

LABORATORIO 10 - Modello di regressione lineare

**STATISTICA E LABORATORIO (CDL in INTERNET OF THINGS,
BIG DATA, MACHINE LEARNING)**

Anno Accademico 2021-2022

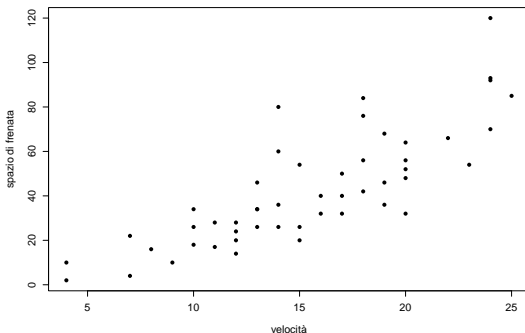
Section 1

Analisi di correlazione

Velocita'

Si considerano i dati sulla velocità X e sullo spazio di frenata Y di $n = 50$ automobili degli anni 20.

```
plot(cars$speed,cars$dist,main=" ",xlab="velocità",  
      ylab="spazio di frenata",cex.axis=1.2,cex.lab=1.2,pch=16)
```



```
ro <- cor(cars$speed,cars$dist) # coefficiente di correlazione  
ro
```

```
## [1] 0.8068949
```

```
# valore osservato per la statistica test sotto  $H_0$   
ro*sqrt(length(cars$speed)-2)/sqrt(1-ro^2)
```

```
## [1] 9.46399
```

```
# soglia superiore: si rifiuta  $H_0$   
qt(0.01,length(cars$speed)-2,lower.tail=FALSE)
```

```
## [1] 2.406581
```

```
# p-value  
pt(ro*sqrt(length(cars$speed)-2)/sqrt(1-ro^2),length(cars$speed)-2,  
  lower.tail=FALSE)
```

```
## [1] 7.449182e-13
```

```
# in alternativa
```

```
cor.test(cars$speed,cars$dist,alternative="greater")
```

```
##
```

```
## Pearson's product-moment correlation
```

```
##
```

```
## data: cars$speed and cars$dist
```

```
## t = 9.464, df = 48, p-value = 7.449e-13
```

```
## alternative hypothesis: true correlation is greater than 0
```

```
## 95 percent confidence interval:
```

```
## 0.7054856 1.0000000
```

```
## sample estimates:
```

```
## cor
```

```
## 0.8068949
```

```
# test di correlazione utilizzando l'indice di Spearman e  
# l'indice di Kendall
```

```
cor.test(cars$speed,cars$dist,  
         method="spearman",alternative="greater")
```

```
## Warning in cor.test.default(cars$speed, cars$dist, method = "spea  
## compute exact p-value with ties
```

```
##  
## Spearman's rank correlation rho
```

```
##  
## data: cars$speed and cars$dist
```

```
## S = 3532.8, p-value = 4.412e-14
```

```
## alternative hypothesis: true rho is greater than 0
```

```
## sample estimates:
```

```
## rho
```

```
## 0.8303568
```

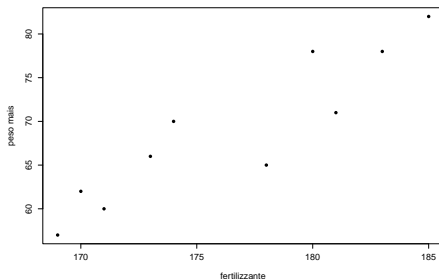
```
cor.test(cars$speed,cars$dist,  
         method="kendall",alternative="greater")
```

```
##  
## Kendall's rank correlation tau  
##  
## data: cars$speed and cars$dist  
## z = 6.6655, p-value = 1.319e-11  
## alternative hypothesis: true tau is greater than 0  
## sample estimates:  
##      tau  
## 0.6689901
```

Mais

Si considerano i dati sulla dose di fertilizzante utilizzata x e sulla quantità di mais prodotta Y (peso della granella in Kg), con riferimento a $n = 10$ distinte parcelle sperimentali, simili per caratteristiche e della medesima dimensione.

```
x <- c(171,169,181,173,178,180,185,183,170,174)
y <- c(60,57,71,66,65,78,82,78,62,70)
plot(x,y,main=" ",xlab="fertilizzante",ylab="peso mais",
      cex.axis=1.2,cex.lab=1.2,pch=16)
```




```
mean(y)
```

```
## [1] 68.9
```

```
mean(x)
```

```
## [1] 176.4
```

```
cov(y,x)*9/10
```

```
## [1] 39.64
```

```
var(x)*9/10
```

```
## [1] 29.64
```

```
b <- cov(x,y)/var(x)  
b
```

```
## [1] 1.337382
```

