 14 17
```

```
lm2=lm(log.meter~floor.count,data=dati)
summary(lm2)
```

```
##
## Call:
## lm(formula = log.meter ~ floor.count, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.2574 -0.7638  0.1433  0.9773  5.6040
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.2165802   0.0058319   551.5  <2e-16 ***
## floor.count  0.1512026   0.0006967   217.0  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.258 on 512870 degrees of freedom
## Multiple R-squared:  0.08412, Adjusted R-squared:  0.08412
## F-statistic: 4.711e+04 on 1 and 512870 DF, p-value: < 2.2e-16
```

```
AIC(lm2)
```

```
## [1] 1691140
```

I risultati ottenuti mostrano una percentuale di varianza espressa pari 8.41% ed uno score AIC maggiore, pertanto peggiorato, si ricordi che il modello con l'AIC più basso fornisce un migliore adattamento ai dati con meno complessità di variabili.

Si provi, dunque, a ricercare un modello migliore introducendo una seconda variabile “*dew.temperature*”, inoltre essendo molto vasta la scala della variabile “*sq.feet*” se ne consideri il logaritmo.

```
plot
```

```
## function (x, y, ...)
## UseMethod("plot")
## <bytecode: 0x10b60eea8>
## <environment: namespace:base>
```

```
MLA.regression3d(dati[,c(16,4,14)])
```

```
lm2=lm(log.meter~log(sq.feet)+dew_temperature,data=dati)
summary(lm2)
```

```
##
## Call:
## lm(formula = log.meter ~ log(sq.feet) + dew_temperature, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.2920 -0.6533  0.0234  0.7907  6.3781
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.8908288   0.0217184  -271.24  <2e-16 ***
## log(sq.feet)   0.9607956   0.0019853   483.95  <2e-16 ***
## dew_temperature -0.0171078   0.0002913   -58.73  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.087 on 512869 degrees of freedom
## Multiple R-squared:  0.3162, Adjusted R-squared:  0.3162
## F-statistic: 1.186e+05 on 2 and 512869 DF, p-value: < 2.2e-16
```

```
AIC(lm1)
```

```
## [1] 1554354
```

Si noti che il contributo al coefficiente di determinazione multipla “R-squared” non è aumentato sensibilmente, inoltre anche lo stimatore t-value è sensibilmente piccolo, pertanto effettua un'esplorazione successiva introducendo delle variabili fattoriali.

Mod(3)

```
l3=lm(log.meter~sq.feet+primary_use,data=dati)
summary(l3)
```

```
##
## Call:
## lm(formula = log.meter ~ sq.feet + primary_use, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.9459 -0.3888  0.1170  0.5758  6.0398
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.664e+00  3.461e-03 1058.48  <2e-16
## sq.feet          1.688e-05  3.723e-08  453.38  <2e-16
## primary_useEntertainment/public assembly -9.279e-01  1.133e-02  -81.89  <2e-16
## primary_useLodging/residential          -8.528e-01  4.114e-03 -207.29  <2e-16
## primary_useOffice          -5.427e-01  3.530e-03 -153.73  <2e-16
## primary_usePublic services          -8.428e-01  6.825e-03 -123.47  <2e-16
##
## (Intercept)          ***
## sq.feet              ***
## primary_useEntertainment/public assembly ***
## primary_useLodging/residential          ***
## primary_useOffice          ***
## primary_usePublic services          ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.041 on 512866 degrees of freedom
## Multiple R-squared:  0.3727, Adjusted R-squared:  0.3727
## F-statistic: 6.095e+04 on 5 and 512866 DF, p-value: < 2.2e-16
```

```
AIC(l3)
```

```
## [1] 1497024
```

L'introduzione della variabile `primary_use` ha portato ad un sensibile miglioramento del modello, sia in termini di Varianza espressa che riduzione dell'indice AIC.

Si provi adesso ad effettuare una stima del modello introducendo tutte le variabili indipendenti e successivamente attraverso la tecnica "backward selection" si stimano tutti i modelli i-esimi eliminando di volta in volta la variabile meno significativa, ovvero con il p-value più alto e il t-value più piccolo.

Mod.(l17)

```
l17=lm(log.meter~log(sq.feet)+as.factor(primary_use)+floor.count+air_temperature+cloud_coverage+dew_temperature+
recip_depth_1_hr+sea_level_pressure+wind_direction+wind_speed+meter+as.factor(day%7)+as.factor(hour),data=dati )
summary(l17)
```



```
##
## Call:
## lm(formula = log.meter ~ log(sq.feet) + as.factor(primary_use) +
##     floor.count + air_temperature + cloud_coverage + dew_temperature +
##     precip_depth_1_hr + sea_level_pressure + wind_direction +
##     wind_speed + meter + as.factor(day%7) + as.factor(hour),
##     data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.8035 -0.3152  0.0988  0.4463  6.7145
##
## Coefficients:
##              Estimate Std. Error
## (Intercept)      -9.066e+00  3.077e-01
## log(sq.feet)       9.886e-01  2.294e-03
## as.factor(primary_use)Entertainment/public assembly -1.306e+00  1.068e-02
## as.factor(primary_use)Lodging/residential          -1.163e+00  4.080e-03
## as.factor(primary_use)Office                       -6.603e-01  3.313e-03
## as.factor(primary_use)Public services              -9.542e-01  6.563e-03
## floor.count      -1.637e-04  7.045e-04
## air_temperature  -1.806e-02  5.556e-04
## cloud_coverage    2.501e-01  1.262e-02
## dew_temperature   2.035e-03  6.936e-04
## precip_depth_1_hr -4.740e-01  2.323e-02
## sea_level_pressure 2.943e-03  2.863e-04
## wind_direction    1.404e-04  1.828e-05
## wind_speed        7.680e-02  3.547e-03
## meter            -1.174e-01  1.419e-03
## as.factor(day%7)1 -1.040e-01  5.063e-03
## as.factor(day%7)2 -1.476e-01  5.097e-03
## as.factor(day%7)3  2.584e-03  5.091e-03
## as.factor(day%7)4  2.327e-02  5.088e-03
## as.factor(day%7)5  2.571e-02  5.089e-03
## as.factor(day%7)6  1.969e-02  5.091e-03
## as.factor(hour)1  -1.457e-02  9.496e-03
## as.factor(hour)2  -4.161e-02  9.485e-03
## as.factor(hour)3  -5.798e-02  9.486e-03
## as.factor(hour)4  -6.369e-02  9.483e-03
## as.factor(hour)5  -5.220e-02  9.475e-03
## as.factor(hour)6  -7.831e-03  9.454e-03
## as.factor(hour)7   5.343e-02  9.457e-03
## as.factor(hour)8   1.151e-01  9.457e-03
## as.factor(hour)9   1.908e-01  9.473e-03
## as.factor(hour)10  2.388e-01  9.531e-03
## as.factor(hour)11  2.716e-01  9.611e-03
## as.factor(hour)12  2.966e-01  9.709e-03
## as.factor(hour)13  3.020e-01  9.753e-03
## as.factor(hour)14  3.047e-01  9.827e-03
## as.factor(hour)15  3.008e-01  9.834e-03
## as.factor(hour)16  2.906e-01  9.790e-03
## as.factor(hour)17  2.742e-01  9.718e-03
## as.factor(hour)18  2.438e-01  9.631e-03
## as.factor(hour)19  2.000e-01  9.571e-03
## as.factor(hour)20  1.501e-01  9.519e-03
## as.factor(hour)21  1.089e-01  9.488e-03
## as.factor(hour)22  7.143e-02  9.490e-03
## as.factor(hour)23  4.085e-02  9.483e-03
##              t value Pr(>|t|)
## (Intercept)      -29.469 < 2e-16 ***
## log(sq.feet)     431.005 < 2e-16 ***
## as.factor(primary_use)Entertainment/public assembly -122.355 < 2e-16 ***
## as.factor(primary_use)Lodging/residential          -284.966 < 2e-16 ***
## as.factor(primary_use)Office                       -199.320 < 2e-16 ***
## as.factor(primary_use)Public services              -145.398 < 2e-16 ***
## floor.count       -0.232  0.81622
## air_temperature  -32.498 < 2e-16 ***
## cloud_coverage    19.811 < 2e-16 ***
## dew_temperature   2.934  0.00335 **
## precip_depth_1_hr -20.408 < 2e-16 ***
## sea_level_pressure 10.280 < 2e-16 ***
## wind_direction    7.678  1.62e-14 ***
## wind_speed       21.650 < 2e-16 ***
## meter            -82.711 < 2e-16 ***
## as.factor(day%7)1 -20.547 < 2e-16 ***
## as.factor(day%7)2 -28.951 < 2e-16 ***
## as.factor(day%7)3  0.508  0.61177
## as.factor(day%7)4  4.573  4.81e-06 ***
## as.factor(day%7)5  5.052  4.38e-07 ***
## as.factor(day%7)6  3.867  0.00011 ***
## as.factor(hour)1  -1.535  0.12483
## as.factor(hour)2  -4.386  1.15e-05 ***
## as.factor(hour)3  -6.112  9.83e-10 ***
## as.factor(hour)4  -6.716  1.87e-11 ***
## as.factor(hour)5  -5.509  3.61e-08 ***
## as.factor(hour)6  -0.828  0.40754
## as.factor(hour)7   5.650  1.60e-08 ***
```

