FDA Clustering + ANOVA

Tommaso Giorgi

2023-07-11

```
library(fda)
library(KernSmooth)
library(fields)
library(fdacluster)
library(lubridate, warn.conflicts = TRUE)
```

Per un approccio con Functional data analysis, per quanto abbiamo fatto nel corso non è possibile utilizzare le covariate (forecasted load, gas price etc. . .). Inoltre per non sovraccaricare il mio povero pc non ho potuto utilizzare tutte le osservazioni ma per ogni mese ho calcolato il prezzo medio giornaliero. Più precisamente per ogni mese e per ogni ora ho calcolato la media del prezzo nel day ahead market.

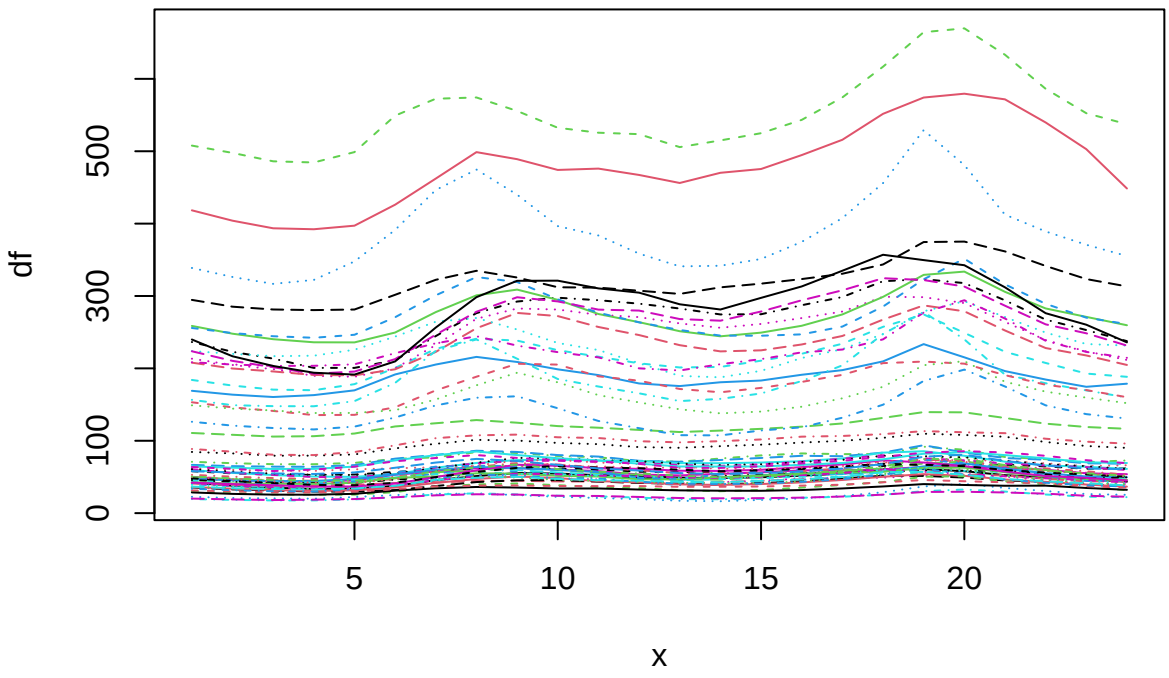
Per ogni mese abbiamo quindi una spezzata per la quale suppongo che sia l'approssimazione di una funzione liscia.