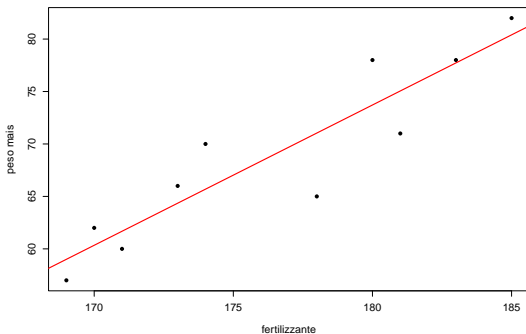
```
a <- mean(y)-b*mean(x)
a
```

```
## [1] -167.0142
```

```
# in alternativa si usa la funzione lm() che permette
# la stima di un modello lineare e fornisce un oggetto
# di classe 'lm'
lm_mais <- lm(y~x)
lm_mais$coefficients
```

```
## (Intercept)          x
## -167.014170    1.337382
```

```
plot(x,y,main=" ",xlab="fertilizzante",ylab="peso mais",  
     cex.axis=1.2,cex.lab=1.2,pch=16)  
# per aggiungere la retta di regressione stimata  
abline(lm_mais,lwd=2,col='red')
```



```
# la funzione summary(), con argomento un oggetto di classe 'lm',  
# fornisce: alcune statistiche riassuntive sui residui stimati,  
# una tabella con le stime di a e b, gli associati standard error  
# stimati, il valore della statistica test e del p-value per  
#  $H_0: a=0$  e  $H_0: b=0$ . Inoltre, una stima per la deviazione  
# standard dei residui (residual standard error), che elevata  
# al quadrato corrisponde  
# alla stima per la varianza dei residui
```

```
summary(lm_mais)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
```

| | Min | 1Q | Median | 3Q | Max |
|----|---------|---------|--------|--------|--------|
| ## | -6.0398 | -1.9221 | 0.9359 | 1.6562 | 4.3097 |

```
##
## Coefficients:
```

| | Estimate | Std. Error | t value | Pr(> t) | |
|----------------|-----------|------------|---------|----------|-----|
| ## (Intercept) | -167.0142 | 37.0956 | -4.502 | 0.001996 | ** |
| ## x | 1.3374 | 0.2102 | 6.363 | 0.000218 | *** |

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.619 on 8 degrees of freedom
## Multiple R-squared:  0.835, Adjusted R-squared:  0.8144
## F-statistic: 40.48 on 1 and 8 DF, p-value: 0.0002176
```

```
summary(lm_mais)$sigma^2 # stima della varianza dei residui
```

```
## [1] 13.09523
```

```
fitted.values(lm_mais) # valori stimati dal modello
```

```
##          1          2          3          4          5          6          7
## 61.67814 59.00337 75.05196 64.35290 71.03981 73.71457 80.40148 77
##          9         10
## 60.34076 65.69028
```

```
residuals(lm_mais) # residui stimati
```

```
##          1          2          3          4          5          6
## -1.6781377 -2.0033738 -4.0519568  1.6470985 -6.0398111  4.2854251
##          8          9         10
##  0.2732794  1.6592443  4.3097166
```

```
# i residui stimati hanno media nulla  
mean(residuals(lm_mais))
```

```
## [1] 5.00034e-17
```

```
# stima della varianza dei residui  
sum(residuals(lm_mais)^2)/(length(y)-2)
```

```
## [1] 13.09523
```

```
# matrice con stime, standard error, test e p-value
```

```
summary(lm_mais)$coefficients
```

```
##              Estimate Std. Error  t value    Pr(>|t|)
## (Intercept) -167.014170 37.0956276 -4.50226 0.0019961194
## x            1.337382  0.2101926  6.36265 0.0002176421
```

```
# intervallo di confidenza per a di livello 0.95
```

```
summary(lm_mais)$coefficients[1,1]+c(qt(0.025,8),
qt(0.025,8,lower.tail=FALSE))*summary(lm_mais)$coefficients[1,2]
```

```
## [1] -252.5568 -81.4715
```

```
# intervallo di confidenza per b di livello 0.95
```

```
summary(lm_mais)$coefficients[2,1]+c(qt(0.025,8),
qt(0.025,8,lower.tail=FALSE))*summary(lm_mais)$coefficients[2,2]
```