```
## as.factor(hour)8          12.175 < 2e-16 ***
## as.factor(hour)9          20.138 < 2e-16 ***
## as.factor(hour)10         25.052 < 2e-16 ***
## as.factor(hour)11         28.260 < 2e-16 ***
## as.factor(hour)12         30.547 < 2e-16 ***
## as.factor(hour)13         30.961 < 2e-16 ***
## as.factor(hour)14         31.007 < 2e-16 ***
## as.factor(hour)15         30.582 < 2e-16 ***
## as.factor(hour)16         29.685 < 2e-16 ***
## as.factor(hour)17         28.214 < 2e-16 ***
## as.factor(hour)18         25.313 < 2e-16 ***
## as.factor(hour)19         20.894 < 2e-16 ***
## as.factor(hour)20         15.767 < 2e-16 ***
## as.factor(hour)21         11.473 < 2e-16 ***
## as.factor(hour)22          7.527 5.20e-14 ***
## as.factor(hour)23          4.308 1.65e-05 ***
## ----
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9755 on 512828 degrees of freedom
## Multiple R-squared:  0.4496, Adjusted R-squared:  0.4495
## F-statistic: 9741 on 43 and 512828 DF, p-value: < 2.2e-16
```

```
AIC(l17)
```

```
## [1] 1430083
```

In questo modello, computazionalmente elaborato, considerando tutte le variabili indipendenti numeriche ed le variabili fattoriali data, ora, e primary_use; otteniamo una percentuale di varianza espressa pari a 44.95% ed uno score di AIC pari a 1430083.

```
library(leaps)
mod_back=regsubsets(log.meter~log(sq.feet)+primary_use+floor.count+air_temperature+cloud_coverage+dew_temperature
+precip_depth_1_hr+sea_level_pressure+wind_direction+wind_speed+meter+as.factor(day%7)+as.factor(hour), data=dat
i, nvmax = 13, method = "backward")
summary(mod_back)
```

```

## Subset selection object
## Call: regsubsets.formula(log.meter ~ log(sq.feet) + primary_use + floor.count +
##   air_temperature + cloud_coverage + dew_temperature + precip_depth_1_hr +
##   sea_level_pressure + wind_direction + wind_speed + meter +
##   as.factor(day%7) + as.factor(hour), data = dati, nvmax = 13,
##   method = "backward")
## 43 Variables (and intercept)
##
##                                     Forced in Forced out
## log(sq.feet)                        FALSE      FALSE
## primary_useEntertainment/public assembly  FALSE      FALSE
## primary_useLodging/residential          FALSE      FALSE
## primary_useOffice                       FALSE      FALSE
## primary_usePublic services              FALSE      FALSE
## floor.count                            FALSE      FALSE
## air_temperature                        FALSE      FALSE
## cloud_coverage                         FALSE      FALSE
## dew_temperature                        FALSE      FALSE
## precip_depth_1_hr                     FALSE      FALSE
## sea_level_pressure                     FALSE      FALSE
## wind_direction                         FALSE      FALSE
## wind_speed                            FALSE      FALSE
## meter                                 FALSE      FALSE
## as.factor(day%7)1                      FALSE      FALSE
## as.factor(day%7)2                      FALSE      FALSE
## as.factor(day%7)3                      FALSE      FALSE
## as.factor(day%7)4                      FALSE      FALSE
## as.factor(day%7)5                      FALSE      FALSE
## as.factor(day%7)6                      FALSE      FALSE
## as.factor(hour)1                      FALSE      FALSE
## as.factor(hour)2                      FALSE      FALSE
## as.factor(hour)3                      FALSE      FALSE
## as.factor(hour)4                      FALSE      FALSE
## as.factor(hour)5                      FALSE      FALSE
## as.factor(hour)6                      FALSE      FALSE
## as.factor(hour)7                      FALSE      FALSE
## as.factor(hour)8                      FALSE      FALSE
## as.factor(hour)9                      FALSE      FALSE
## as.factor(hour)10                     FALSE      FALSE
## as.factor(hour)11                     FALSE      FALSE
## as.factor(hour)12                     FALSE      FALSE
## as.factor(hour)13                     FALSE      FALSE
## as.factor(hour)14                     FALSE      FALSE
## as.factor(hour)15                     FALSE      FALSE
## as.factor(hour)16                     FALSE      FALSE
## as.factor(hour)17                     FALSE      FALSE
## as.factor(hour)18                     FALSE      FALSE
## as.factor(hour)19                     FALSE      FALSE
## as.factor(hour)20                     FALSE      FALSE
## as.factor(hour)21                     FALSE      FALSE
## as.factor(hour)22                     FALSE      FALSE
## as.factor(hour)23                     FALSE      FALSE
## 1 subsets of each size up to 13
## Selection Algorithm: backward
##           log(sq.feet) primary_useEntertainment/public assembly
## 1 ( 1 ) "*"            " "
## 2 ( 1 ) "*"            " "
## 3 ( 1 ) "*"            " "
## 4 ( 1 ) "*"            " "
## 5 ( 1 ) "*"            "*"
## 6 ( 1 ) "*"            "*"
## 7 ( 1 ) "*"            "*"
## 8 ( 1 ) "*"            "*"
## 9 ( 1 ) "*"            "*"
## 10 ( 1 ) "*"           "*"
## 11 ( 1 ) "*"           "*"
## 12 ( 1 ) "*"           "*"
## 13 ( 1 ) "*"           "*"
##           primary_useLodging/residential primary_useOffice
## 1 ( 1 ) " "                " "
## 2 ( 1 ) "*"                " "
## 3 ( 1 ) "*"                "*"
## 4 ( 1 ) "*"                "*"
## 5 ( 1 ) "*"                "*"
## 6 ( 1 ) "*"                "*"
## 7 ( 1 ) "*"                "*"
## 8 ( 1 ) "*"                "*"
## 9 ( 1 ) "*"                "*"
## 10 ( 1 ) "*"               "*"
## 11 ( 1 ) "*"               "*"
## 12 ( 1 ) "*"               "*"
## 13 ( 1 ) "*"               "*"
##           primary_usePublic services floor.count air_temperature cloud_coverage
## 1 ( 1 ) " "                " "            " "            " "
## 2 ( 1 ) " "                " "            " "            " "
## 3 ( 1 ) " "                " "            " "            " "
## 4 ( 1 ) "*"                " "            " "            " "
## 5 ( 1 ) "*"                " "            " "            " "

```

