

Esercizio 1 - Analisi dei furti di automobili a Chicago

Uberto Vittorio Favero^{1,2} and Ludovico Loreti^{1,3}

¹Università di Bologna

²Statistica Numerica

6 febbraio 2017

1 Introduzione

In questa relazione esamineremo il file reso disponibile dal portale della città di Chicago dal 2001 inerente i furti di autoveicoli così da descrivere il campione di dati, analizzarlo formandone un modello.

Il datasource è liberamente scaricabile a: <https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2>

Ed è composto da:

- Data e ora di rilievo del reato
- Indirizzo
- Tipologia di reato
- Descrizione del tipo di veicolo
- Descrizione del luogo del furto (Strada, parcheggio privato, supermercato etc)
- Ward, ovvero il distretto di polizia di riferimento
- Community Area, ovvero il quartiere
- Anno
- Latitudine e longitudine

`kruskal.test` help page into a \LaTeX document:

2 Preparazione dei dati

Il dati sono disponibili per il download in un file .csv di circa 200k record.

Innanzitutto valuto il file con la funzione scan per comprenderne la conformazione:

```
library(ggplot2)
library(stringr)
library(data.table)
library(ggmap)
library(dplyr)

## -----
## data.table + dplyr code now lives in dtplyr.
## Please library(dtplyr)!
## -----
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:data.table':
##
##   between, first, last
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

datiChicago <- read.csv('~/Desktop/Statistica /ProbabilityAndStatisticsUsingR/dataset/motor_vehicle_theft.csv',
                        stringsAsFactors = FALSE)

str(datiChicago)

## 'data.frame': 295225 obs. of  9 variables:
##  $ Date           : chr  "03/20/2006 11:00:00 PM" "03/15/2006 03:00:00 PM" "03/22/2006 03:00:00 PM" ...
##  $ Block          : chr  "064XX S WOLCOTT AVE" "092XX S LANGLEY AVE" "0000X E 118TH" ...
##  $ Primary.Type    : chr  "MOTOR VEHICLE THEFT" "MOTOR VEHICLE THEFT" "MOTOR VEHICLE THEFT" ...
##  $ Description     : chr  "AUTOMOBILE" "AUTOMOBILE" "THEFT/RECOVERY: AUTOMOBILE" "AUTOMOBILE" ...
##  $ Location.Description: chr  "STREET" "STREET" "STREET" "STREET" ...
##  $ Ward           : int   15  9  9  6  2  43  47  15  4  15 ...
##  $ Community.Area  : int   67  44  53  69  28  8  3  67  38  67 ...
##  $ Year            : int   2006 2006 2006 2006 2006 2005 2006 2006 2006 2006 ...
##  $ Location        : chr   "(41.776711023, -87.671429342)" "(41.72632605, -87.60699981)" ...
```

Ora che ne ho verificato la conformazione posso passare alla valutazione di eventuali dati mancanti

3 Analisi quantitativa dei dati

Al fine di fare un'analisi quantitativa dei dati, decido di guardare in generale l'andamento dei furti in Chicago per anno. Quindi appoggiandomi al pacchetto `dplyr` raggruppo il file per anno e calcolo i totali:

```
totAnnuo <- group_by(datiChicago, Year, Description)

totsPerCateg <- summarize(totAnnuo, count=n())

str(totsPerCateg)

## Classes 'grouped_df', 'tbl_df', 'tbl' and 'data.frame': 174 obs. of  3 variables:
## $ Year      : int  2001 2001 2001 2001 2001 2001 2001 2001 2001 2001 2002 ...
## $ Description: chr  "ATT: AUTOMOBILE" "ATT: TRUCK, BUS, MOTOR HOME" "ATTEMPT: CYCLE, SC
## $ count      : int  1439 56 3 20692 220 3228 11 274 1626 1092 ...
## - attr(*, "vars")=List of 1
## ..$ : symbol Year
## - attr(*, "drop")= logi TRUE

#aggrego il dataframe precedente per anno sommandone le categorie -

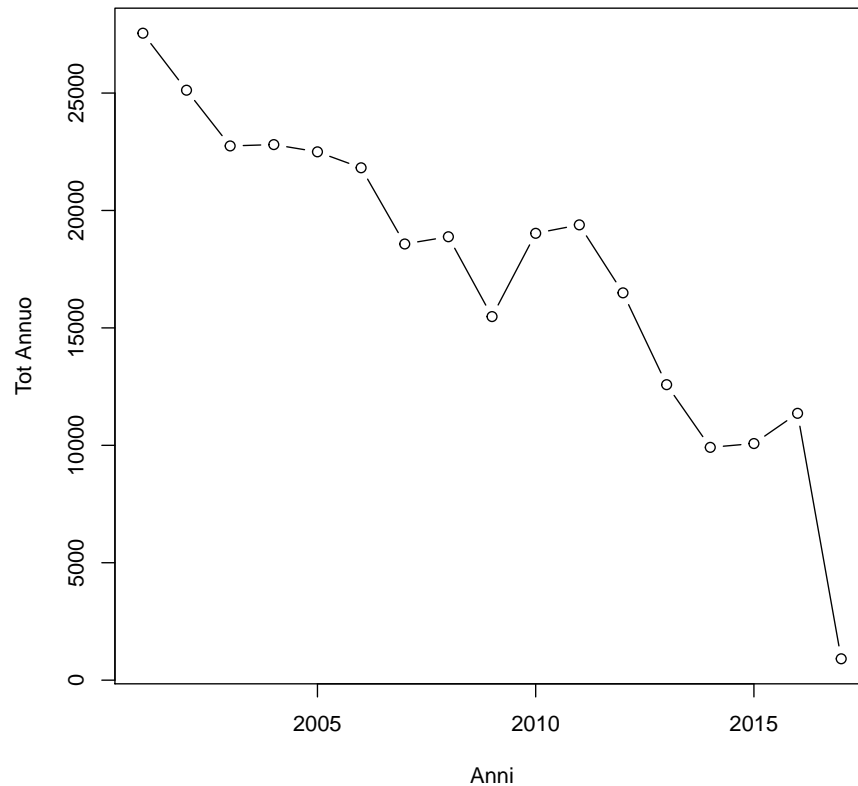
totPerAnno <- aggregate(totsPerCateg$count, by = list(Year=totsPerCateg$Year), FUN = sum)

#rinomino colonna
names(totPerAnno)[names(totPerAnno) == 'x'] <- 'Totale.annuo'

# xlab - ylab = labels. type = b both (linee e punti)

plot(totPerAnno, ylab = "Tot Annuo", xlab = "Anni",
      main = "Furti totali di veicoli a Chicago dal 2001 a oggi ", type = 'b')
```

