

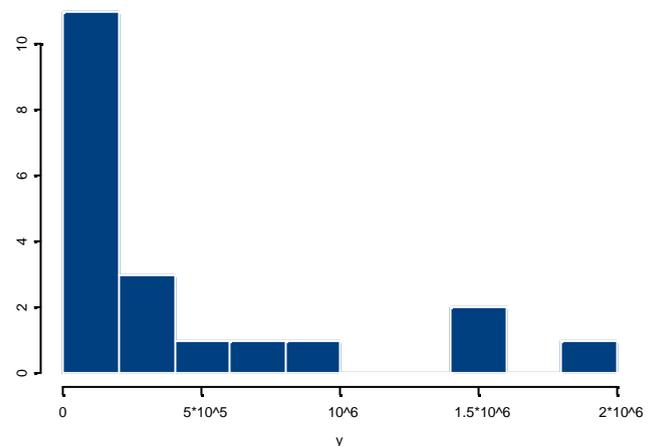
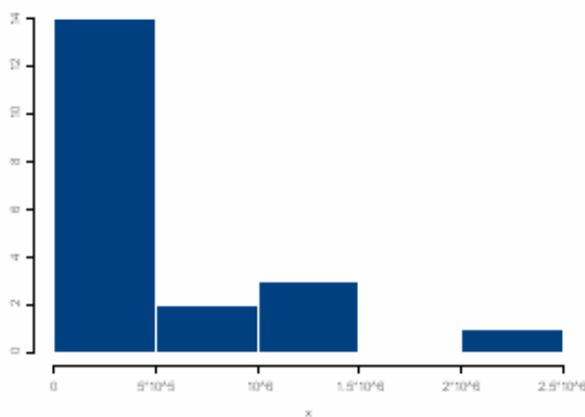
Laboratorio di analisi statistica dei dati – Appello luglio 2004

Esercizio n°1

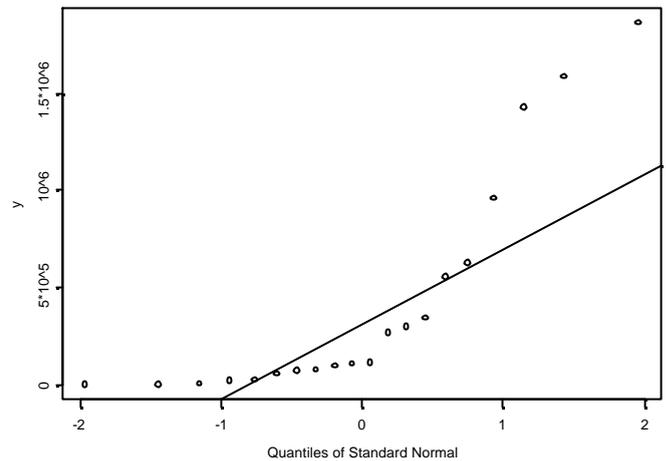
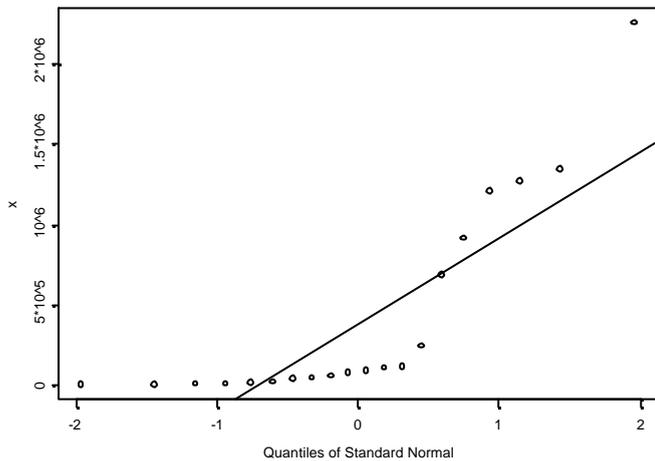
a- Nella figura sottostante vi è l'acquisizione dei dati

```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
Linear
S-PLUS : Copyright (c) 1988, 2002 Insightful Corp.
S : Copyright Lucent Technologies, Inc.
Professional Edition Version 6.1.2 Release 1 for Microsoft Windows : 200
2
Working data will be in C:\Programmi\Insightful\splus61\users\salvo
> x<-scan()
1: 120500 1343500 4100 95000 1207600 17100 59100 916100
9: 2253100 3500 686000 78600 1270500 40700 11900 13500
17: 22300 245300 46900 111000
21:
> y<-scan()
1: 109500 629000 1700 298600 959500 9900 78000 1423800
9: 1580400 1400 555200 26500 1856000 98200 82500 60700
17: 30400 346800 117100 267200
21:
> |
```