```

## 6 ( 1 ) "*" " " " " " "
## 7 ( 1 ) "*" " " "*" " "
## 8 ( 1 ) "*" " " "*" " "
## 9 ( 1 ) "*" " " "*" " "
## 10 ( 1 ) "*" " " "*" " "
## 11 ( 1 ) "*" " " "*" " "
## 12 ( 1 ) "*" " " "*" " "
## 13 ( 1 ) "*" " " "*" " "
##
## dew_temperature precip_depth_1_hr sea_level_pressure wind_direction
## 1 ( 1 ) " " " " " " " "
## 2 ( 1 ) " " " " " " " "
## 3 ( 1 ) " " " " " " " "
## 4 ( 1 ) " " " " " " " "
## 5 ( 1 ) " " " " " " " "
## 6 ( 1 ) " " " " " " " "
## 7 ( 1 ) " " " " " " " "
## 8 ( 1 ) " " " " " " " "
## 9 ( 1 ) " " " " " " " "
## 10 ( 1 ) " " " " " " " "
## 11 ( 1 ) " " " " " " " "
## 12 ( 1 ) " " " " " " " "
## 13 ( 1 ) " " " " " " " "
##
## wind_speed meter as.factor(day%7)1 as.factor(day%7)2
## 1 ( 1 ) " " " " " " " "
## 2 ( 1 ) " " " " " " " "
## 3 ( 1 ) " " " " " " " "
## 4 ( 1 ) " " " " " " " "
## 5 ( 1 ) " " " " " " " "
## 6 ( 1 ) " " "*" " " " " "
## 7 ( 1 ) " " "*" " " " " "
## 8 ( 1 ) " " "*" " " " "*"
## 9 ( 1 ) " " "*" "*" " "*"
## 10 ( 1 ) " " "*" "*" " "*"
## 11 ( 1 ) " " "*" "*" " "*"
## 12 ( 1 ) " " "*" "*" " "*"
## 13 ( 1 ) " " "*" "*" " "*"
##
## as.factor(day%7)3 as.factor(day%7)4 as.factor(day%7)5
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) " " " " " "
##
## as.factor(day%7)6 as.factor(hour)1 as.factor(hour)2 as.factor(hour)3
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) " " " " " "
##
## as.factor(hour)4 as.factor(hour)5 as.factor(hour)6 as.factor(hour)7
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) " " " " " "
##
## as.factor(hour)8 as.factor(hour)9 as.factor(hour)10 as.factor(hour)11
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "

```

```
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) " " " " " "
## as.factor(hour)12 as.factor(hour)13 as.factor(hour)14
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) "*" " " " "
## as.factor(hour)15 as.factor(hour)16 as.factor(hour)17
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) "*" " " " "
## 13 ( 1 ) "*" " " " "
## as.factor(hour)18 as.factor(hour)19 as.factor(hour)20
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) " " " " " "
## as.factor(hour)21 as.factor(hour)22 as.factor(hour)23
## 1 ( 1 ) " " " " " "
## 2 ( 1 ) " " " " " "
## 3 ( 1 ) " " " " " "
## 4 ( 1 ) " " " " " "
## 5 ( 1 ) " " " " " "
## 6 ( 1 ) " " " " " "
## 7 ( 1 ) " " " " " "
## 8 ( 1 ) " " " " " "
## 9 ( 1 ) " " " " " "
## 10 ( 1 ) " " " " " "
## 11 ( 1 ) " " " " " "
## 12 ( 1 ) " " " " " "
## 13 ( 1 ) " " " " " "
```

```
summary(mod_back)$adjr2
```

```
## [1] 0.3115724 0.3633353 0.3894418 0.4127404 0.4274688 0.4333319 0.4354284
## [8] 0.4367908 0.4377596 0.4384376 0.4391667 0.4399717 0.4408372
```

Se si considera l'andamento delle percentuali di varianza espressa, il modello migliore risulta essere quello a 6 predittori.

Effettuando una stima dei modelli attraverso il metodo backward analysis non otteniamo valori soddisfacenti pertanto, si passa ad un'ulteriore esplorazione considerando le variabili fattoriali : ***year_built*** e ***primary_use***.

Mod.(l5)

```
l5=lm(log.meter~log(sq.feet)+primary_use+year_built+day+hour,data=dati)
summary(l5)
```

```
##
## Call:
## lm(formula = log.meter ~ log(sq.feet) + primary_use + year_built +
##     day + hour, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.7699 -0.3366  0.1125  0.4951  6.4553
##
## Coefficients:
##              Estimate Std. Error t value
## (Intercept)    -5.889e+00  1.090e-01  -54.051
## log(sq.feet)     9.888e-01  1.934e-03  511.318
## primary_useEntertainment/public assembly -1.242e+00  1.079e-02 -115.068
## primary_useLodging/residential    -1.099e+00  3.927e-03 -279.724
## primary_useOffice    -6.164e-01  3.347e-03 -184.165
## primary_usePublic services    -9.937e-01  6.711e-03 -148.066
## year_built      2.470e-07  5.190e-05    0.005
## day            -5.089e-04  1.298e-05  -39.214
## hour           8.493e-03  2.005e-04   42.361
##
##              Pr(>|t|)
## (Intercept)    <2e-16 ***
## log(sq.feet)    <2e-16 ***
## primary_useEntertainment/public assembly <2e-16 ***
## primary_useLodging/residential    <2e-16 ***
## primary_useOffice    <2e-16 ***
## primary_usePublic services    <2e-16 ***
## year_built         0.996
## day                <2e-16 ***
## hour                <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9916 on 512863 degrees of freedom
## Multiple R-squared:  0.4312, Adjusted R-squared:  0.4312
## F-statistic: 4.859e+04 on 8 and 512863 DF,  p-value: < 2.2e-16
```

AIC(l5)

```
## [1] 1446876
```

Il modello appena analizzato, seppur con meno variabili introdotte, ha delle prestazioni migliori in termini di varianza espressa e **AIC score**, quindi provo ad introdurre un ulteriore variabile fattoriale che identifica univocamente l'immobile ovvero la variabile building_id.

```
l17=lm(log.meter~log(sq.feet)+as.factor(primary_use)+floor.count+air_temperature+cloud_coverage+dew_temperature+p
recip_depth_1_hr+sea_level_pressure+wind_direction+wind_speed+meter+as.factor(building_id)+as.factor(day%7)+as.f
actor(hour),data=dati)
summary(l17)
```