Furti totali di veicoli a Chicago dal 2001 a oggi



```
print(summary(totPerAnno))
```

```
##      Year      Totale.annuo
##  Min.   :2001   Min.    :  910
## 1st Qu.:2005   1st Qu.:12582
## Median :2009   Median :18881
## Mean   :2009   Mean    :17366
## 3rd Qu.:2013   3rd Qu.:22497
## Max.   :2017   Max.    :27549
```

Analizzando i dati appena stampanti si nota come ci sia un generale calo dei furti. Il dato che fa sorridere è il picco del 2017. Picco comprensibile in quanto l'anno non è ancora terminato.

Sarebbe interessante nei paragrafi successivi stimare i furti di veicoli per quest'anno.

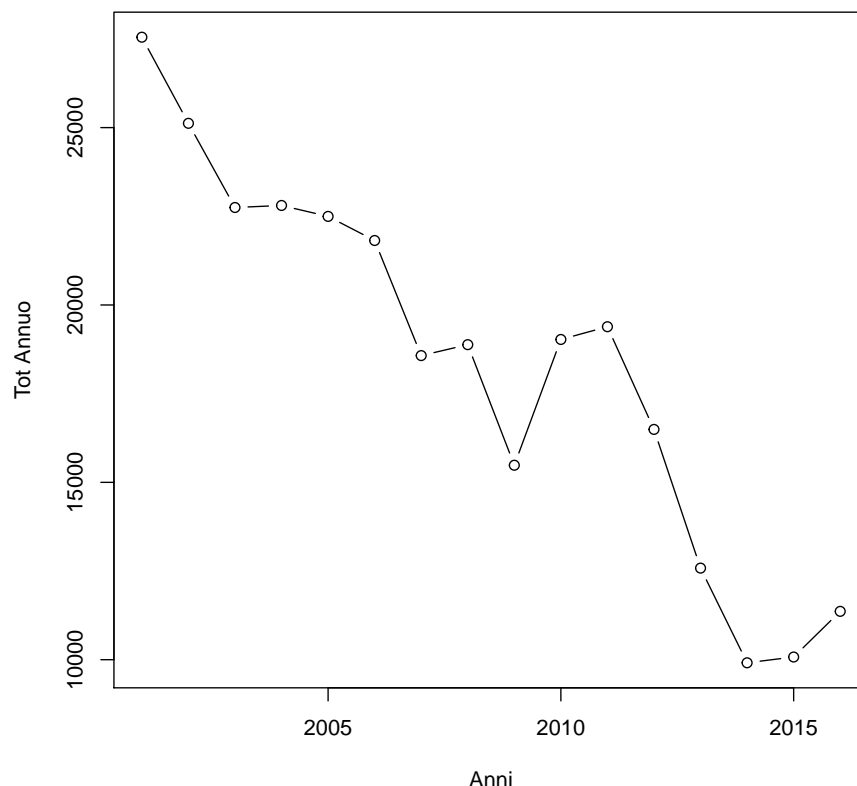
A questo punto il 2017 lo tratto come un outlier e per ora non lo prendo in considerazione, aggiornando i dataset rimuovendo i dati inerenti il 2017, e ricalcolo il **summary**, visualizzando anche mediana, massimo e minimo

```
totsPerCateg <- totsPerCateg[!(totsPerCateg$Year == '2017'),]  
totPerAnno <- totPerAnno[!(totPerAnno$Year == '2017'),]  
  
knitr::kable(totPerAnno)
```