Dobbiamo verificare se la X e la Y seguano una distribuzione normale. Per fare ciò analizziamo gli istogrammi e il qqnorm.



A sinistra abbiamo l'istogramma della x e a destra quello della y



A destra abbiamo il qqnorm per la X e a sinistra per la Y

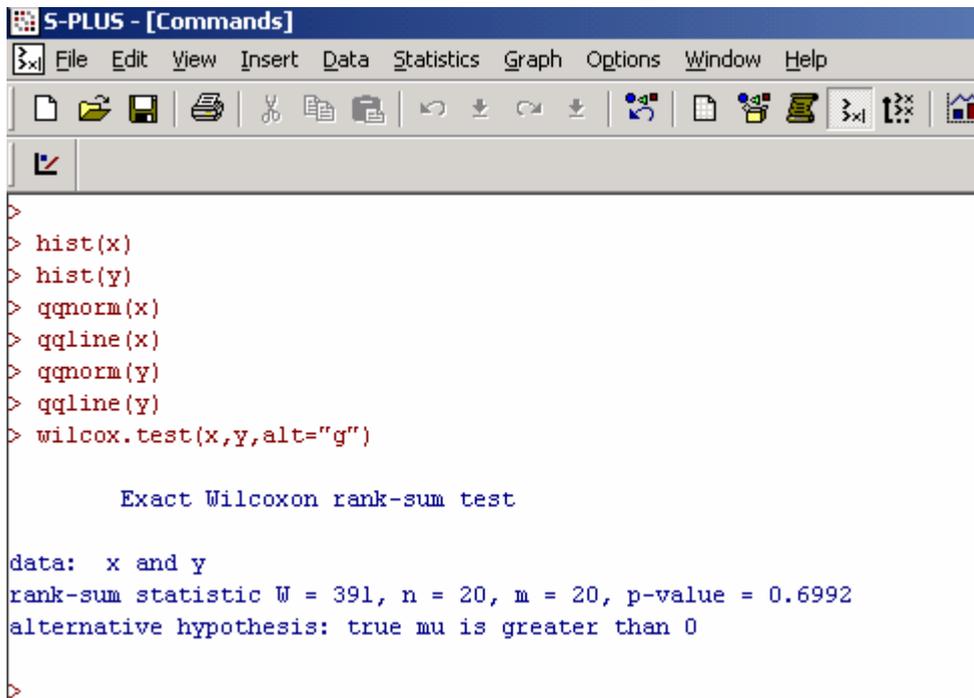
Si vede subito che sia X che Y non hanno una distribuzione normale, infatti dal qqnorm si deduce che i punti non sono disposti lungo la retta.

Poiché sia X che Y non seguono la distribuzione normale e $n, m < 30$ (con n = numero del campione di X e m = numero del campione di Y), ho deciso di utilizzare il test non parametrico di Wilcoxon Della Somma Dei Ranghi.

Il test sarà:

$$H_0 : m_x \approx m_y \text{ contro } H_1 : m_x \neq m_y$$

con $\alpha = 0,05$



```
>
> hist(x)
> hist(y)
> qqnorm(x)
> qqline(x)
> qqnorm(y)
> qqline(y)
> wilcox.test(x,y,alt="g")

      Exact Wilcoxon rank-sum test

data:  x and y
rank-sum statistic W = 391, n = 20, m = 20, p-value = 0.6992
alternative hypothesis: true mu is greater than 0
>
```

si rigetta H_0 se il p-value ? ?