```
##
## Call:
## lm(formula = log.meter ~ log(sq.feet) + as.factor(primary_use) +
##     floor.count + air_temperature + cloud_coverage + dew_temperature +
##     precip_depth_1_hr + sea_level_pressure + wind_direction +
##     wind_speed + meter + as.factor(building_id) + as.factor(day%7) +
##     as.factor(hour), data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.0227 -0.2140  0.0188  0.2657  5.5543
##
## Coefficients: (6 not defined because of singularities)
##              Estimate Std. Error
## (Intercept)    2.358e+01  3.170e-01
## log(sq.feet)   -2.716e+00  2.466e-02
## as.factor(primary_use)Entertainment/public assembly -7.710e-01  1.020e-02
## as.factor(primary_use)Lodging/residential          -3.988e+00  2.036e-02
## as.factor(primary_use)Office                       -2.607e+00  2.058e-02
## as.factor(primary_use)Public services              -4.563e-01  9.749e-03
## floor.count      9.086e-01  5.061e-03
## air_temperature -2.076e-02  3.772e-04
## cloud_coverage   3.301e-01  8.570e-03
## dew_temperature  3.778e-03  4.707e-04
## precip_depth_1_hr -6.228e-01  1.577e-02
## sea_level_pressure 4.274e-03  1.943e-04
## wind_direction   1.989e-04  1.241e-05
## wind_speed       9.953e-02  2.408e-03
## meter           -9.752e-02  1.121e-03
## as.factor(building_id)106 -8.188e+00  5.143e-02
## as.factor(building_id)107 -2.509e+00  1.526e-02
## as.factor(building_id)108  2.437e+00  1.544e-02
## as.factor(building_id)109  4.323e-01  9.794e-03
## as.factor(building_id)110 -2.944e+00  3.115e-02
## as.factor(building_id)111  2.183e+00  1.558e-02
## as.factor(building_id)112 -3.548e+00  1.849e-02
## as.factor(building_id)113 -1.130e+00  1.144e-02
## as.factor(building_id)114 -2.483e+00  2.106e-02
## as.factor(building_id)115  3.520e+00  2.106e-02
## as.factor(building_id)116 -3.230e-02  1.253e-02
## as.factor(building_id)117 -2.708e+00  2.607e-02
## as.factor(building_id)118  1.454e+00  1.542e-02
## as.factor(building_id)119  1.352e+00  1.077e-02
## as.factor(building_id)120 -2.009e-01  1.080e-02
## as.factor(building_id)121  1.317e+00  1.370e-02
## as.factor(building_id)122  1.743e+00  1.259e-02
## as.factor(building_id)123  9.935e-02  1.015e-02
## as.factor(building_id)124 -2.437e+00  1.543e-02
## as.factor(building_id)125 -3.118e+00  3.353e-02
## as.factor(building_id)126 -1.817e+00  2.333e-02
## as.factor(building_id)127  1.500e-01  1.192e-02
## as.factor(building_id)128  4.260e+00  2.649e-02
## as.factor(building_id)129  4.289e+00  2.653e-02
## as.factor(building_id)130 -6.322e+00  3.443e-02
## as.factor(building_id)131  3.317e+00  1.724e-02
## as.factor(building_id)132  2.217e+00  1.753e-02
## as.factor(building_id)133  1.602e+00  1.300e-02
## as.factor(building_id)134  9.542e-01  1.027e-02
## as.factor(building_id)135 -3.413e-01  1.018e-02
## as.factor(building_id)136 NA NA
## as.factor(building_id)137 NA NA
## as.factor(building_id)138  1.928e+00  1.995e-02
## as.factor(building_id)139 NA NA
## as.factor(building_id)140 -3.744e+00  1.528e-02
## as.factor(building_id)141  2.001e+00  1.788e-02
## as.factor(building_id)142  2.473e+00  2.270e-02
## as.factor(building_id)143  2.502e+00  1.995e-02
## as.factor(building_id)144  1.334e+00  2.327e-02
## as.factor(building_id)145 -1.661e-01  1.117e-02
## as.factor(building_id)146 -2.993e+00  1.121e-02
## as.factor(building_id)147  2.256e+00  2.284e-02
## as.factor(building_id)148  4.635e-01  2.047e-02
## as.factor(building_id)149  1.663e+00  1.850e-02
## as.factor(building_id)150  2.314e+00  1.392e-02
## as.factor(building_id)151 -1.959e+00  1.180e-02
## as.factor(building_id)152 -7.273e-01  1.273e-02
## as.factor(building_id)153 NA NA
## as.factor(building_id)154 NA NA
## as.factor(building_id)155 NA NA
## as.factor(day%7)1 -1.028e-01  3.437e-03
## as.factor(day%7)2 -1.521e-01  3.459e-03
## as.factor(day%7)3  2.620e-03  3.455e-03
## as.factor(day%7)4  2.198e-02  3.454e-03
## as.factor(day%7)5  2.528e-02  3.454e-03
## as.factor(day%7)6  2.039e-02  3.455e-03
## as.factor(hour)1 -1.381e-02  6.445e-03
## as.factor(hour)2 -4.007e-02  6.438e-03
```

## as.factor(hour)3	-5.775e-02	6.438e-03
## as.factor(hour)4	-6.311e-02	6.436e-03
## as.factor(hour)5	-4.872e-02	6.431e-03
## as.factor(hour)6	6.362e-03	6.417e-03
## as.factor(hour)7	6.661e-02	6.418e-03
## as.factor(hour)8	1.285e-01	6.418e-03
## as.factor(hour)9	2.058e-01	6.429e-03
## as.factor(hour)10	2.541e-01	6.469e-03
## as.factor(hour)11	2.871e-01	6.523e-03
## as.factor(hour)12	3.133e-01	6.590e-03
## as.factor(hour)13	3.180e-01	6.619e-03
## as.factor(hour)14	3.226e-01	6.670e-03
## as.factor(hour)15	3.156e-01	6.675e-03
## as.factor(hour)16	3.068e-01	6.645e-03
## as.factor(hour)17	2.904e-01	6.596e-03
## as.factor(hour)18	2.591e-01	6.537e-03
## as.factor(hour)19	2.140e-01	6.496e-03
## as.factor(hour)20	1.625e-01	6.460e-03
## as.factor(hour)21	1.203e-01	6.440e-03
## as.factor(hour)22	7.669e-02	6.441e-03
## as.factor(hour)23	4.351e-02	6.436e-03
##	t value	Pr(> t)
## (Intercept)	74.397	< 2e-16 ***
## log(sq.feet)	-110.121	< 2e-16 ***
## as.factor(primary_use)Entertainment/public assembly	-75.563	< 2e-16 ***
## as.factor(primary_use)Lodging/residential	-195.873	< 2e-16 ***
## as.factor(primary_use)Office	-126.632	< 2e-16 ***
## as.factor(primary_use)Public services	-46.812	< 2e-16 ***
## floor.count	179.536	< 2e-16 ***
## air_temperature	-55.022	< 2e-16 ***
## cloud_coverage	38.522	< 2e-16 ***
## dew_temperature	8.027	1.00e-15 ***
## precip_depth_1_hr	-39.494	< 2e-16 ***
## sea_level_pressure	21.994	< 2e-16 ***
## wind_direction	16.027	< 2e-16 ***
## wind_speed	41.331	< 2e-16 ***
## meter	-86.981	< 2e-16 ***
## as.factor(building_id)106	-159.192	< 2e-16 ***
## as.factor(building_id)107	-164.385	< 2e-16 ***
## as.factor(building_id)108	157.818	< 2e-16 ***
## as.factor(building_id)109	44.141	< 2e-16 ***
## as.factor(building_id)110	-94.532	< 2e-16 ***
## as.factor(building_id)111	140.115	< 2e-16 ***
## as.factor(building_id)112	-191.858	< 2e-16 ***
## as.factor(building_id)113	-98.751	< 2e-16 ***
## as.factor(building_id)114	-117.885	< 2e-16 ***
## as.factor(building_id)115	167.141	< 2e-16 ***
## as.factor(building_id)116	-2.577	0.00997 **
## as.factor(building_id)117	-103.861	< 2e-16 ***
## as.factor(building_id)118	94.285	< 2e-16 ***
## as.factor(building_id)119	125.594	< 2e-16 ***
## as.factor(building_id)120	-18.596	< 2e-16 ***
## as.factor(building_id)121	96.092	< 2e-16 ***
## as.factor(building_id)122	138.434	< 2e-16 ***
## as.factor(building_id)123	9.787	< 2e-16 ***
## as.factor(building_id)124	-157.940	< 2e-16 ***
## as.factor(building_id)125	-93.008	< 2e-16 ***
## as.factor(building_id)126	-77.909	< 2e-16 ***
## as.factor(building_id)127	12.586	< 2e-16 ***
## as.factor(building_id)128	160.831	< 2e-16 ***
## as.factor(building_id)129	161.671	< 2e-16 ***
## as.factor(building_id)130	-183.637	< 2e-16 ***
## as.factor(building_id)131	192.450	< 2e-16 ***
## as.factor(building_id)132	126.478	< 2e-16 ***
## as.factor(building_id)133	123.231	< 2e-16 ***
## as.factor(building_id)134	92.920	< 2e-16 ***
## as.factor(building_id)135	-33.539	< 2e-16 ***
## as.factor(building_id)136	NA	NA
## as.factor(building_id)137	NA	NA
## as.factor(building_id)138	96.640	< 2e-16 ***
## as.factor(building_id)139	NA	NA
## as.factor(building_id)140	-245.013	< 2e-16 ***
## as.factor(building_id)141	111.924	< 2e-16 ***
## as.factor(building_id)142	108.933	< 2e-16 ***
## as.factor(building_id)143	125.363	< 2e-16 ***
## as.factor(building_id)144	57.341	< 2e-16 ***
## as.factor(building_id)145	-14.872	< 2e-16 ***
## as.factor(building_id)146	-266.864	< 2e-16 ***
## as.factor(building_id)147	98.767	< 2e-16 ***
## as.factor(building_id)148	22.639	< 2e-16 ***
## as.factor(building_id)149	89.873	< 2e-16 ***
## as.factor(building_id)150	166.196	< 2e-16 ***
## as.factor(building_id)151	-165.983	< 2e-16 ***
## as.factor(building_id)152	-57.132	< 2e-16 ***
## as.factor(building_id)153	NA	NA
## as.factor(building_id)154	NA	NA
## as.factor(building_id)155	NA	NA


```
## as.factor(day%7)1 -29.928 < 2e-16 ***
## as.factor(day%7)2 -43.959 < 2e-16 ***
## as.factor(day%7)3 0.758 0.44833
## as.factor(day%7)4 6.364 1.97e-10 ***
## as.factor(day%7)5 7.321 2.47e-13 ***
## as.factor(day%7)6 5.902 3.60e-09 ***
## as.factor(hour)1 -2.143 0.03212 *
## as.factor(hour)2 -6.225 4.82e-10 ***
## as.factor(hour)3 -8.971 < 2e-16 ***
## as.factor(hour)4 -9.806 < 2e-16 ***
## as.factor(hour)5 -7.576 3.58e-14 ***
## as.factor(hour)6 0.991 0.32146
## as.factor(hour)7 10.378 < 2e-16 ***
## as.factor(hour)8 20.027 < 2e-16 ***
## as.factor(hour)9 32.009 < 2e-16 ***
## as.factor(hour)10 39.279 < 2e-16 ***
## as.factor(hour)11 44.008 < 2e-16 ***
## as.factor(hour)12 47.540 < 2e-16 ***
## as.factor(hour)13 48.033 < 2e-16 ***
## as.factor(hour)14 48.365 < 2e-16 ***
## as.factor(hour)15 47.277 < 2e-16 ***
## as.factor(hour)16 46.170 < 2e-16 ***
## as.factor(hour)17 44.020 < 2e-16 ***
## as.factor(hour)18 39.631 < 2e-16 ***
## as.factor(hour)19 32.941 < 2e-16 ***
## as.factor(hour)20 25.146 < 2e-16 ***
## as.factor(hour)21 18.680 < 2e-16 ***
## as.factor(hour)22 11.907 < 2e-16 ***
## as.factor(hour)23 6.761 1.37e-11 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6621 on 512784 degrees of freedom
## Multiple R-squared: 0.7465, Adjusted R-squared: 0.7464
## F-statistic: 1.735e+04 on 87 and 512784 DF, p-value: < 2.2e-16
```