Year	Totale.annuo
2001	27549
2002	25121
2003	22748
2004	22805
2005	22497
2006	21818
2007	18573
2008	18881
2009	15482
2010	19028
2011	19387
2012	16492
2013	12582
2014	9913
2015	10076
2016	11363

```
plot(totPerAnno, ylab = "Tot Annuo", xlab = "Anni",  
      main = "Furti totali di veicoli a Chicago dal 2001 a oggi ", type = 'b')
```

Furti totali di veicoli a Chicago dal 2001 a oggi



Ora, per studiare i dati in maniera più consona, calcolo le somme mensili per ogni tipologia di furto. Ma per fare questo per evitare errori sui formati di data, converto la colonna **Date** secondo lo standard POSIXct - mentre ora sono salvati in POSIXlt. Successivamente creo una tabella con le righe contenenti sulla prima colonna una sequenza di numeri dal 2001 al 2017 ordinati in maniera crescente divisi separati da un intervallo di 1/12 e calcolo per ogni mese i totali per ciascuno delle tipologie di furto indicate. Per terminare salvo il csv così da poterlo riutilizzare successivamente

```
datiChicago$Date <- as.POSIXct(datiChicago$Date, format="%Y%m%d %H%M%S")

#creo una tabella vuota dove la prima colonna sono gli anni dal 2006 a oggi in forma numerica 2006, 2006.08333-...
monthlyByCateg <- data.frame(Months =seq(as.Date("2001-01-01"),
                                         as.Date("2017-01-01"), by="months" ))
```

```

#monthlyByCateg <- data.frame(Months = seq(from = 2001, to = 2017, by= 1/12 ))
descrizioni <- as.character(unique(unlist(datiChicago$Description)))
#genero la matrice con dati vuoti
monthlyByCateg[,descrizioni] <-NA

anno = 2001
mese = 1
rowsTot = nrow(monthlyByCateg)
categTot <- length(descrizioni)
#descrizione vale come indice colonna per la matrice monthlyByCa-
teg

for(riga in 1: rowsTot){
  #so fare due nested for, ma mi serviva così. ci mette 3 minuti buo-
  ni a calcolare tutto
  monthlyByCateg[riga, 2] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[1] & month(Dat
monthlyByCateg[riga, 3] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[2] & month(Dat
monthlyByCateg[riga, 4] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[3] & month(Dat
monthlyByCateg[riga, 5] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[4] & month(Dat
monthlyByCateg[riga, 6] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[5] & month(Dat
monthlyByCateg[riga, 7] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[6] & month(Dat
monthlyByCateg[riga, 8] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[7] & month(Dat
monthlyByCateg[riga, 9] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[8] & month(Dat
monthlyByCateg[riga, 10] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[9] & month(Da
monthlyByCateg[riga, 11] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[10] & month(D
monthlyByCateg[riga, 12] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[11] & month(D
monthlyByCateg[riga, 13] <- summarize(subset(datiChicago, Year == anno
                                                & Description == descrizioni[12] & month(D

  mese = mese+1
  if(riga %% 12 == 0){
    anno = anno +1
    mese = 1
  }
}

```

```

}

write.csv(monthlyByCateg, file = "monthlyByCateg.csv")
#rinomino colonna
names(monthlyByCateg)[names(monthlyByCateg) == 'Months'] <- 'time'

knitr::kable(summary(monthlyByCateg))

```

time	AUTOMOBILE	THEFT/RECOVERY: AUTOMOBILE	TRUCK, BUS, MOTOR HOME
Min. :2001-01-01	Min. :0	Min. :0	Min. :0
1st Qu.:2005-01-01	1st Qu.:0	1st Qu.:0	1st Qu.:0
Median :2009-01-01	Median :0	Median :0	Median :0
Mean :2008-12-30	Mean :0	Mean :0	Mean :0
3rd Qu.:2013-01-01	3rd Qu.:0	3rd Qu.:0	3rd Qu.:0
Max. :2017-01-01	Max. :0	Max. :0	Max. :0

A questo punto posso creare una funzione che mi gestisce le varie colonne e mi stampa un summary

```

analisiDescr <- function(file, colStart, colEnd){
  dataset <- file[,colStart:colEnd]
  dim <- length(dataset)
  #2 significa colonne, #1 righe
  media <- apply(dataset, 2, mean, na.rm = TRUE)
  varianza <- apply(dataset, 2, var, na.rm = TRUE)
  devStd <- sqrt(varianza)
  minimo <- apply(dataset, 2, min, na.rm = TRUE)
  massimo <- apply(dataset, 2, max, na.rm = TRUE)
  Desc <- data.frame(media, varianza, devStd, minimo, massimo)
  colnames <- c("media", "varianza", "devStd", "minimo", "massimo")
  return(Desc)
}