in questo caso $p\text{-value} = 0.6992 > 0,05$, quindi si accetta H_0

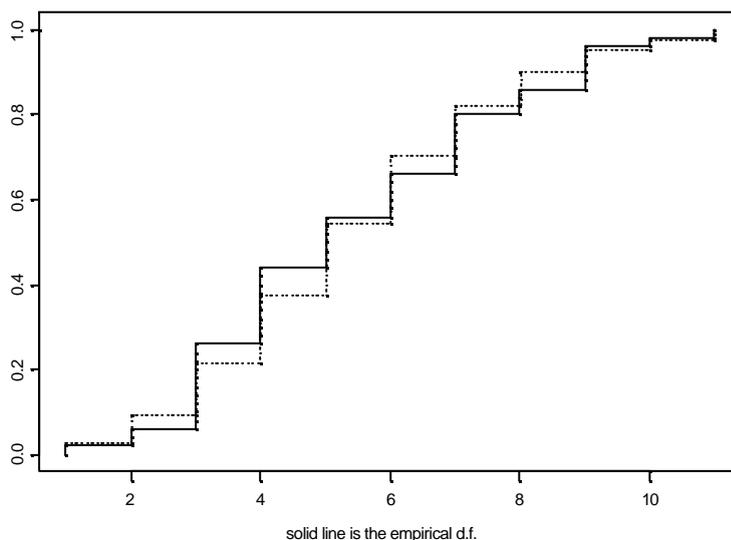
b- Nella figura sottostante vi è l'acquisizione dei dati

```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
S-PLUS : Copyright (c) 1988, 2002 Insightful Corp.
S : Copyright Lucent Technologies, Inc.
Professional Edition Version 6.1.2 Release 1 for Microsoft Windows : 200
2
Working data will be in C:\Programmi\Insightful\splus61\users\salvo
> x<-scan()
1: 7 4 3 6 4 4 5 3 5 3
11: 5 5 3 2 5 4 3 3 7 6
21: 6 4 3 11 9 6 7 4 5 4
31: 7 3 2 8 6 7 4 1 9 8
41: 4 8 9 3 9 7 7 9 3 10
51:
|
```

Compariamo ora la c.d.f della distribuzione di probabilità assegnata con la quella della poisson Usando la funzione presente nel S-PLUS

```
cdf.compare(x,distribution="pois",lambda=mean(x))
```