Creazione del nuovo dataset:

```
data <- read.table('data_w_temp.csv', header = TRUE, sep=",")
dates <- ymd_hms(data$date)
data = data.frame(dates, data[,-1])

df1 = data.frame(data$dam)

df <- matrix(df1$data.dam[-46681], nrow = 1945, ncol = 24, byrow = TRUE)
df <- as.data.frame(df)


rows <- floor(nrow(df) / 30)
new_df <- data.frame(matrix(NA, nrow = rows, ncol = 24))

# Raggruppamento delle righe e calcolo della media per ogni gruppo
for (i in 1:rows) {
  ri <- (i - 1) * 30 + 1
  rf <- min(i * 30, nrow(df))
  new_df[i, ] <- colMeans(df[ri:rf, ])
}
```

Plot delle funzioni:

```
df <- t(new_df)
x <- seq(1,24,by=1)
matplot(x, df,type='l')
```



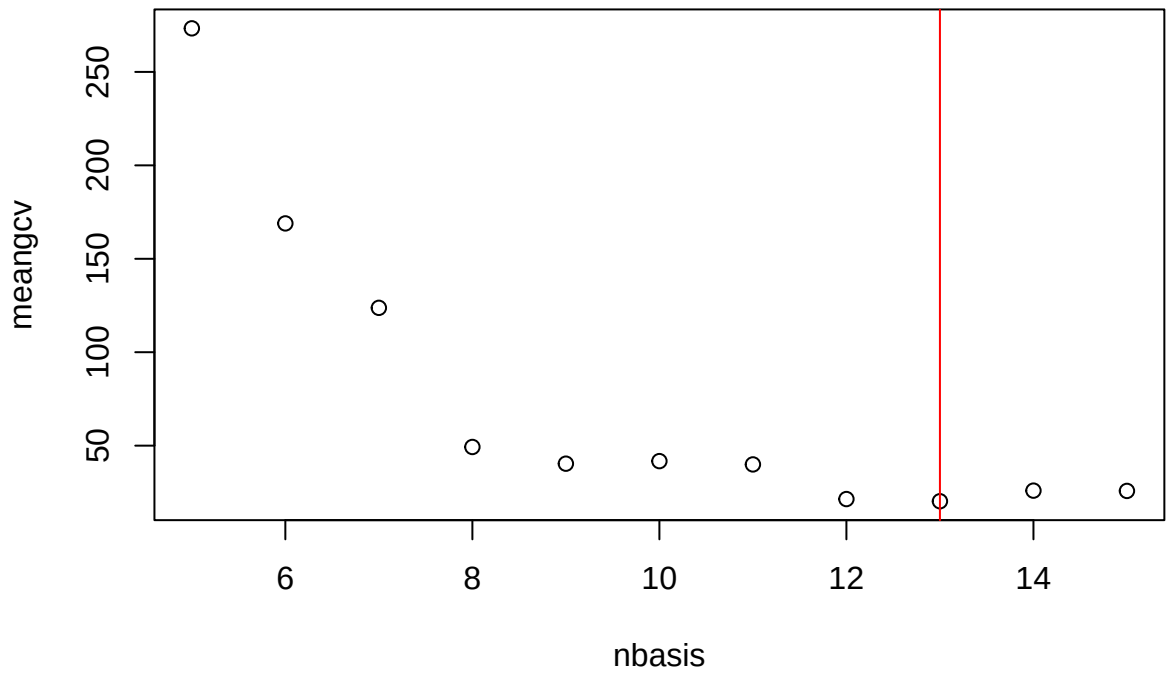

```
N <- dim(df)[2]
```

Per fare smoothing ho usato delle splines di ordine 5 e per trovare il giusto numero di basi ho usato la generalized cross-validation

```
nbasis <- 5:15
m <- 5
gcv <- matrix(0, nrow = length(nbasis), ncol = dim(df)[2])
for (i in 1:length(nbasis)){
  basis <- create.bspline.basis(c(x[1],x[length(x)]), nbasis[i], m)
  for(j in 1:dim(df)[2]){
  gcv[i,j] <- smooth.basis(x, as.numeric(df[,j]), basis)$gcv
  }
}
meangcv <- rowMeans(gcv)

par(mfrow=c(1,1))
```

