**Esercizio 1. Trovare un intervallo della distribuzione di COR che contenga il 95% dell'area.**

Soluzione.

Ricordiamo che i comandi per ottenere (a livello simulativo) la distribuzione di COR sono (ad es. con n=27)

n=27; N=100000

COR = 1:N

for (i in 1:N) {

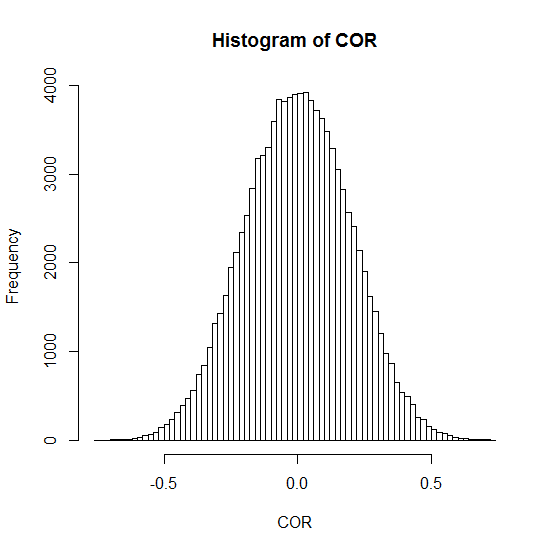
x= rnorm(n)

y= rnorm(n)

COR[i] = cor(x,y)

}

hist(COR,100)



Il modo più semplice è l’uso del comando quantile:

> quantile(COR,0.025)

2.5%

-0.3823481

> quantile(COR,0.975)

97.5%

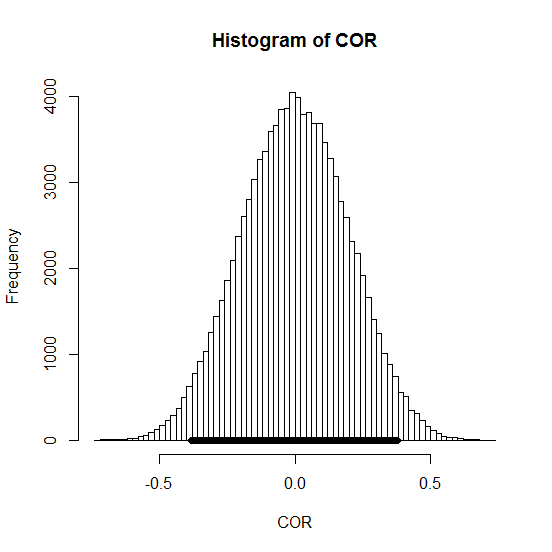
0.3799307

Risposta alla domanda dell’esercizio: l’intervallo [-0.382,0.380].

Possiamo tracciare l’intervallo di confidenza sul grafico:

PP=seq(quantile(COR,0.025), quantile(COR,0.975),0.001)

lines(PP,PP\*0,type="p")



**Esercizio: tracciare, invece, il complementare, cioè le due code utilizzabili come regione di rifiuto del test che vuole scoprire che le due variabili del campione non sono indipendenti.**

**Esercizio di contorno: dare una soluzione parametrica.**

Nota: con questo si intende che si deve trovare una densità di forma nota che corrisponda il meglio possibile ai dati COR e poi si deve calcolare l’intervallo usando tale densità.

Soluzione.

Dal grafico di COR si intuisce che la densità giusta potrebbe essere gaussiana. Se lo è, i suoi parametri sono

> mean(COR)

[1] -0.000860945

> sd(COR)