```
AIC(l17)
```

```
## [1] 1032562
```

Il modello appena elaborato, sicuramente complesso, considerate tutte le variabili a disposizione ottiene un AIC Score significativamente basso, ed un percentuale di varianza espressa pari a 74.64%, quindi un significativo miglioramento del modello, il ch  suggerisce di effettuare un analisi basata sulla storicit  degli immobili.

Si consideri pertanto il seguente modello. :

```
l20=lm(log.meter~as.factor(building_id)+day+hour,data=dati)
summary(l20)
```

```
##
## Call:
## lm(formula = log.meter ~ as.factor(building_id) + day + hour,
##     data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.1934 -0.2264  0.0161  0.2974  5.4387
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.333e+00  7.642e-03  566.934 < 2e-16 ***
## as.factor(building_id)106 -3.065e+00  9.549e-03 -321.017 < 2e-16 ***
## as.factor(building_id)107  2.536e-01  1.032e-02   24.581 < 2e-16 ***
## as.factor(building_id)108  1.142e+00  1.032e-02  110.643 < 2e-16 ***
## as.factor(building_id)109  9.715e-01  9.732e-03   99.823 < 2e-16 ***
## as.factor(building_id)110  1.408e+00  1.032e-02  136.473 < 2e-16 ***
## as.factor(building_id)111  1.694e+00  1.032e-02  164.202 < 2e-16 ***
## as.factor(building_id)112 -1.477e+00  9.506e-03 -155.390 < 2e-16 ***
## as.factor(building_id)113  5.054e-01  8.981e-03   56.267 < 2e-16 ***
## as.factor(building_id)114  1.933e+00  9.253e-03  208.946 < 2e-16 ***
## as.factor(building_id)115  1.873e+00  1.032e-02  181.541 < 2e-16 ***
## as.factor(building_id)116  7.997e-01  1.032e-02   77.512 < 2e-16 ***
## as.factor(building_id)117 -5.367e-01  9.009e-03  -59.572 < 2e-16 ***
## as.factor(building_id)118  1.450e+00  1.032e-02  140.552 < 2e-16 ***
## as.factor(building_id)119  1.440e+00  9.022e-03  159.667 < 2e-16 ***
## as.factor(building_id)120  8.065e-01  1.032e-02   78.176 < 2e-16 ***
## as.factor(building_id)121  1.857e+00  8.993e-03  206.508 < 2e-16 ***
## as.factor(building_id)122  1.308e+00  1.032e-02  126.747 < 2e-16 ***
## as.factor(building_id)123  4.925e-01  1.032e-02   47.738 < 2e-16 ***
## as.factor(building_id)124 -7.724e-01  1.032e-02  -74.864 < 2e-16 ***
## as.factor(building_id)125  7.854e-01  1.032e-02   76.132 < 2e-16 ***
## as.factor(building_id)126  5.033e-01  1.032e-02   48.784 < 2e-16 ***
## as.factor(building_id)127 -1.230e+00  1.032e-02 -119.214 < 2e-16 ***
## as.factor(building_id)128  1.657e-01  1.032e-02   16.057 < 2e-16 ***
## as.factor(building_id)129  1.896e-01  1.032e-02   18.375 < 2e-16 ***
## as.factor(building_id)130 -9.050e-01  1.032e-02  -87.715 < 2e-16 ***
## as.factor(building_id)131  3.986e-01  1.032e-02   38.637 < 2e-16 ***
## as.factor(building_id)132 -3.922e-01  1.032e-02  -38.016 < 2e-16 ***
## as.factor(building_id)133 -3.279e-01  1.032e-02  -31.781 < 2e-16 ***
## as.factor(building_id)134 -2.523e-01  1.032e-02  -24.454 < 2e-16 ***
## as.factor(building_id)135 -5.287e-01  1.032e-02  -51.246 < 2e-16 ***
## as.factor(building_id)136 -6.506e-01  1.032e-02  -63.061 < 2e-16 ***
## as.factor(building_id)137 -5.002e-01  1.032e-02  -48.483 < 2e-16 ***
## as.factor(building_id)138 -3.155e-02  9.268e-03  -3.404 0.000664 ***
## as.factor(building_id)139  9.275e-02  9.640e-03   9.622 < 2e-16 ***
## as.factor(building_id)140 -8.666e-01  1.032e-02  -83.995 < 2e-16 ***
## as.factor(building_id)141 -2.733e-01  1.032e-02  -26.489 < 2e-16 ***
## as.factor(building_id)142 -7.433e-01  1.032e-02  -72.043 < 2e-16 ***
## as.factor(building_id)143 -3.798e-01  1.032e-02  -36.818 < 2e-16 ***
## as.factor(building_id)144 -1.693e+00  9.554e-03 -177.201 < 2e-16 ***
## as.factor(building_id)145 -3.430e-01  9.156e-03  -37.460 < 2e-16 ***
## as.factor(building_id)146 -1.595e+00  1.032e-02 -154.600 < 2e-16 ***
## as.factor(building_id)147  1.388e-01  1.032e-02   13.452 < 2e-16 ***
## as.factor(building_id)148  8.548e-01  1.032e-02   82.855 < 2e-16 ***
## as.factor(building_id)149  4.760e-01  1.032e-02   46.134 < 2e-16 ***
## as.factor(building_id)150  1.727e+00  1.032e-02  167.396 < 2e-16 ***
## as.factor(building_id)151 -4.628e-01  1.032e-02  -44.856 < 2e-16 ***
## as.factor(building_id)152 -1.736e+00  1.032e-02 -168.262 < 2e-16 ***
## as.factor(building_id)153 -1.311e+00  1.032e-02 -127.119 < 2e-16 ***
## as.factor(building_id)154  1.164e+00  1.032e-02  112.838 < 2e-16 ***
## as.factor(building_id)155  6.270e-02  1.032e-02   6.078 1.22e-09 ***
## day                -6.347e-04  8.944e-06  -70.965 < 2e-16 ***
## hour                8.729e-03  1.381e-04   63.223 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6829 on 512819 degrees of freedom
## Multiple R-squared:  0.7302, Adjusted R-squared:  0.7302
## F-statistic: 2.67e+04 on 52 and 512819 DF,  p-value: < 2.2e-16
```

AIC(l20)

[1] 1064316

Il risultato ottenuto dal modello basato sullo storico dei consumi giornalieri ed orari dell'immobile esprime una percentuale di varianza pari al 73.5% ed uno Score AIC di poco peggiore rispetto al modello complessivo in cui si tenevano conto tutte le variabili a disposizione,

Quindi si consideri la seguente espressione :

```
log.meter = 4.226551+bi*building_id) + bj*day +bn*hour
```

Validazione del modello lineare

Si effettua adesso una validazione dei modelli individuati attraverso la tecnica K-fold Cross-Validation

Si utilizza il package “caret” utilizzando i seguenti parametri: si suddivide il dataframe in k=52 fold che rappresenteranno il dataset di training, quindi su 52 fold di dimensione superiore al 75% del dataset originario un fold verrà “addestrato” con il modello scelto.

```
Model_l20=lm(log.meter~as.factor(building_id)+day+hour,data=dati) #modello da validare
control <- trainControl(method = "cv", number = 52) #train
#View(control)
results <- train(log.meter~as.factor(building_id)+day+hour,data=dati, method = "lm", trControl = control)
print(results)
```

```
## Linear Regression
##
## 512872 samples
##      3 predictor
##
## No pre-processing
## Resampling: Cross-Validated (52 fold)
## Summary of sample sizes: 503009, 503009, 503009, 503009, 503010, 503008, ...
## Resampling results:
##
##      RMSE      Rsquared   MAE
## 0.6828597 0.7302573 0.4259166
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

L'errore medio quadratico stimato rispetto ai valori reali è pari al 68%, mentre la percentuale di varianza espressa è pari al 73%. Mentre l'errore medio assoluto è pari al 42%.