```

```

dati <- analisiDescr(monthlyByCateg, 2, 13)
dati

```

```

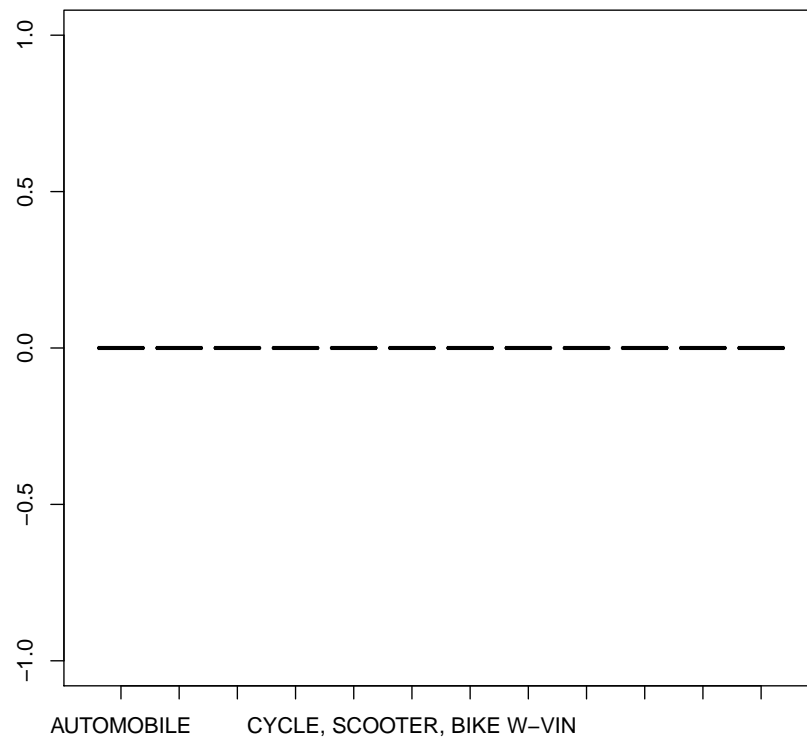
##                media varianza devStd minimo
## AUTOMOBILE          0         0      0      0
## THEFT/RECOVERY: AUTOMOBILE          0         0      0      0
## TRUCK, BUS, MOTOR HOME          0         0      0      0
## ATT: AUTOMOBILE          0         0      0      0
## THEFT/RECOVERY: TRUCK,BUS,MHOME          0         0      0      0
## CYCLE, SCOOTER, BIKE W-VIN          0         0      0      0
## ATT: TRUCK, BUS, MOTOR HOME          0         0      0      0
## THEFT/RECOVERY: CYCLE, SCOOTER, BIKE NO VIN          0         0      0      0
## ATTEMPT: CYCLE, SCOOTER, BIKE W-VIN          0         0      0      0
## THEFT/RECOVERY: CYCLE, SCOOTER, BIKE W-VIN          0         0      0      0

```


## CYCLE, SCOOTER, BIKE NO VIN	0	0	0	0
## ATTEMPT: CYCLE, SCOOTER, BIKE NO VIN	0	0	0	0
##	massimo			
## AUTOMOBILE	0			
## THEFT/RECOVERY: AUTOMOBILE	0			
## TRUCK, BUS, MOTOR HOME	0			
## ATT: AUTOMOBILE	0			
## THEFT/RECOVERY: TRUCK,BUS,MHOME	0			
## CYCLE, SCOOTER, BIKE W-VIN	0			
## ATT: TRUCK, BUS, MOTOR HOME	0			
## THEFT/RECOVERY: CYCLE, SCOOTER, BIKE NO VIN	0			
## ATTEMPT: CYCLE, SCOOTER, BIKE W-VIN	0			
## THEFT/RECOVERY: CYCLE, SCOOTER, BIKE W-VIN	0			
## CYCLE, SCOOTER, BIKE NO VIN	0			
## ATTEMPT: CYCLE, SCOOTER, BIKE NO VIN	0			

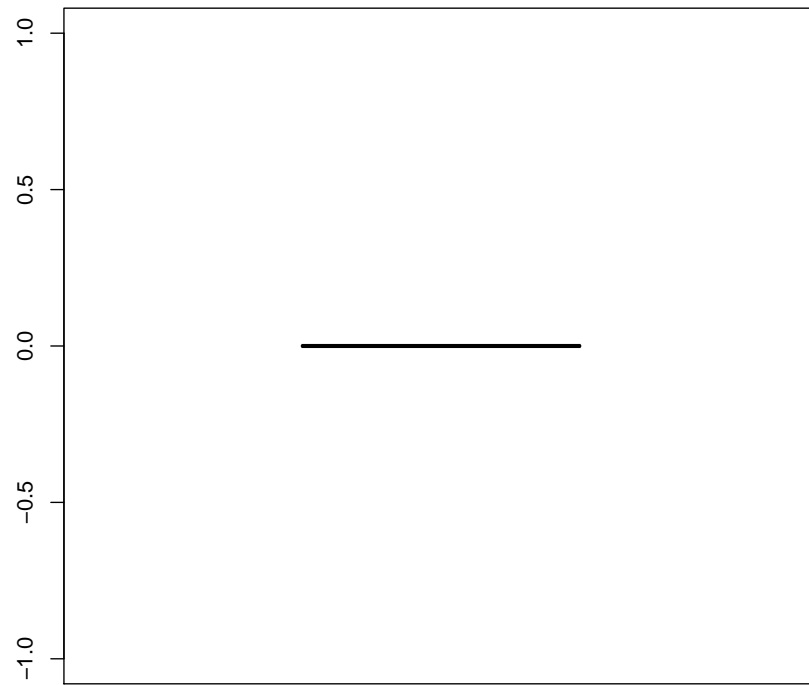
Dall'analisi dei dati notiamo che i dati sui furti di automobili (come quelli di mezzi pesanti) variano notevolmente. Studio quindi la presenza di outliers. Piccola osservazione: se vi rubano un mezzo a Chicago non sperate di recuperarlo, sciocchi.

```
furti <- boxplot(x= as.list(monthlyByCateg[,2:13]))
```