[1] 0.1961472

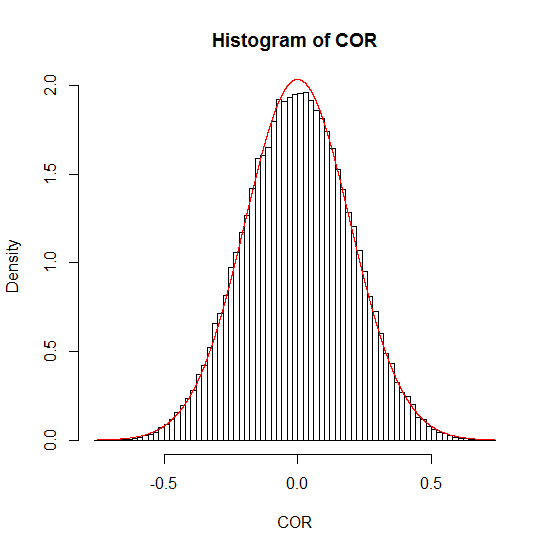
Proviamo allora a sovrapporre la densità gaussiana con questi parametri, all’istogramma:

hist(COR,100,FALSE)

COR.ord=sort(COR)

Y=dnorm(COR.ord,mean(COR),sd(COR))

lines(COR.ord,Y, col=2)



Combacia perfettamente. Allora possiamo calcolare

> qnorm(0.025,mean(COR),sd(COR))

[1] -0.3853024

> qnorm(0.975,mean(COR),sd(COR))

[1] 0.3835805

Con questo metodo l’intervallo è [-0.385,0.383].

**Quando scegliere tra parametrico e non-parametrico?** Tra i pregi della non-parametrica c’è la semplicità algoritmica, la generalità, il fatto che funziona (con un senso) anche quando non troviamo distribuzioni note che corrispondano abbastanza bene ai dati. Il metodo parametrico invece è l’unico che può calcolare cose significative delle code, quando i dati non sono abbastanza ricchi rispetto alle code stesse.

**Esercizio (area corrispondente al punto).**

Invece che fissare una confidenza (o significatività, nel linguaggio dei test), qui 95% e vedere se il valore sperimentale (es. 0.8) cade nella regione di rifiuto, si può calcolare la probabilità che la distribuzione di COR abbia valori più estremi del valore sperimentale 0.8.

L’area dopo quel valore è pari a 1- la cumulativa in quel punto, calcolata con pnorm(0.8,…). Ecco il risultato:

> 1-pnorm(0.8,mean(COR),sd(COR))

[1] 2.180131e-05

Questo è il **p-value del test** unilaterale. [Quello del test bilaterale è il doppio.]

Nota: in modo non-parametrico sarebbe stato impossibile arrivare a questa conclusione tramite il campione COR generato precedentemente: il massimo di COR è inferiore a 0.8, quindi la percentuale di punti di COR che supera 0.8 è zero. La stima del p-value sarebbe stata pari a 0; eventualmente a 1/N.

**Esercizio 2: trovare una banda attorno alla retta di regressione che contenga il 95% dei punti.**

Soluzione.

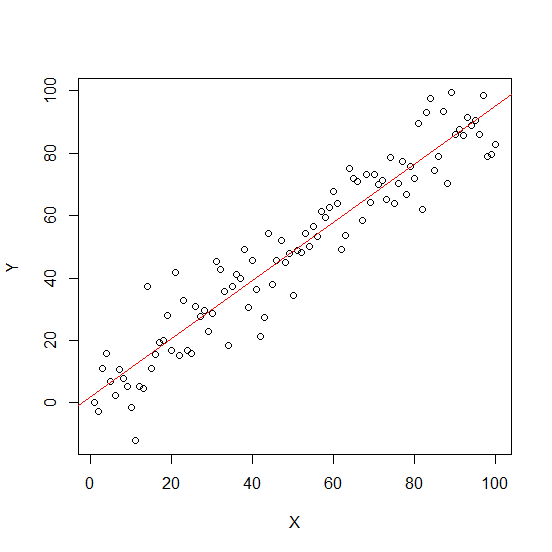
Prima risolviamo l’esercizio relativamente a dati artificiali, per fare più in fretta. Supponiamo di avere i dati

X=1:100; Y=X+10\*rnorm(100)

plot(X,Y)

REG= lm(Y~X)