Si effettua anche la validazione anche del modello con tutte le variabili.

```
Model_l17=lm(log.meter~log(sq.feet)+as.factor(primary_use)+floor.count+air_temperature+cloud_coverage+dew_temperature+precip_depth_1_hr+sea_level_pressure+wind_direction+wind_speed+meter+as.factor(building_id)+as.factor(day%7)+as.factor(hour),data=dati)
control <- trainControl(method = "cv", number = 52) #train
#View(control)
results <- train(log.meter~log(sq.feet)+as.factor(primary_use)+floor.count+air_temperature+cloud_coverage+dew_temperature+precip_depth_1_hr+sea_level_pressure+wind_direction+wind_speed+meter+as.factor(building_id)+as.factor(day%7)+as.factor(hour), data=dati, method = "lm", trControl = control)
print(results)
```

```
## Linear Regression
##
## 512872 samples
##     14 predictor
##
## No pre-processing
## Resampling: Cross-Validated (52 fold)
## Summary of sample sizes: 503009, 503008, 503008, 503007, 503009, 503009, ...
## Resampling results:
##
##      RMSE      Rsquared   MAE
## 0.6620175 0.7464643 0.4019235
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

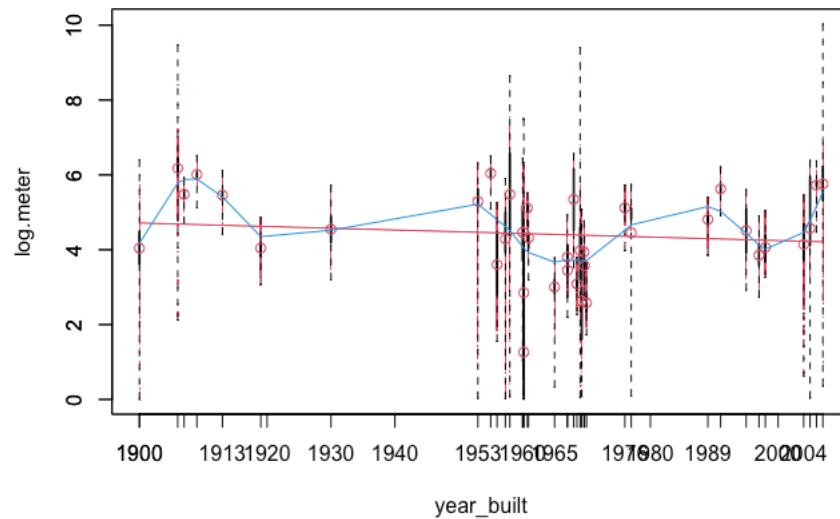
L'errore medio quadratico sui dati reali è pari al 66%, la percentuale di varianza espressa è pari al 74% mentre l'errore medio assoluto è pari al 40%.

Indagini statistiche varie.

Si analizza la relazione tra **log.meter** e **year_built** utilizzando un grafico

```
MLA.boxplot(log.meter,year_built,pch=".",facwidth=6,lines0 = FALSE)
```

Boxplot, regression line, linear regression



```
## $m
##      1900      1906      1907      1909
##      4.042152  6.176059  5.481212  6.011138
##      1913      1919      1930      1953
##      5.458566  4.043773  4.547722  5.292544
##      1955      1956  1957.29384696099  1958
##      6.044086  3.602390  4.293645  5.480861
##      1960  1960.17153742397  1960.17644711509  1960.74158592168
##      4.466631  1.263335  2.852572  5.116749
## 1960.92149019334  1965  1967  1967.01362832237
##      4.317062  3.005569  3.802608  3.450486
##      1968  1968.50082104721  1968.9979672967  1969.25314627075
##      5.348329  3.087130  3.977365  2.632409
## 1969.60659205267  1969.67903720893  1970  1976
##      3.937213  3.573796  2.580988  5.123604
##      1977      1989      1991      1995
##      4.455845  4.809576  5.624716  4.513681
##      1997      1998      2004      2005
##      3.854279  4.064769  4.140219  4.570665
##      2006      2007
##      5.725053  5.758838
##
## $x
## [1] 1900.000 1906.000 1907.000 1909.000 1913.000 1919.000 1930.000 1953.000
## [9] 1955.000 1956.000 1957.294 1958.000 1960.000 1960.172 1960.176 1960.742
## [17] 1960.921 1965.000 1967.000 1967.014 1968.000 1968.501 1968.998 1969.253
## [25] 1969.607 1969.679 1970.000 1976.000 1977.000 1989.000 1991.000 1995.000
## [33] 1997.000 1998.000 2004.000 2005.000 2006.000 2007.000
```

Si stima adesso il modello lineare considerando la variabile year_built dipendente dalla variabile sq.feet in quanto è necessario tenere conto anche dell'aspetto principale emerso ovvero la dimensione degli edifici rappresentata dalla variabile sq.feet.

```
l21=lm(log.meter~as.factor(year_built)/log(sq.feet)+as.factor(day%7)+hour,data=dati)
summary(l21)
```