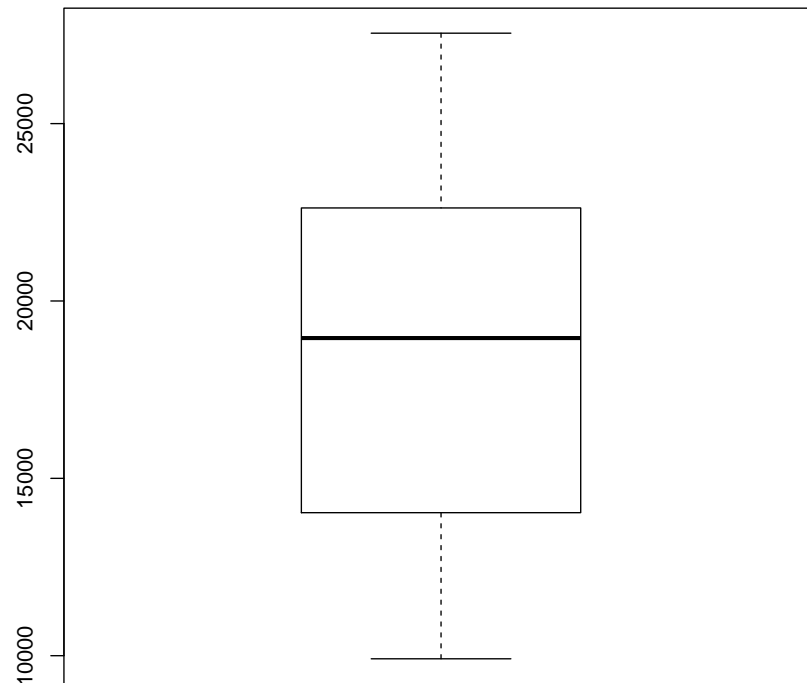


Come si evince dal grafico ci sono degli outliers potenziali soprattutto nei grafici 1, 2, 4 e 6 che, mi accingo a rimuovere tramite identify, che fornisce la possibilità di intervenire direttamente col mouse sul grafico.

```
furtiAuto <- boxplot(x= monthlyByCateg$AUTOMOBILE)
darimuovere <- identify(rep(3, length(monthlyByCateg[,2]))+ monthlyByCateg[2])
```



```
boxplot( totPerAnno$Totale.annuo)
```



Sarò sincero... la funzione "identify" mi ha sempre risposto **warning: no point within 0.25 inches** nonostante cliccassi correttamente sul punto outlier segnalato, al che ho proceduto alla rimozione manuale dell'outlier, corrispondente alla misurazione su Ottobre 2001.

Ora studiamo i file come time-series. Il formato TimeSerie permette tutta una serie di analisi sul trend e sulle previsioni grazie alle funzionalità built-in di R. Pertanto converto il dataset tramite la funzione in una serie temporale che ha una ciclicità di 12 mesi. Successivamente applico la funzione decompose che mi permette di studiare la serie temporale stessa.

```
timeSerie <- ts(select(monthlyByCateg, AUTOMOBILE), frequency = 12, start = c(2001,1) )
timeSerieDec <- decompose(timeSerie, type = "multiplicative")
plot(timeSerieDec)

## Warning in min(x): no non-missing arguments to min; returning Inf
## Warning in max(x): no non-missing arguments to max; returning -Inf
## Error in plot.window(...): i valori di 'ylim' devono essere finiti
```



Questo grafico, tramite l'analisi di una serie temporale (`timeserie`) mostra:

- Andamento dell'osservazione
- Il trend, che abbastanza assimilabile a un processo lineare
- Stagionalità. Come vediamo c'è una certa ripetitività nel fenomeno e quindi sì, il fenomeno è stagionale (o meglio, nel grafico è annuale)
- Random, ovvero la stima dei dati che non seguono l'andamento (componenti irregolari)

Da una `timeserie` e il relativo andamento risulta abbastanza agevole lavorare sulle stime per la creazione di un modello Utilizzando una previsione di Holt,

```
serieFurti <- HoltWinters(timeserie, beta = FALSE, gamma = FALSE)
serieFurti$fitted
```