Dato che a lezione avevamo visto questo approccio per una sola curva ho iterato su tutte le curve e per ogni nbasis ho calcolato il gcv medio tra tutte le curve (ditemi se è giusto)

```
plot(nbasis,meangcv)
nbasis.opt <- nbasis[which.min(meangcv)] # Optimal value

abline(v=nbasis[which.min(meangcv)],col='red')
```



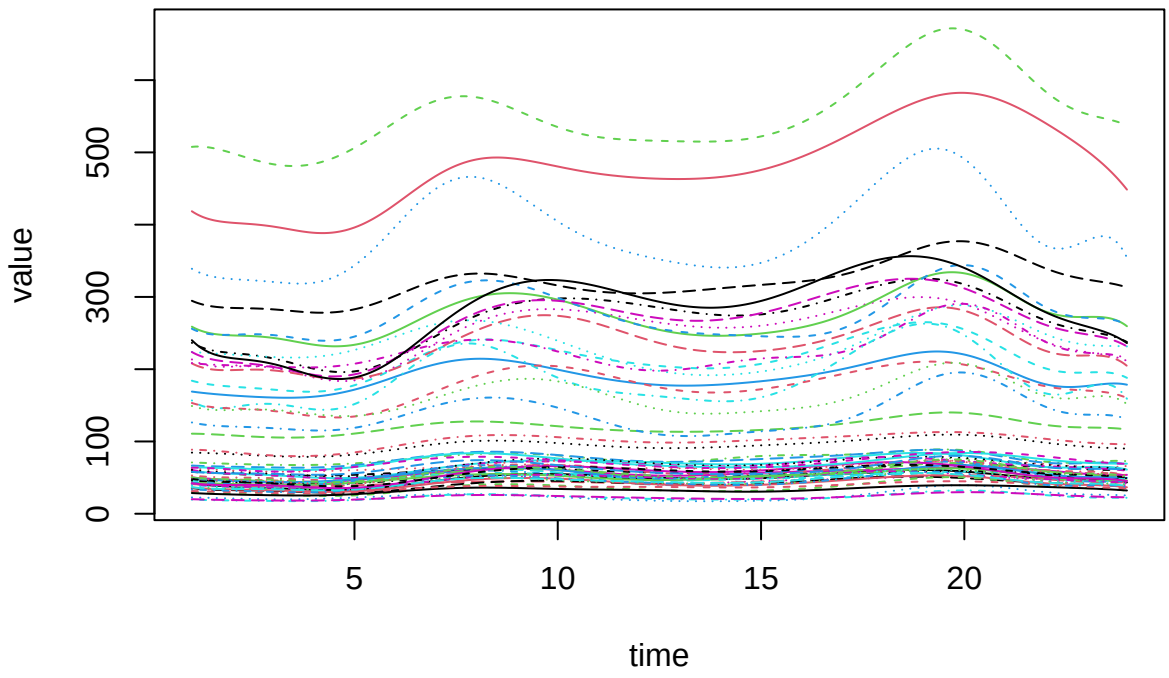
Numero di basi ottimale:

```
nbasis.opt
```

```
## [1] 13
```

Plot delle curve smoothed:

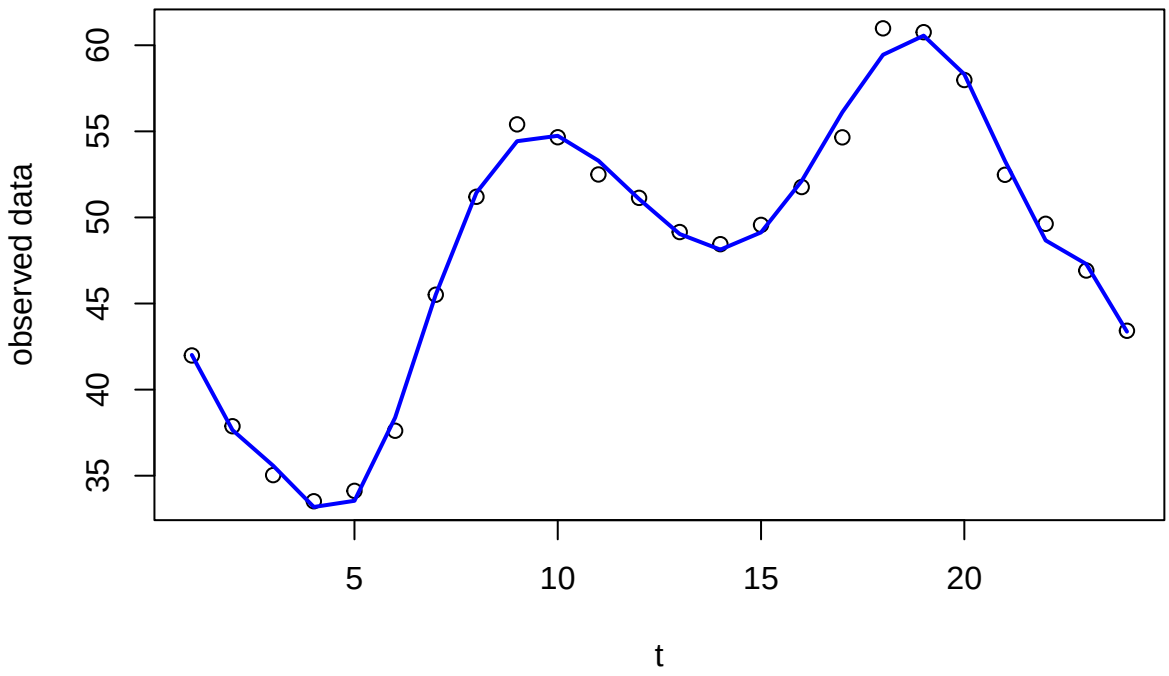
```
nbasis <- nbasis.opt
m <- 5
degree <- m-1
N <- dim(df)[2]
basis <- create.bspline.basis(rangeval=c(x[1],x[length(x)]),nbasis=nbasis,norder=m)
df.fun <- Data2fd(y = df,argvals = x,basisobj = basis)
plot.fd(df.fun)
```