```
##
## Call:
## lm(formula = log.meter ~ as.factor(year_built)/log(sq.feet) +
##     as.factor(day%7) + hour, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.2429 -0.2308  0.0265  0.3078  5.3708
##
## Coefficients: (29 not defined because of singularities)
##                                Estimate Std. Error
## (Intercept)                   -1.997e+01  2.707e-01
## as.factor(year_built)1906       2.607e+01  2.707e-01
## as.factor(year_built)1907       2.538e+01  2.708e-01
## as.factor(year_built)1909       2.591e+01  2.708e-01
## as.factor(year_built)1913       2.536e+01  2.708e-01
## as.factor(year_built)1919       2.394e+01  2.708e-01
## as.factor(year_built)1930       9.947e+00  5.278e-01
## as.factor(year_built)1953       2.519e+01  2.708e-01
## as.factor(year_built)1955       2.594e+01  2.708e-01
## as.factor(year_built)1956       7.146e+00  2.878e-01
## as.factor(year_built)1957.29384696099 2.419e+01  2.707e-01
## as.factor(year_built)1958      -2.545e+01  3.886e-01
## as.factor(year_built)1960       4.360e+00  2.826e-01
## as.factor(year_built)1960.17153742397 2.116e+01  2.708e-01
## as.factor(year_built)1960.17644711509 2.275e+01  2.708e-01
## as.factor(year_built)1960.74158592168 2.501e+01  2.708e-01
## as.factor(year_built)1960.92149019334 2.422e+01  2.708e-01
## as.factor(year_built)1965       2.290e+01  2.708e-01
## as.factor(year_built)1967       3.193e+01  3.041e+00
## as.factor(year_built)1967.01362832237 2.335e+01  2.708e-01
## as.factor(year_built)1968      -5.969e+01  5.948e-01
## as.factor(year_built)1968.50082104721 2.299e+01  2.708e-01
## as.factor(year_built)1968.9979672967 2.387e+01  2.707e-01
## as.factor(year_built)1969.25314627075 2.253e+01  2.708e-01
## as.factor(year_built)1969.60659205267 2.384e+01  2.708e-01
## as.factor(year_built)1969.67903720893 2.347e+01  2.708e-01
## as.factor(year_built)1970       2.248e+01  2.708e-01
## as.factor(year_built)1976       2.502e+01  2.708e-01
## as.factor(year_built)1977       2.435e+01  2.708e-01
## as.factor(year_built)1989       2.471e+01  2.708e-01
## as.factor(year_built)1991       2.552e+01  2.708e-01
## as.factor(year_built)1995       3.217e+01  2.794e-01
## as.factor(year_built)1997       2.375e+01  2.708e-01
## as.factor(year_built)1998       2.396e+01  2.708e-01
## as.factor(year_built)2004      -6.675e+00  3.833e-01
## as.factor(year_built)2005       2.447e+01  2.708e-01
## as.factor(year_built)2006       2.562e+01  2.708e-01
## as.factor(year_built)2007       2.566e+01  2.707e-01
## as.factor(day%7)1              -1.045e-01  3.604e-03
## as.factor(day%7)2              -1.516e-01  3.625e-03
## as.factor(day%7)3              -3.998e-03  3.618e-03
## as.factor(day%7)4               1.793e-02  3.614e-03
## as.factor(day%7)5               1.921e-02  3.618e-03
## as.factor(day%7)6               1.534e-02  3.619e-03
## hour                           8.706e-03  1.405e-04
## as.factor(year_built)1900: log(sq.feet) 2.221e+00 2.511e-02
## as.factor(year_built)1906: log(sq.feet)      NA      NA
## as.factor(year_built)1907: log(sq.feet)      NA      NA
## as.factor(year_built)1909: log(sq.feet)      NA      NA
## as.factor(year_built)1913: log(sq.feet)      NA      NA
## as.factor(year_built)1919: log(sq.feet)      NA      NA
## as.factor(year_built)1930: log(sq.feet)      1.320e+00 4.126e-02
## as.factor(year_built)1953: log(sq.feet)      NA      NA
## as.factor(year_built)1955: log(sq.feet)      NA      NA
## as.factor(year_built)1956: log(sq.feet)      1.493e+00 8.904e-03
## as.factor(year_built)1957.29384696099: log(sq.feet) NA      NA
## as.factor(year_built)1958: log(sq.feet)      4.356e+00 2.389e-02
## as.factor(year_built)1960: log(sq.feet)      1.758e+00 7.132e-03
## as.factor(year_built)1960.17153742397: log(sq.feet) NA      NA
## as.factor(year_built)1960.17644711509: log(sq.feet) NA      NA
## as.factor(year_built)1960.74158592168: log(sq.feet) NA      NA
## as.factor(year_built)1960.92149019334: log(sq.feet) NA      NA
## as.factor(year_built)1965: log(sq.feet)      NA      NA
## as.factor(year_built)1967: log(sq.feet)     -7.419e-01 2.732e-01
## as.factor(year_built)1967.01362832237: log(sq.feet) NA      NA
## as.factor(year_built)1968: log(sq.feet)      7.286e+00 4.543e-02
## as.factor(year_built)1968.50082104721: log(sq.feet) NA      NA
## as.factor(year_built)1968.9979672967: log(sq.feet) NA      NA
## as.factor(year_built)1969.25314627075: log(sq.feet) NA      NA
## as.factor(year_built)1969.60659205267: log(sq.feet) NA      NA
## as.factor(year_built)1969.67903720893: log(sq.feet) NA      NA
## as.factor(year_built)1970: log(sq.feet)      NA      NA
## as.factor(year_built)1976: log(sq.feet)      NA      NA
## as.factor(year_built)1977: log(sq.feet)      NA      NA
## as.factor(year_built)1989: log(sq.feet)      NA      NA
## as.factor(year_built)1991: log(sq.feet)      NA      NA
```

## as.factor(year_built)1995: log(sq. feet)	-7.367e-01	6.566e-03
## as.factor(year_built)1997: log(sq. feet)	NA	NA
## as.factor(year_built)1998: log(sq. feet)	NA	NA
## as.factor(year_built)2004: log(sq. feet)	3.146e+00	2.780e-02
## as.factor(year_built)2005: log(sq. feet)	NA	NA
## as.factor(year_built)2006: log(sq. feet)	NA	NA
## as.factor(year_built)2007: log(sq. feet)	NA	NA
##	t value	Pr(> t)
## (Intercept)	-73.766	< 2e-16 ***
## as.factor(year_built)1906	96.304	< 2e-16 ***
## as.factor(year_built)1907	93.726	< 2e-16 ***
## as.factor(year_built)1909	95.683	< 2e-16 ***
## as.factor(year_built)1913	93.643	< 2e-16 ***
## as.factor(year_built)1919	88.418	< 2e-16 ***
## as.factor(year_built)1930	18.846	< 2e-16 ***
## as.factor(year_built)1953	93.025	< 2e-16 ***
## as.factor(year_built)1955	95.805	< 2e-16 ***
## as.factor(year_built)1956	24.833	< 2e-16 ***
## as.factor(year_built)1957.29384696099	89.350	< 2e-16 ***
## as.factor(year_built)1958	-65.487	< 2e-16 ***
## as.factor(year_built)1960	15.429	< 2e-16 ***
## as.factor(year_built)1960.17153742397	78.142	< 2e-16 ***
## as.factor(year_built)1960.17644711509	84.016	< 2e-16 ***
## as.factor(year_built)1960.74158592168	92.380	< 2e-16 ***
## as.factor(year_built)1960.92149019334	89.427	< 2e-16 ***
## as.factor(year_built)1965	84.584	< 2e-16 ***
## as.factor(year_built)1967	10.500	< 2e-16 ***
## as.factor(year_built)1967.01362832237	86.227	< 2e-16 ***
## as.factor(year_built)1968	-100.356	< 2e-16 ***
## as.factor(year_built)1968.50082104721	84.885	< 2e-16 ***
## as.factor(year_built)1968.9979672967	88.185	< 2e-16 ***
## as.factor(year_built)1969.25314627075	83.218	< 2e-16 ***
## as.factor(year_built)1969.60659205267	88.024	< 2e-16 ***
## as.factor(year_built)1969.67903720893	86.682	< 2e-16 ***
## as.factor(year_built)1970	83.016	< 2e-16 ***
## as.factor(year_built)1976	92.406	< 2e-16 ***
## as.factor(year_built)1977	89.940	< 2e-16 ***
## as.factor(year_built)1989	91.246	< 2e-16 ***
## as.factor(year_built)1991	94.256	< 2e-16 ***
## as.factor(year_built)1995	115.125	< 2e-16 ***
## as.factor(year_built)1997	87.718	< 2e-16 ***
## as.factor(year_built)1998	88.495	< 2e-16 ***
## as.factor(year_built)2004	-17.415	< 2e-16 ***
## as.factor(year_built)2005	90.364	< 2e-16 ***
## as.factor(year_built)2006	94.627	< 2e-16 ***
## as.factor(year_built)2007	94.766	< 2e-16 ***
## as.factor(day%7)1	-28.992	< 2e-16 ***
## as.factor(day%7)2	-41.812	< 2e-16 ***
## as.factor(day%7)3	-1.105	0.26917
## as.factor(day%7)4	4.960	7.06e-07 ***
## as.factor(day%7)5	5.310	1.10e-07 ***
## as.factor(day%7)6	4.240	2.23e-05 ***
## hour	61.977	< 2e-16 ***
## as.factor(year_built)1900: log(sq. feet)	88.457	< 2e-16 ***
## as.factor(year_built)1906: log(sq. feet)	NA	NA
## as.factor(year_built)1907: log(sq. feet)	NA	NA
## as.factor(year_built)1909: log(sq. feet)	NA	NA
## as.factor(year_built)1913: log(sq. feet)	NA	NA
## as.factor(year_built)1919: log(sq. feet)	NA	NA
## as.factor(year_built)1930: log(sq. feet)	32.002	< 2e-16 ***
## as.factor(year_built)1953: log(sq. feet)	NA	NA
## as.factor(year_built)1955: log(sq. feet)	NA	NA
## as.factor(year_built)1956: log(sq. feet)	167.735	< 2e-16 ***
## as.factor(year_built)1957.29384696099: log(sq. feet)	NA	NA
## as.factor(year_built)1958: log(sq. feet)	182.309	< 2e-16 ***
## as.factor(year_built)1960: log(sq. feet)	246.446	< 2e-16 ***
## as.factor(year_built)1960.17153742397: log(sq. feet)	NA	NA
## as.factor(year_built)1960.17644711509: log(sq. feet)	NA	NA
## as.factor(year_built)1960.74158592168: log(sq. feet)	NA	NA
## as.factor(year_built)1960.92149019334: log(sq. feet)	NA	NA
## as.factor(year_built)1965: log(sq. feet)	NA	NA
## as.factor(year_built)1967: log(sq. feet)	-2.716	0.00661 **
## as.factor(year_built)1967.01362832237: log(sq. feet)	NA	NA
## as.factor(year_built)1968: log(sq. feet)	160.373	< 2e-16 ***
## as.factor(year_built)1968.50082104721: log(sq. feet)	NA	NA
## as.factor(year_built)1968.9979672967: log(sq. feet)	NA	NA
## as.factor(year_built)1969.25314627075: log(sq. feet)	NA	NA
## as.factor(year_built)1969.60659205267: log(sq. feet)	NA	NA
## as.factor(year_built)1969.67903720893: log(sq. feet)	NA	NA
## as.factor(year_built)1970: log(sq. feet)	NA	NA
## as.factor(year_built)1976: log(sq. feet)	NA	NA
## as.factor(year_built)1977: log(sq. feet)	NA	NA
## as.factor(year_built)1989: log(sq. feet)	NA	NA
## as.factor(year_built)1991: log(sq. feet)	NA	NA
## as.factor(year_built)1995: log(sq. feet)	-112.193	< 2e-16 ***
## as.factor(year_built)1997: log(sq. feet)	NA	NA
## as.factor(year_built)1998: log(sq. feet)	NA	NA