##		xhat	level
##	Feb 2001	0	0
##	Mar 2001	0	0
##	Apr 2001	0	0
##	May 2001	0	0
##	Jun 2001	0	0
##	Jul 2001	0	0
##	Aug 2001	0	0
##	Sep 2001	0	0
##	Oct 2001	0	0
##	Nov 2001	0	0
##	Dec 2001	0	0
##	Jan 2002	0	0
##	Feb 2002	0	0
##	Mar 2002	0	0
##	Apr 2002	0	0
##	May 2002	0	0
##	Jun 2002	0	0
##	Jul 2002	0	0
##	Aug 2002	0	0
##	Sep 2002	0	0
##	Oct 2002	0	0
##	Nov 2002	0	0
##	Dec 2002	0	0
##	Jan 2003	0	0
##	Feb 2003	0	0
##	Mar 2003	0	0
##	Apr 2003	0	0
##	May 2003	0	0
##	Jun 2003	0	0
##	Jul 2003	0	0
##	Aug 2003	0	0
##	Sep 2003	0	0
##	Oct 2003	0	0
##	Nov 2003	0	0
##	Dec 2003	0	0
##	Jan 2004	0	0
##	Feb 2004	0	0
##	Mar 2004	0	0
##	Apr 2004	0	0
##	May 2004	0	0
##	Jun 2004	0	0
##	Jul 2004	0	0
##	Aug 2004	0	0
##	Sep 2004	0	0

##	Oct	2004	0	0
##	Nov	2004	0	0
##	Dec	2004	0	0
##	Jan	2005	0	0
##	Feb	2005	0	0
##	Mar	2005	0	0
##	Apr	2005	0	0
##	May	2005	0	0
##	Jun	2005	0	0
##	Jul	2005	0	0
##	Aug	2005	0	0
##	Sep	2005	0	0
##	Oct	2005	0	0
##	Nov	2005	0	0
##	Dec	2005	0	0
##	Jan	2006	0	0
##	Feb	2006	0	0
##	Mar	2006	0	0
##	Apr	2006	0	0
##	May	2006	0	0
##	Jun	2006	0	0
##	Jul	2006	0	0
##	Aug	2006	0	0
##	Sep	2006	0	0
##	Oct	2006	0	0
##	Nov	2006	0	0
##	Dec	2006	0	0
##	Jan	2007	0	0
##	Feb	2007	0	0
##	Mar	2007	0	0
##	Apr	2007	0	0
##	May	2007	0	0
##	Jun	2007	0	0
##	Jul	2007	0	0
##	Aug	2007	0	0
##	Sep	2007	0	0
##	Oct	2007	0	0
##	Nov	2007	0	0
##	Dec	2007	0	0
##	Jan	2008	0	0
##	Feb	2008	0	0
##	Mar	2008	0	0
##	Apr	2008	0	0
##	May	2008	0	0
##	Jun	2008	0	0

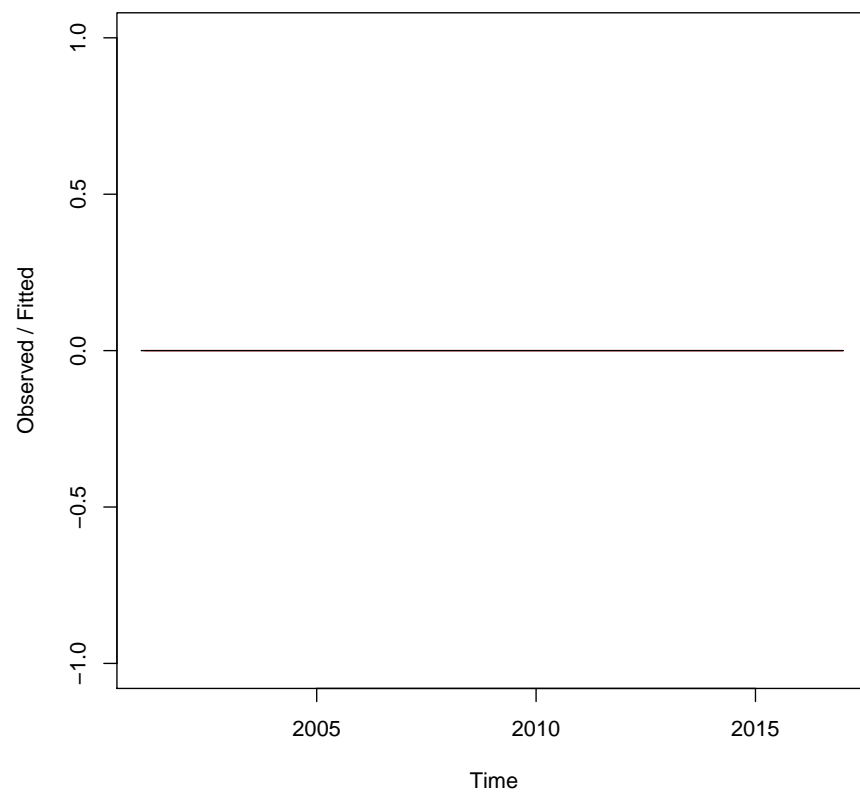
## Jul 2008	0	0
## Aug 2008	0	0
## Sep 2008	0	0
## Oct 2008	0	0
## Nov 2008	0	0
## Dec 2008	0	0
## Jan 2009	0	0
## Feb 2009	0	0
## Mar 2009	0	0
## Apr 2009	0	0
## May 2009	0	0
## Jun 2009	0	0
## Jul 2009	0	0
## Aug 2009	0	0
## Sep 2009	0	0
## Oct 2009	0	0
## Nov 2009	0	0
## Dec 2009	0	0
## Jan 2010	0	0
## Feb 2010	0	0
## Mar 2010	0	0
## Apr 2010	0	0
## May 2010	0	0
## Jun 2010	0	0
## Jul 2010	0	0
## Aug 2010	0	0
## Sep 2010	0	0
## Oct 2010	0	0
## Nov 2010	0	0
## Dec 2010	0	0
## Jan 2011	0	0
## Feb 2011	0	0
## Mar 2011	0	0
## Apr 2011	0	0
## May 2011	0	0
## Jun 2011	0	0
## Jul 2011	0	0
## Aug 2011	0	0
## Sep 2011	0	0
## Oct 2011	0	0
## Nov 2011	0	0
## Dec 2011	0	0
## Jan 2012	0	0
## Feb 2012	0	0
## Mar 2012	0	0