```
## [1] 0.8526769 1.8220869
```



```
# la funzione predict(), con argomento un oggetto di classe 'lm',  
# permette la stima della media della variabile risposta e il  
# calcolo degli associati intervalli di confidenza. Inoltre si  
# possono calcolare previsioni e intervalli di previsione  
# valori stimati dal modello (senza argomenti aggiuntivi si  
# considerano i valori osservati di x); stessi risultati di  
# fitted.values(lm_mais)  
predict(lm_mais)
```

```
##          1          2          3          4          5          6          7  
## 61.67814 59.00337 75.05196 64.35290 71.03981 73.71457 80.40148 77.00000  
##          9          10  
## 60.34076 65.69028
```

*# oltre ai valori stimati dal modello si ottengono gli estremi
inferiore e superiore degli intervalli di confidenza (di livello
0.95) per la media delle Y*

```
predict(lm_mais, interval="confidence")
```

| ## | | fit | lwr | upr |
|-------|--|----------|----------|----------|
| ## 1 | | 61.67814 | 57.96136 | 65.39491 |
| ## 2 | | 59.00337 | 54.55041 | 63.45633 |
| ## 3 | | 75.05196 | 71.59726 | 78.50665 |
| ## 4 | | 64.35290 | 61.24171 | 67.46409 |
| ## 5 | | 71.03981 | 68.28935 | 73.79027 |
| ## 6 | | 73.71457 | 70.55097 | 76.87818 |
| ## 7 | | 80.40148 | 75.46796 | 85.33501 |
| ## 8 | | 77.72672 | 73.57973 | 81.87371 |
| ## 9 | | 60.34076 | 56.26808 | 64.41343 |
| ## 10 | | 65.69028 | 62.80639 | 68.57418 |

```

# definendo l'argomento newdata (che deve essere un dataframe con
# un elemento con lo stesso nome della variabile esplicativa), si
# possono ottenere previsioni per Y (che coincidono con stime
# della media di Y) riferite a nuovi valori di x, l'intervallo di
# di confidenza (di livello 0.95) per la media di Y e
# l'intervallo di previsione (di livello 0.95) per Y
xnew<-data.frame(x=c(176))
# stima della media di Y (con x=xnew) e intervallo di confidenza
# di livello 0.95
predict(lm_mais,newdata=xnew,interval="confidence")

```

```

##          fit          lwr          upr
## 1 68.36505 65.71907 71.01102

```

```
# previsione per Y (con x=xnew) e intervallo di previsione
# di livello 0.95
predict(lm_mais,newdata=xnew,interval="prediction")
```

```
##          fit          lwr          upr
## 1 68.36505 59.61079 77.11931
```

```
# stima della media di Y e previsione per Y coincidono ma
# l'intervallo di previsione e' piu' ampio
var(residuals(lm_mais))*9/10 # varianza residua
```

```
## [1] 10.47618
```

```
var(y)*9/10 # varianza totale
```

```
## [1] 63.49
```

```
1-var(residuals(lm_mais))/var(y) # indice di determinazione lineare
```

```
## [1] 0.8349948
```

```
cor(x,y)^2 # # indice di determinazione lineare come quadrato di cor
```

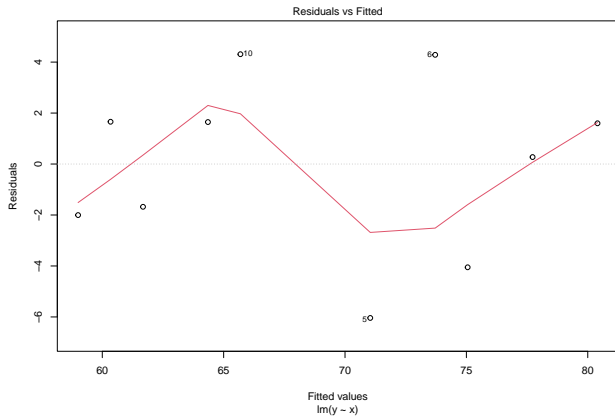
```
## [1] 0.8349948
```

```
# alternativa
```

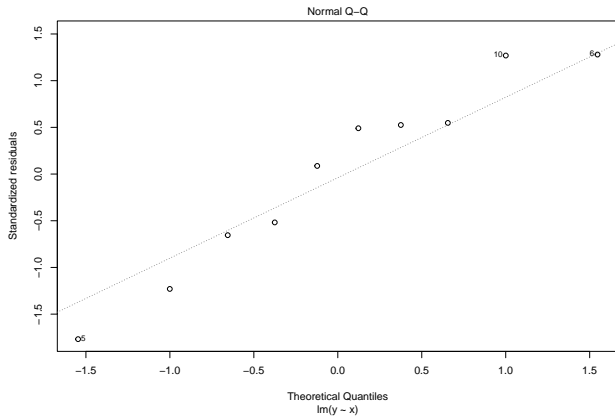
```
summary(lm_mais)$r.squared
```