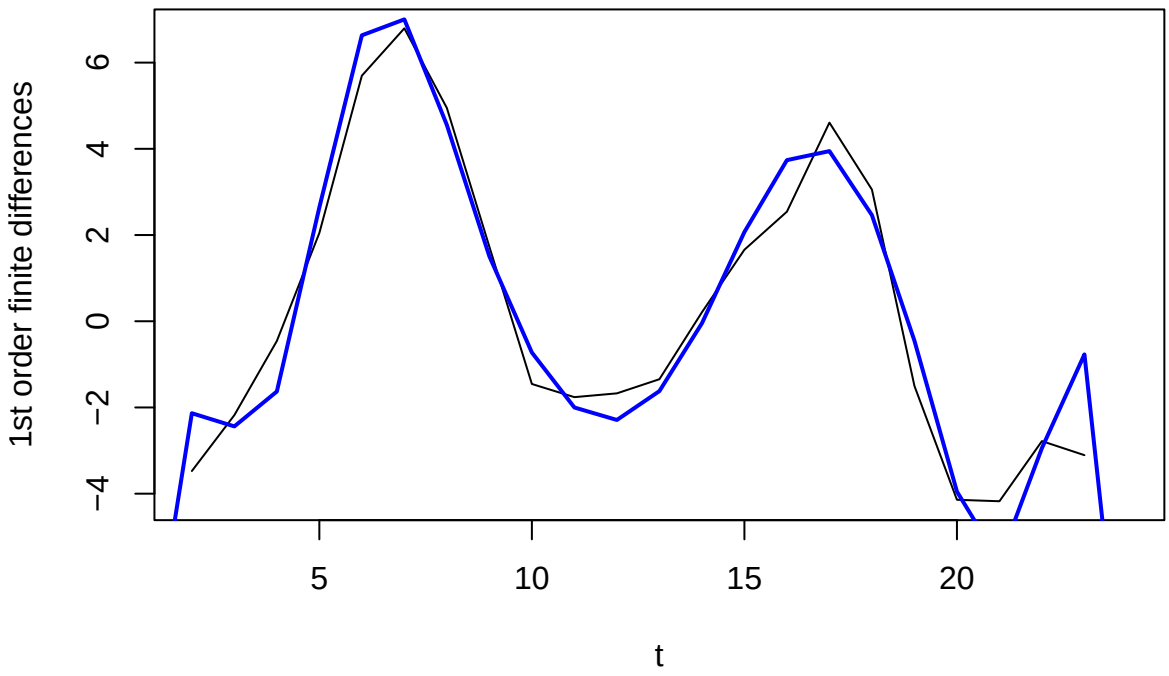

```
## [1] "done"
```

Plot della derivata prima contro quella data dalle differenze finite:

```
y.spline <- smooth.basis(argvals=x, y=df[,1], fdParobj=basis)
y.spline0 <- eval.fd(x, y.spline$fd) #  the curve smoothing the data
y.spline1 <- eval.fd(x, y.spline$fd, Lfd=1) # first derivative
y.spline2 <- eval.fd(x, y.spline$fd, Lfd=2) # second derivative

rappinc1 <- (df[,1][3:N]-df[,1][1:(N-2)])/(x[3:N]-x[1:(N-2)])
rappinc2 <- ((df[,1][3:N]-df[,1][2:(N-1)])/(x[3:N]-x[2:(N-1)])-(df[,1][2:(N-1)]-df[,1][1:(N-2)])/(x[2:(


plot(x,df[,1],xlab="t",ylab="observed data")
points(x,y.spline0 ,type="l",col="blue",lwd=2)
```




```
plot(x[2:(N-1)],rappinc1,xlab="t",ylab="1st order finite differences",type="l")
points(x,y.spline1 ,type="l",col="blue",lwd=2)
```



Dal plot delle funzioni smoothed io ho messo 2 cluster (fatemi sapere):

```
k <- 2


y0 <- t(eval.fd(x, df.fun, Lfd=0)) # evaluations of the functions or derivatives after smoothing
y1 <- t(eval.fd(x, df.fun, Lfd=1)) # evaluations of original functions (matrix with 1 row for each func
```

Clustering

Per fare clustering con allignment ho usato il pacchetto di Secchi, Sangalli &co.

In particolare invece che usare kmean allignment che non funzionava troppo bene ho usato hierarchical clustering con warping "srsf" che per quello che ho letto tiene i bordi costanti quindi non shifta le funzioni (guardate anche voi se riuscite per capire se ha senso)