## Apr 2012	0	0
## May 2012	0	0
## Jun 2012	0	0
## Jul 2012	0	0
## Aug 2012	0	0
## Sep 2012	0	0
## Oct 2012	0	0
## Nov 2012	0	0
## Dec 2012	0	0
## Jan 2013	0	0
## Feb 2013	0	0
## Mar 2013	0	0
## Apr 2013	0	0
## May 2013	0	0
## Jun 2013	0	0
## Jul 2013	0	0
## Aug 2013	0	0
## Sep 2013	0	0
## Oct 2013	0	0
## Nov 2013	0	0
## Dec 2013	0	0
## Jan 2014	0	0
## Feb 2014	0	0
## Mar 2014	0	0
## Apr 2014	0	0
## May 2014	0	0
## Jun 2014	0	0
## Jul 2014	0	0
## Aug 2014	0	0
## Sep 2014	0	0
## Oct 2014	0	0
## Nov 2014	0	0
## Dec 2014	0	0
## Jan 2015	0	0
## Feb 2015	0	0
## Mar 2015	0	0
## Apr 2015	0	0
## May 2015	0	0
## Jun 2015	0	0
## Jul 2015	0	0
## Aug 2015	0	0
## Sep 2015	0	0
## Oct 2015	0	0
## Nov 2015	0	0
## Dec 2015	0	0

```
## Jan 2016    0    0
## Feb 2016    0    0
## Mar 2016    0    0
## Apr 2016    0    0
## May 2016    0    0
## Jun 2016    0    0
## Jul 2016    0    0
## Aug 2016    0    0
## Sep 2016    0    0
## Oct 2016    0    0
## Nov 2016    0    0
## Dec 2016    0    0
## Jan 2017    0    0
```

```
plot(serieFurti)
```

Holt-Winters filtering



```

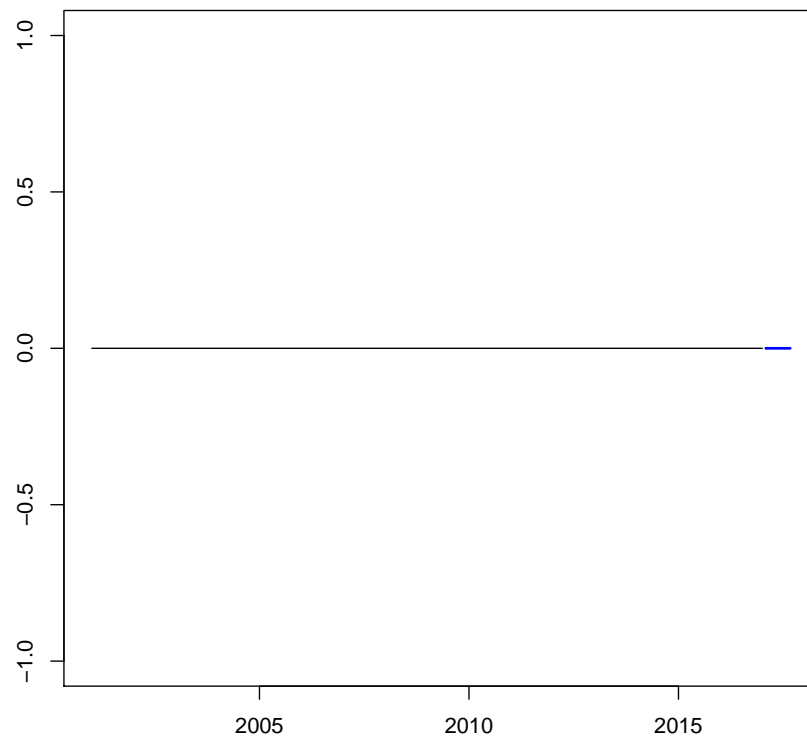
require(forecast)

## Loading required package: forecast
## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##   as.Date, as.Date.numeric
## Loading required package: timeDate
## This is forecast 7.3

serieFurti2 <- forecast::forecast.HoltWinters(serieFurti, h = 8)
plot(serieFurti2, main = "Previsione furti Veicoli per il 2017")