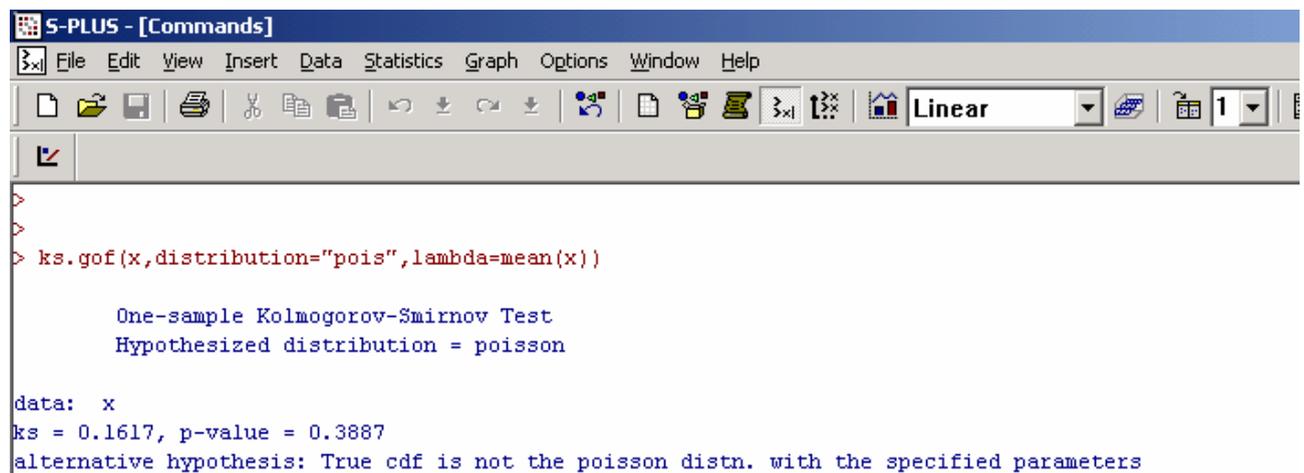
Empirical and Hypothesized poisson CDFs



Per verificare che la v.a. X ha una distribuzione di poisson uso il test :
 χ^2 - bontà di adattamento.

```
> chisq.gof(x,distribution="pois",n.param.est=1,lambda=mean(x))
Problem in chisq.gof(x, distribution = "pois", n.param.est = 1, lambda = mean(x)): You must supply the cutpoints for a c
Use traceback() to see the call stack
> chisq.gof(x,cut.points=30,distribution="pois",n.param.est=1,lambda=mean(x))
Problem in tabulate(num, n.classes): nbins must be at least as large as max(bin)
Use traceback() to see the call stack
> |
```

Applicando la funzione `chisq.gof(...)` , si sono riscontrati alcuni problemi , all'apparenza irrisolvibili. Quindi ho preferito usare il test di Kolmogorov – Smirnov per la bontà di adattamento.



```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
Linear 1
>
>
> ks.gof(x,distribution="pois",lambda=mean(x))

      One-sample Kolmogorov-Smirnov Test
      Hypothesized distribution = poisson

data: x
ks = 0.1617, p-value = 0.3887
alternative hypothesis: True cdf is not the poisson distn. with the specified parameters
```

H_0 : X segue una distribuzione di Poisson

H_1 : X non segue una distribuzione di Poisson

con $\alpha = 0,05$

si rigetta H_0 se il p-value $< \alpha$

in questo caso $p\text{-value} = 0.3887 < 0,05$, quindi si accetta H_1

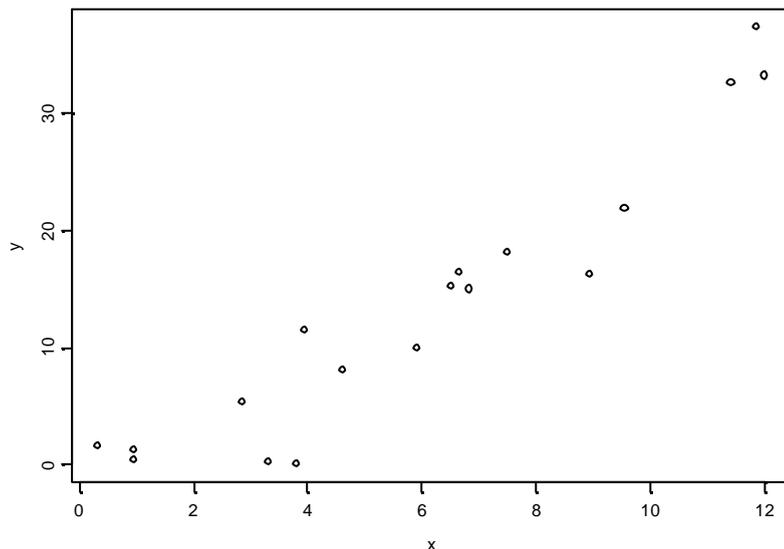
Esercizio n°2

Nella figura sottostante vi è l'acquisizione dei dati e la chiamata alla funzione per creare un diagramma di dispersione.