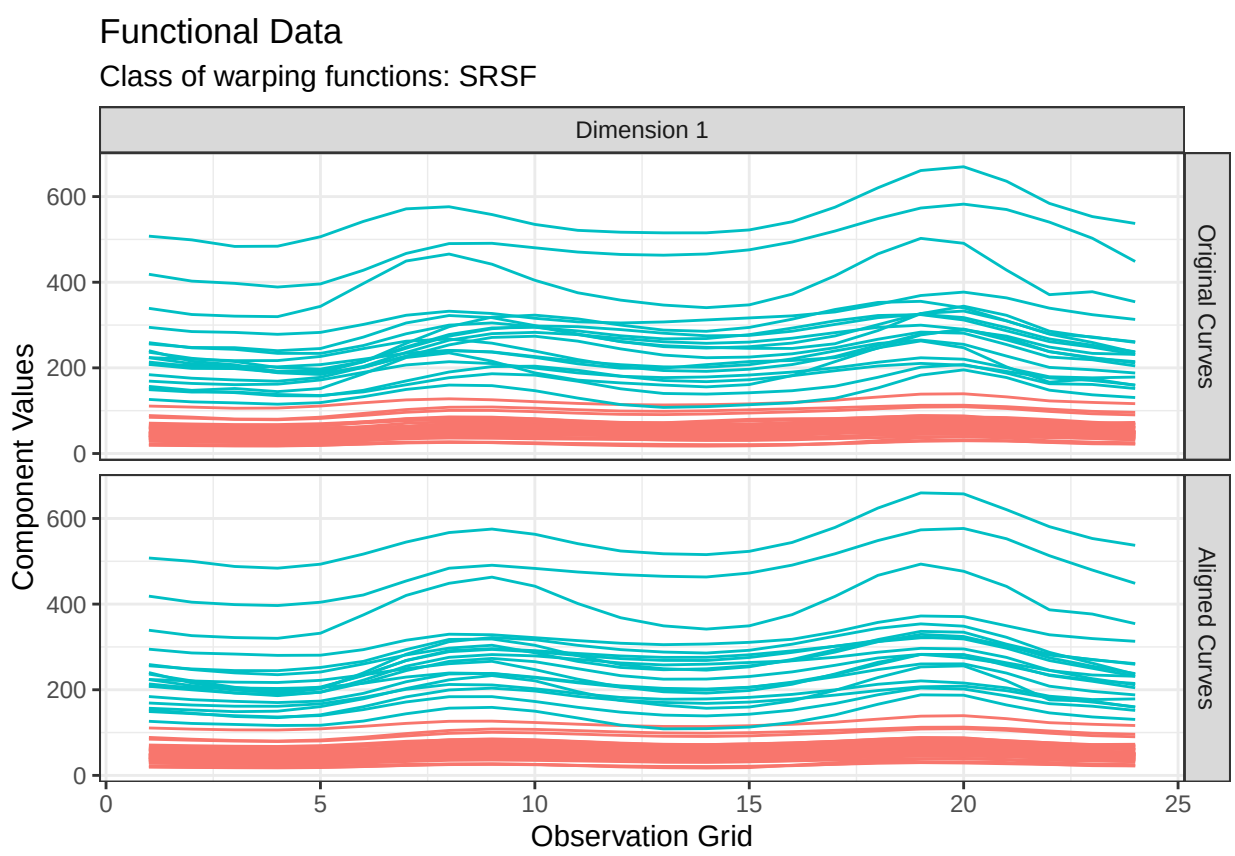
```
set.seed(1)
fdahcl <- fdahclust(
  x=x, y=y0, n_clusters = 2,
  warping_class = 'srsf',
  metric = 'l2',
  centroid_type = 'mean',
  linkage_criterion = 'complete')
```

```
## i Computing the distance matrix...
```

```
## Warning: il pacchetto 'fdasrvf' è stato creato con R versione 4.2.3
```

```
## i Calculating the tree...
## i Aligning all curves with respect to their centroid...
## i Consolidating output...
```

```
plot(fdahcl, type =  "amplitude")
```




```
id <- which.min(fdahcl$silhouettes)
clusters <- fdahcl$memberships
```

ANOVA sui cluster identificati

Adesso data la funzione di 24 ore per il prezzo mensile ho fatto anche la media tra le 24 ore perchè sennò avrei avuto 24 covariate. Ho fatto il log del prezzo così da avere le ipotesi soddisfatte