```

Previsione furti Veicoli per il 2017



```
acf(serieFurti2$residuals, lag.max = 20)

## Error in na.fail.default(as.ts(x)): missing values in object
```

La stima di Holt, (1957) ci permette di prevedere l'andamento del fenomeno preso in esame utilizzando 3 equazioni:

Una equazione per la previsione:

$$\hat{y}_{t+h|t} = l_t + hb_t$$

Una equazione di smoothing per il livello:

$$l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1})$$

Una equazione di smoothing per il trend:

$$b_t = \beta(l_t - l_{t-1}) + (1 + \beta)b_{t-1}$$

dove \hat{y}_t denota uno stimatore per il livello della serie al momento t . l_t denota invece una stima della pendenza del trend della serie temporale al tempo t . *α è invece il parametro di smoothing per il livello della serie.*

La mia analisi ora vuole spostarsi sui giorni (e le relative ore) al fine di valutare tramite un grafico heat, quali siano le ore più rischiose.

```
par(mfrow = c(2,2))

y <- c("Fn(Auto)", "Fn(RD)", "Fn(ROE)")
x <- c("Auto", "Ritrovamenti", "ROE")

plot(ecdf(monthlyByCateg[,2]), main = "Auto", xlab = x[1], ylab = y[1])
plot(ecdf(monthlyByCateg[,3]), main = "Attempted auto", xlab = x[1], ylab = y[1])
plot(ecdf(monthlyByCateg[,5]), main = "Recovered Auto", xlab = x[1], ylab = y[1])

prova <- filter(tots, Description == "ATT: AUTOMOBILE", Year == '2001')

## Error in filter_(.data, .dots = lazyeval::lazy_dots(...)): oggetto
## "tots" non trovato

#tipologie di descrizione
n_distinct(tots$Description)

## Error in n_distinct_multi(list(...), na.rm): oggetto "tots" non
## trovato

tots.transposed <- t(tots[,3])

## Error in t(tots[, 3]): oggetto "tots" non trovato
```

```

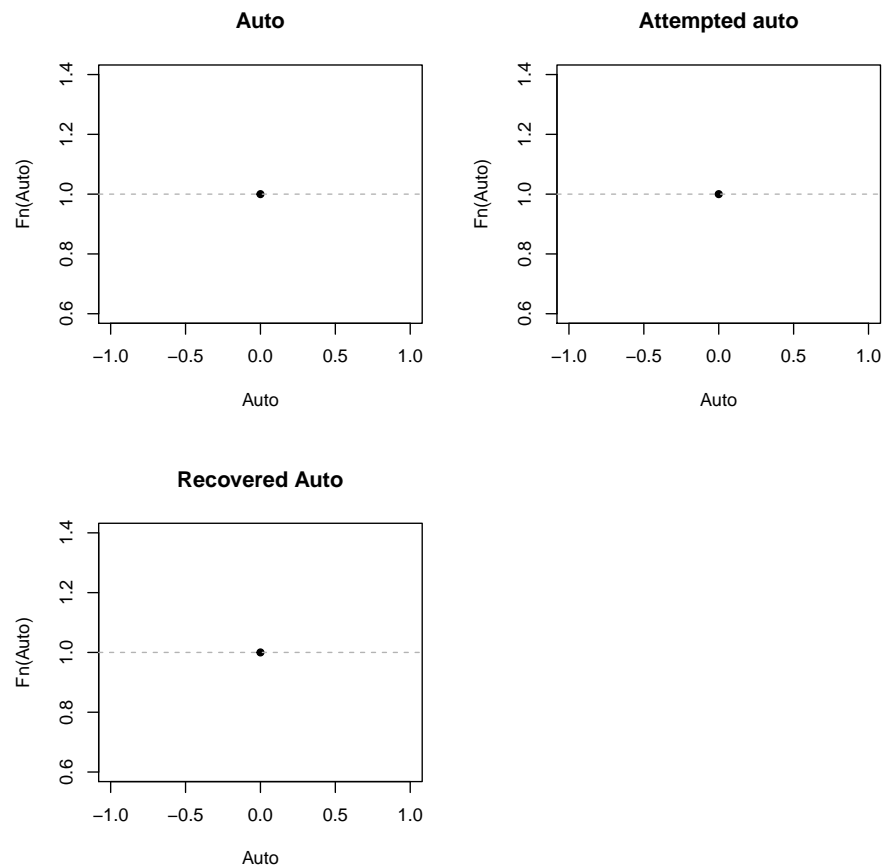
boxplot(filter(tots,tots$Year == 2005))

## Error in filter_(.data, .dots = lazyeval::lazy_dots(...)): oggetto
"tots" non trovato

qplot(data = tots, x = Year, y = count, )

## Error in ggplot(data, aesthetics, environment = env): oggetto "tots"
non trovato

```



Noto immediatamente che il 2017 ha solo 910 furti. Chiaramente essendo il 2017 iniziato da poco, questo valore è da trattare adeguatamente per non sfalsare la nostra analisi.

Ora risulterebbe più interessante accoppiare il totale di ogni anno con le tipologie di furto (colonna **Description**)