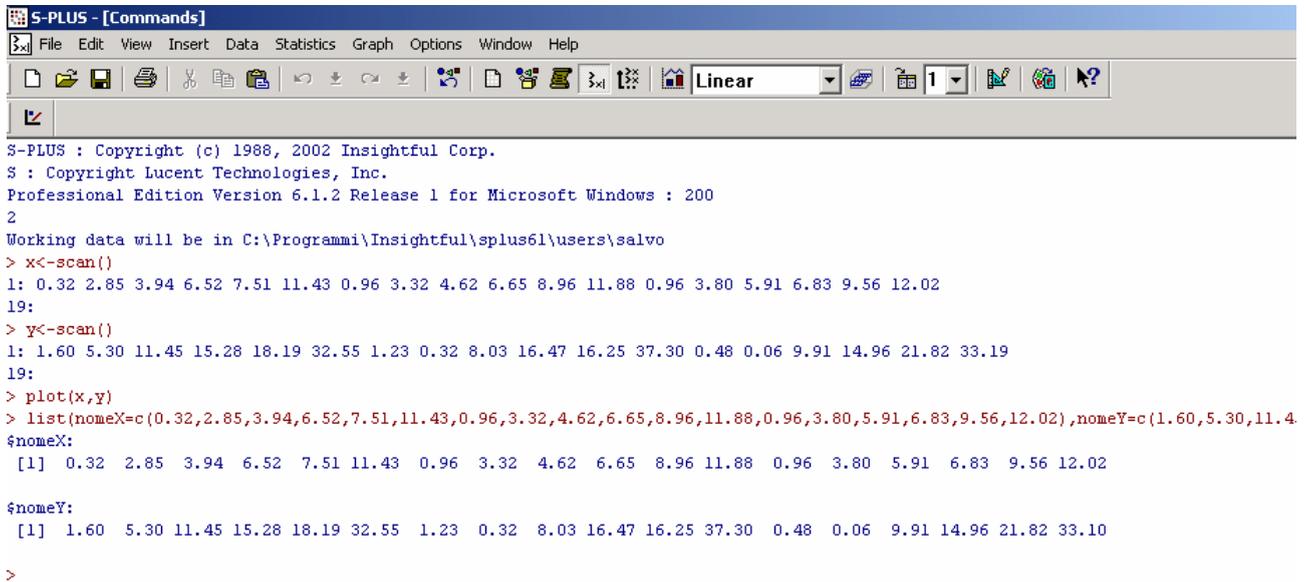
```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
Linear 1
S-PLUS : Copyright (c) 1988, 2002 Insightful Corp.
S : Copyright Lucent Technologies, Inc.
Professional Edition Version 6.1.2 Release 1 for Microsoft Windows : 200
2
Working data will be in C:\Programmi\Insightful\splus61\users\salvo
> x<-scan()
1: 0.32 2.85 3.94 6.52 7.51 11.43 0.96 3.32 4.62 6.65 8.96 11.88 0.96 3.80 5.91 6.83 9.56 12.02
19:
> y<-scan()
1: 1.60 5.30 11.45 15.28 18.19 32.55 1.23 0.32 8.03 16.47 16.25 37.30 0.48 0.06 9.91 14.96 21.82 33.19
19:
> plot(x,y)
>
```