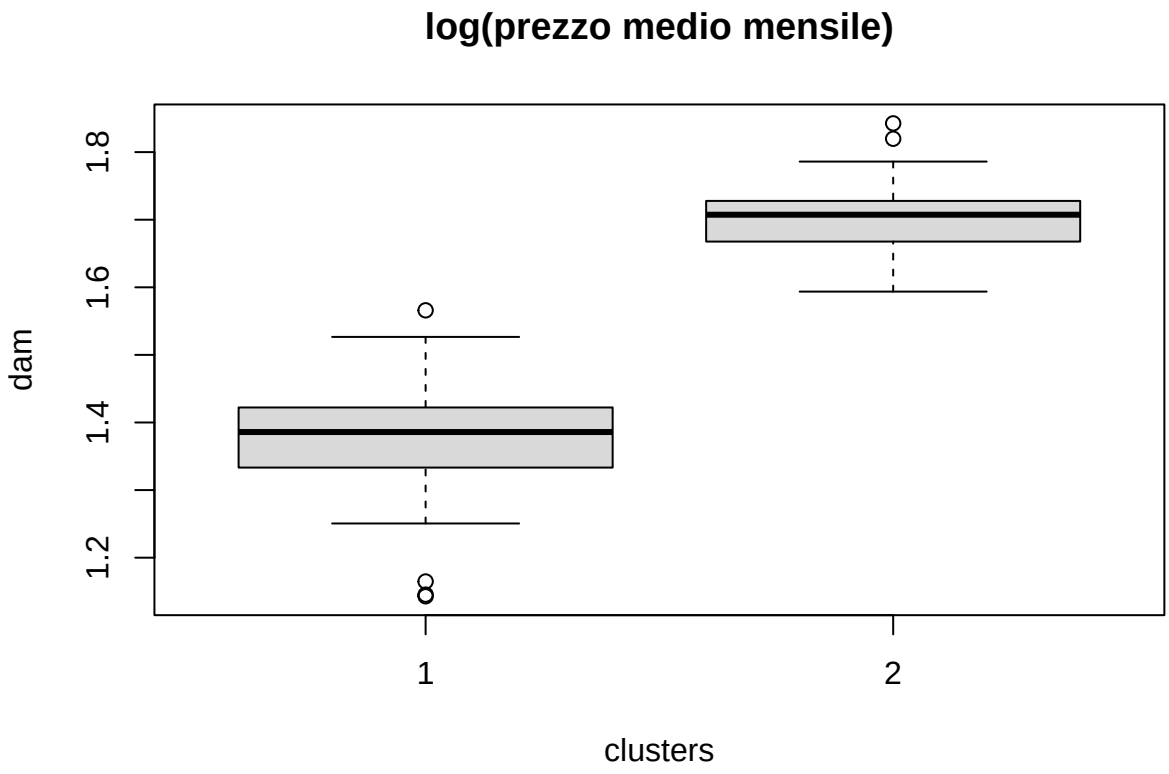
```
df1 <- data$dam
df <- matrix(df1[-(46081:46681)], nrow = 64, ncol = 720, byrow = TRUE)
df <- rowMeans(df)

df <- log(df)


df <- cbind(df,clusters)
df <- as.data.frame(df)

head(df)
```

```
##         df clusters
## 1 3.874920        1
## 2 4.057332        1
## 3 4.041092        1
## 4 3.882119        1
## 5 3.937350        1
## 6 4.019928        1
```

```
names(df)[1] <- 'dam'


data <- df

p <- 1
factor1 <- factor(data$clusters)

x <- log(data$dam)
```

Box plot:

```
plot(factor1, x, xlab='clusters', ylab='dam', col='grey85', main='log(prezzo medio mensile)')
```




```
n <- dim(data)[1]
ng <- table(factor1)
treat <- levels(factor1)
g <- length(treat)
```

Normalità:

```
pvalue <- NULL
for (i in 1:g) {
  pval <- shapiro.test(x[factor1==treat[i]])$p
  pvalue <- c(pvalue, pval)
}
pvalue
```

```
## [1] 0.02357615 0.78715824
```

Omoschedasticità

```
bartlett.test(x, factor1)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  x and factor1
## Bartlett's K-squared = 2.3876, df = 1, p-value = 0.1223
```

Summary dell'ANOVA

```
fit <- aov(x ~ factor1)
summary(fit)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## factor1      1 1.5077  1.5077   231.1 <2e-16 ***
## Residuals   62 0.4045  0.0065
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Evidenza statistica che c'è differenza tra i due cluster...