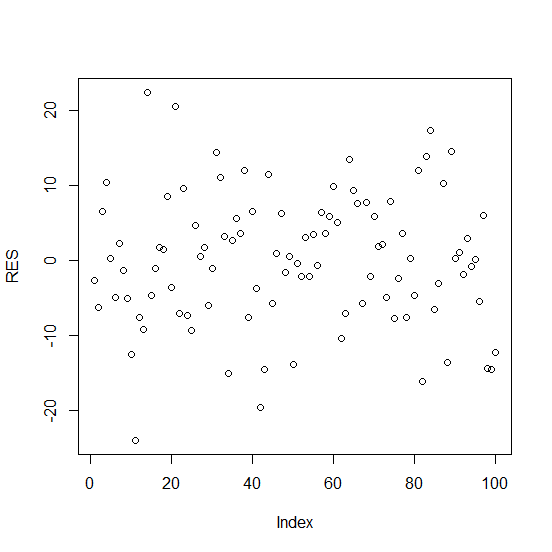
abline(REG,col=2)



Estraiamo i residui (gli errori):

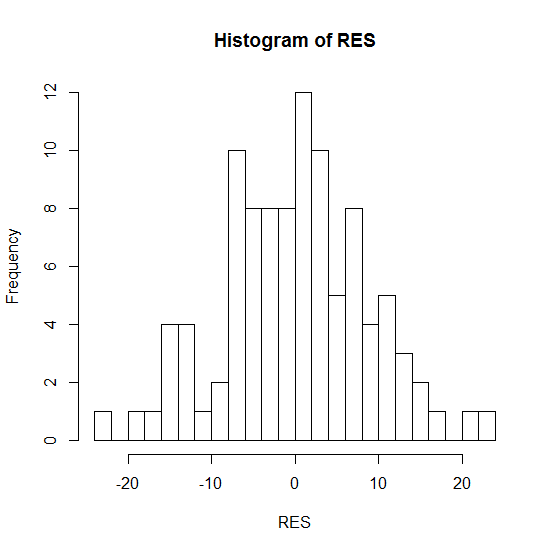
RES = residuals(REG)

plot(RES)



Qui sono sull’asse delle ordinate.

hist(RES,30)



Come sopra calcoliamo

quantile(RES,0.025)

quantile(RES,0.975)

e troviamo così un intervallo al 95% per i residui.

Dovendo invece tracciare una banda di confidenza, dobbiamo sottrarre e sommare ai parametri della retta di regressione una coppia (.,.) in cui la prima componente è lo scarto in intercetta, la seconda è zero (perché non si modifica la pendenza). Quindi

plot(X,Y)

COEF=coef(REG)

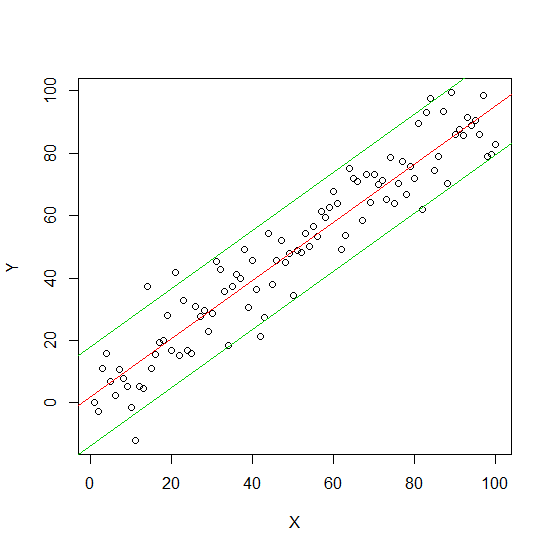
qmin= quantile(RES,0.025)

qmax= quantile(RES,0.975)

abline(REG,col=2)

abline(COEF+c(qmin,0),col=3)

abline(COEF+c(qmax,0),col=3)



**Esercizio: ripetere la cosa per la tabella pesi-altezze.**

A <- read.table ("clipboard")