```
## as.factor(year_built)2004:log(sq.feet)      113.184 < 2e-16 ***
## as.factor(year_built)2005:log(sq.feet)      NA      NA
## as.factor(year_built)2006:log(sq.feet)      NA      NA
## as.factor(year_built)2007:log(sq.feet)      NA      NA
## ----
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6948 on 512818 degrees of freedom
## Multiple R-squared:  0.7208, Adjusted R-squared:  0.7207
## F-statistic: 2.498e+04 on 53 and 512818 DF, p-value: < 2.2e-16
```

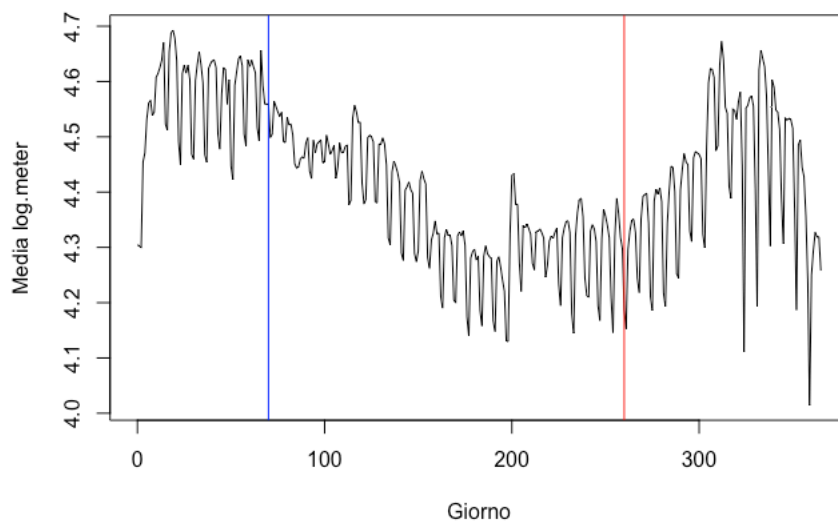
```
AIC(l21)
```

```
## [1] 1082029
```

Sembra che tendenzialmente gli edifici più recenti abbiano un consumo più basso.

Si osservino adesso la media dei consumi giornalieri

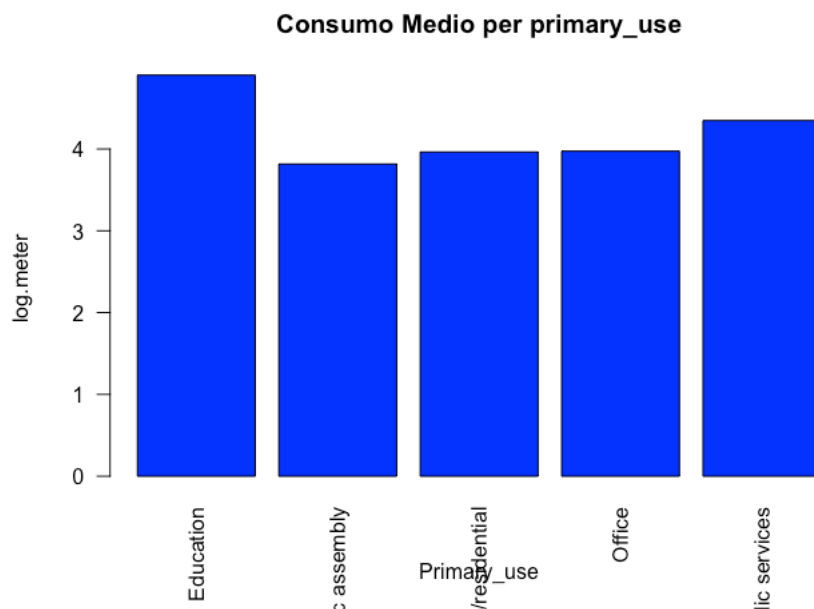
```
daily_mean <- aggregate(log.meter ~ day, data = dati, mean)
plot(daily_mean$day, daily_mean$log.meter, type = "l", xlab = "Giorno", ylab = "Media log.meter")
abline(v = 70, col = "blue", lty = 1) # Linea verticale in blu, con un trattino
abline(v = 260, col = "red", lty = 1) # Linea verticale in blu, con un trattino
```



Si nota una sensibile riduzione dei consumi nei giorni che vanno da marzo a fine maggio, quindi restano stabili fino a settembre in cui riprendono a salire.

Si analizza il consumo medio aggregato per primary_use

```
primary_use_mean <- aggregate(log.meter ~ primary_use, data = dati, mean)
barplot(primary_use_mean$log.meter, names.arg = primary_use_mean$primary_use,
        xlab = "Primary_use", ylab = "log.meter", col = "blue",
        main = "Consumo Medio per primary_use", las=2)
```

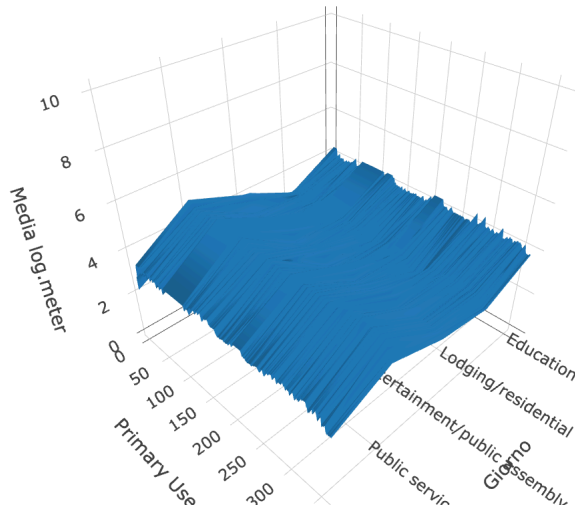


```
# Calcola la media dei consumi giornalieri raggruppati per "day" e "primary_use"
daily_mean <- aggregate(log.meter ~ day + as.factor(primary_use), data = dati, mean)

# Crea un grafico tridimensionale
plot_3d <- plot_ly(data = datisub, x = ~primary_use, y = ~day, z = ~log.meter, type = "mesh3d")

# Personalizza l'aspetto del grafico
plot_3d <- plot_3d %>%
  layout(scene = list(xaxis = list(title = "Giorno"),
                      yaxis = list(title = "Primary Use"),
                      zaxis = list(title = "Media log.meter")))

# Visualizza il grafico
plot_3d
```



Conclusioni

In definitiva dalle analisi effettuate sul dataset fornito, e considerato il focus principale dello studio, ovvero la comprensione di come la variabile **log.meter** dipendesse dalle altre variabili è possibile affermare che senza alcun dubbio la variabile target: **log.meter** dipende linearmente dalla dimensione dell'edificio, **sq.feet** la quale è molto correlata positivamente con il numero di piani dell'edificio **floor.count**. Tuttavia la conoscenza di queste variabili indipendenti non è sufficiente a stimare in maniera apprezzabile la variabile log.meter, poichè la percentuale di varianza espressa non è soddisfacente.

Un significativo miglioramento si ottiene utilizzando tutti i predittori a disposizione, tuttavia risulta un modello sufficientemente complesso, pertanto le ulteriori esplorazioni sulle variabili hanno portato alla definizione di un modello basato sulla storicità del building.

infatti, il modello ottimale è rappresentato dalla seguente espressione: $\text{Mod_l20} = \text{lm}(\text{log.meter} \sim \text{as.factor(building_id)} + \text{day} + \text{hour}, \text{data} = \text{dati})$

ovvero : $\text{log.meter} = 4.226551 + b_i * \text{building_id} + b_j * \text{day} + b_n * \text{hour}$

Questo modello consente quindi di stimare con determinata approssimazione, il valore futuro di log.meter essendo noti il building, il giorno e l'ora.

Si è anche visto che tendenzialmente gli edifici più recenti hanno un consumo medio più basso, quindi risultano più efficienti in termini energetici.

Il consumo medio giornaliero aggregato invece, fa evidenziare un tendenziale comportamento nel tempo, ovvero nei mesi che vanno da marzo a fine maggio si riducono, quindi restano stabili fino a settembre in cui riprendono a salire.

Inoltre, tra tutte le categorie di immobili osservate, la categoria che ha il maggior consumo di energia elettrica è pari agli edifici di tipo "Education".