```
## [1] 0.8349948
```

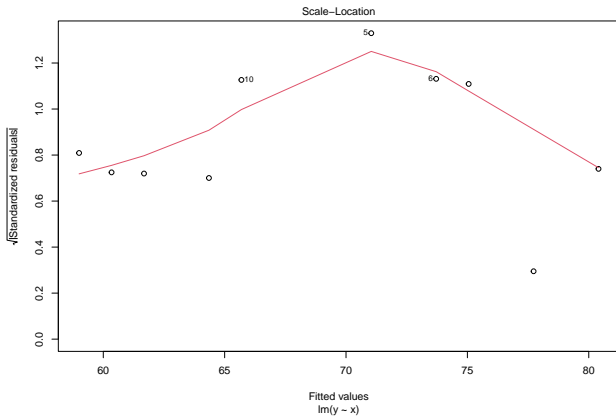
```
# grafici per le diagnostiche  
plot(lm_mais, which=1)
```



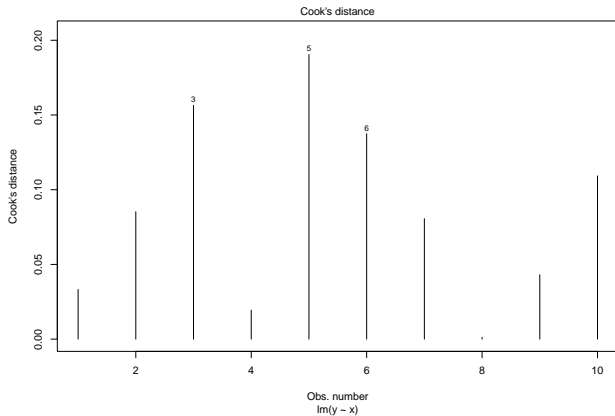
```
plot(lm_mais, which=2)
```



```
plot(lm_mais, which=3)
```




```
plot(lm_mais, which=4)
```

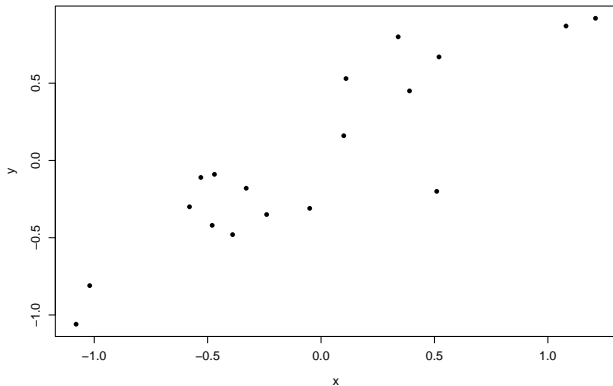


Misurazioni

Per valutare la qualità di un prodotto si può utilizzare una procedura precisa ma costosa, descritta dalla variabile casuale Y , oppure una procedura meno precisa ma anche meno costosa, descritta dalla variabile x . Si considerano le misurazioni, effettuate con entrambe le procedure, riferite a $n = 18$ prodotti.

```
x <- c(-1.08,-1.02,-0.39,-0.48,-0.58,-0.24,-0.05,-0.33,0.51,-0.53,  
      -0.47,0.1,0.39,0.11,0.52,0.34,1.08,1.21)  
y <- c(-1.06,-0.81,-0.48,-0.42,-0.30,-0.35,-0.31,-0.18,-0.20,  
      -0.11,-0.09,0.16,0.45,0.53,0.67,0.80,0.87,0.92)
```

```
plot(x,y,lwd=2,xlab="x", ylab="y", pch=16,cex.axis=1.2,cex.lab=1.2)
```



```
mean(y)
```

```
## [1] 0.005
```

```
mean(x)
```

```
## [1] -0.05055556
```

```
cov(y,x)*17/18
```

```
## [1] 0.3121583
```

```
var(x)*17/18
```

```
## [1] 0.3889608
```

```
b <- cov(x,y)/var(x)  
b
```

```
## [1] 0.8025445
```

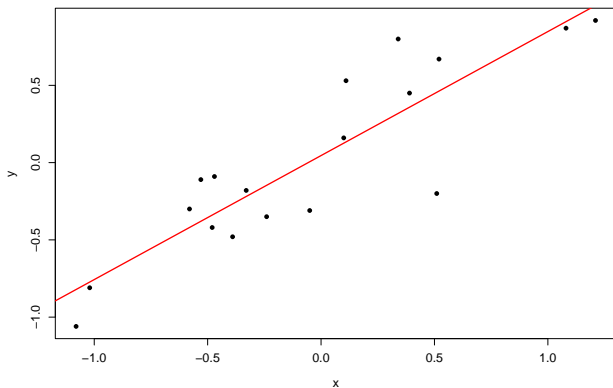
```
a <- mean(y)-b*mean(x)
```