1.67 61 04 0

1.78 62 26 0

1.78 72 5 0

1.84 75 2 0

1.82 76 12 0

1.80 72 2 0

1.95 91 3 0

1.81 80 9 0

1.80 70 9 0

1.86 77 6 0

1.68 65 9 0

1.84 64 8 0

1.72 62 1 0

1.76 65 26 0

1.82 74 4 0

1.86 86 3 0

1.84 75 14 0

1.73 65 19 0

1.72 57 25 1

1.62 56 2 1

1.70 59 3 1

1.61 46 15 1

1.70 64 22 1

1.60 63 26 1

1.61 60 16 1

1.60 55 28 1

1.65 50 12 1

Alt = A[,1]

Peso = A[,2]

Data = A[,3]

Sesso = A[,4]

REG= lm(Peso~Alt)

RES = residuals(REG)

COEF=coef(REG)

qmin= quantile(RES,0.025)

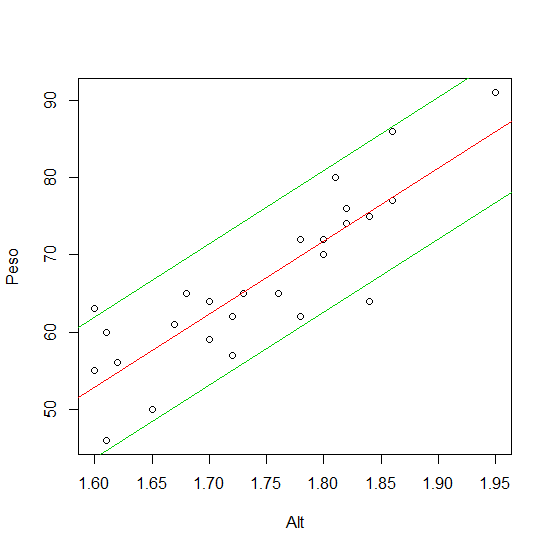
qmax= quantile(RES,0.975)

plot(Alt,Peso)

abline(REG,col=2)

abline(COEF+c(qmin,0),col=3)

abline(COEF+c(qmax,0),col=3)



1. **MATRICE DI CORRELAZIONE E PLOT DI UNA TABELLA**

Si esegua

cor(A)

riconoscendo nella prima posizione la correlazione già calcolata tra peso ed altezza. Una curiosità piuttosto inspiegabile: la correlazione tra altezza e giorno di nascita è significativa rispetto al test di indipendenza eseguito al 95%. Si noti la naturale correlazione tra sesso ed altezza.

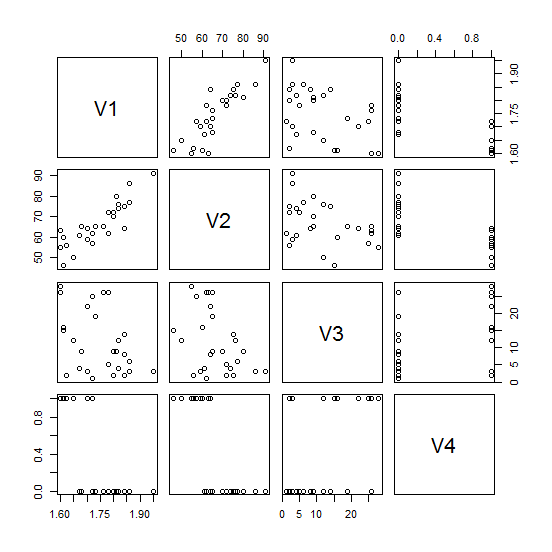
Le posizioni sul triangolo superiore sono quelle che portano informazione; per il resto, sulla diagonale c’è 1 di default ed il triangolo inferiore è la copia dell’altro, per simmetria.

Il comando cov(A) produce invece le covarianze. I valori della covarianza sono poco interpretabili per motivi di scala, come si vede ad es. dal valore tra sesso ed altezza. Invece sulla diagonale ci sono le varianze, a volte utili.

Il comando

plot(A)

raffigura i plot a due a due, dando un’impressione visiva delle varie correlazioni:



Si notino i grafici insoliti quando è coinvolta la variabile sesso, che è discreta. Sono però interessanti e mostrano a colpo d’occhio delle informazioni.