```
a
```

```
## [1] 0.04557308
```

```
lm_mis <- lm(y~x) # stima del modello lineare
```

```
plot(x,y,lwd=2,xlab="x", ylab="y", pch=16,cex.axis=1.2,cex.lab=1.2)  
abline(lm_mis,lwd=2, col="red")
```



```
fitted.values(lm_mis) # valori stimati dal modello
```

```
##           1           2           3           4           5
## -0.821174926 -0.773022259 -0.267419255 -0.339648256 -0.419902701
##           7           8           9          10          11
##  0.005445858 -0.219266588  0.454870750 -0.379775478 -0.331622811
##          13          14          15          16          17
##  0.358565416  0.133852970  0.462896195  0.318438194  0.912321087
```

```
residuals(lm_mis) # residui stimati
```

```
##           1           2           3           4           5
## -0.23882507 -0.03697774 -0.21258074 -0.08035174  0.11990270 -0.20
##           7           8           9          10          11
## -0.31544586  0.03926659 -0.65487075  0.26977548  0.24162281  0.03
##          13          14          15          16          17
##  0.09143458  0.39614703  0.20710381  0.48156181 -0.04232109 -0.09
```

```
# stima della varianza dei residui  
sum(residuals(lm_mis)^2)/(length(y)-2)
```

```
## [1] 0.07994207
```

```
# in alternativa  
summary(lm_mis)$sigma^2
```

```
## [1] 0.07994207
```


matrice con stime, standard error, test e p-value

```
summary(lm_mis)$coefficients
```

```
##              Estimate Std. Error   t value    Pr(>|t|)
## (Intercept) 0.04557308 0.06686112 0.6816081 5.052341e-01
## x           0.80254445 0.10685590 7.5105298 1.245564e-06
```

intervallo di confidenza per a di livello 0.95

```
summary(lm_mis)$coefficients[1,1]+c(qt(0.025,16),
qt(0.025,16,lower.tail=FALSE))*summary(lm_mis)$coefficients[1,2]
```

```
## [1] -0.09616616  0.18731232
```

intervallo di confidenza per b di livello 0.95

```
summary(lm_mis)$coefficients[2,1]+c(qt(0.025,16),
qt(0.025,16,lower.tail=FALSE))*summary(lm_mis)$coefficients[2,2]
```

```
## [1] 0.5760201 1.0290688
```

```
var(residuals(lm_mis))*17/18 # varianza residua
```

```
## [1] 0.07105962
```

```
var(y)*17/18 # varianza totale
```

```
## [1] 0.3215806
```

```
1-var(residuals(lm_mis))/var(y) # indice di determinazione lineare
```

```
## [1] 0.7790301
```

```
# alternativa
```

```
summary(lm_mis)$r.squared
```

```
## [1] 0.7790301
```

```
summary(lm_mis) # sintesi dei principali risultati
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
```

| | Min | 1Q | Median | 3Q | Max |
|----|---------|---------|---------|--------|--------|
| ## | -0.6549 | -0.1764 | -0.0014 | 0.1853 | 0.4816 |

```
##
## Coefficients:
```

| | Estimate | Std. Error | t value | Pr(> t) |
|----------------|----------|------------|---------|--------------|
| ## (Intercept) | 0.04557 | 0.06686 | 0.682 | 0.505 |
| ## x | 0.80254 | 0.10686 | 7.511 | 1.25e-06 *** |

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2827 on 16 degrees of freedom
## Multiple R-squared:  0.779, Adjusted R-squared:  0.7652
## F-statistic: 56.41 on 1 and 16 DF, p-value: 1.246e-06
```

```
# plot diagnostici; l'opzione "which = " individua il  
# grafico che si vuole rappresentare  
par(mfrow=c(1,3))  
plot(lm_mis, which = 1, lwd=2, pch = 16, cex.lab=1.2,cex.axis=1.2,  
      caption="")  
plot(lm_mis, which = 2, lwd=2, pch = 16, cex.lab=1.2,cex.axis=1.2,  
      caption="")  
plot(lm_mis, which = 4, lwd=2, pch = 16, cex.lab=1.2,cex.axis=1.2,  
      caption="")  
  
par(mfrow=c(1,1))
```