Questo è il diagramma che se ne ricava

Diagramma di dispersione di x e y



Ora definiamo la lista degli elementi di x e di y del modello di regressione

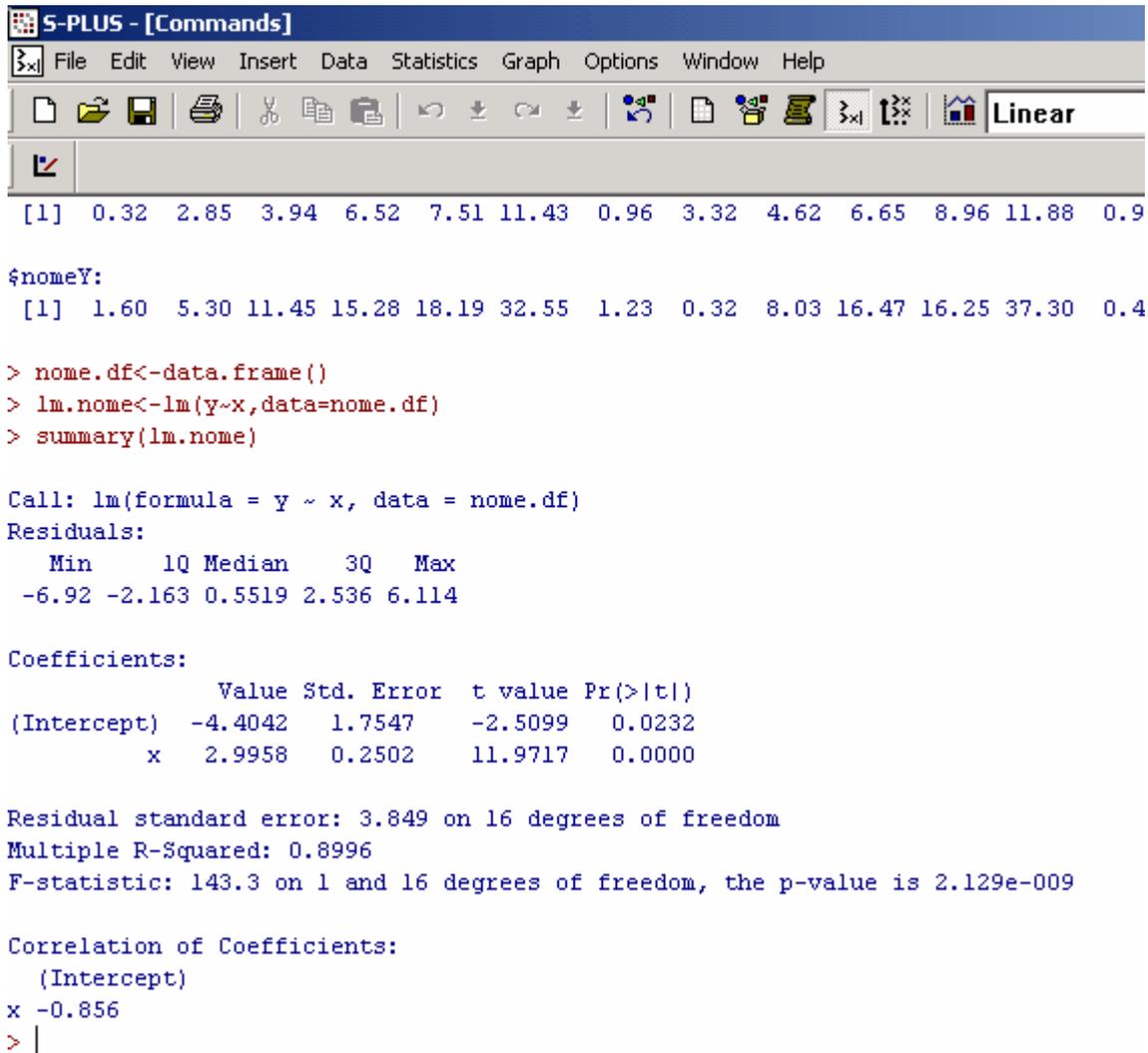


```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
Linear 1
S-PLUS : Copyright (c) 1988, 2002 Insightful Corp.
S : Copyright Lucent Technologies, Inc.
Professional Edition Version 6.1.2 Release 1 for Microsoft Windows : 200
2
Working data will be in C:\Programmi\Insightful\spplus61\users\saltro
> x<-scan()
1: 0.32 2.85 3.94 6.52 7.51 11.43 0.96 3.32 4.62 6.65 8.96 11.88 0.96 3.80 5.91 6.83 9.56 12.02
19:
> y<-scan()
1: 1.60 5.30 11.45 15.28 18.19 32.55 1.23 0.32 8.03 16.47 16.25 37.30 0.48 0.06 9.91 14.96 21.82 33.19
19:
> plot(x,y)
> list(nomeX=c(0.32,2.85,3.94,6.52,7.51,11.43,0.96,3.32,4.62,6.65,8.96,11.88,0.96,3.80,5.91,6.83,9.56,12.02),nomeY=c(1.60,5.30,11.4
$nomeX:
 [1] 0.32 2.85 3.94 6.52 7.51 11.43 0.96 3.32 4.62 6.65 8.96 11.88 0.96 3.80 5.91 6.83 9.56 12.02

$nomeY:
 [1] 1.60 5.30 11.45 15.28 18.19 32.55 1.23 0.32 8.03 16.47 16.25 37.30 0.48 0.06 9.91 14.96 21.82 33.10

>
```

ora effettuiamo l'analisi della regressione lineare



```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
[1] 0.32 2.85 3.94 6.52 7.51 11.43 0.96 3.32 4.62 6.65 8.96 11.88 0.9
$nomeY:
[1] 1.60 5.30 11.45 15.28 18.19 32.55 1.23 0.32 8.03 16.47 16.25 37.30 0.4
> nome.df<-data.frame()
> lm.nome<-lm(y~x,data=nome.df)
> summary(lm.nome)

Call: lm(formula = y ~ x, data = nome.df)
Residuals:
    Min       1Q   Median       3Q      Max
-6.92 -2.163  0.5519  2.536  6.114

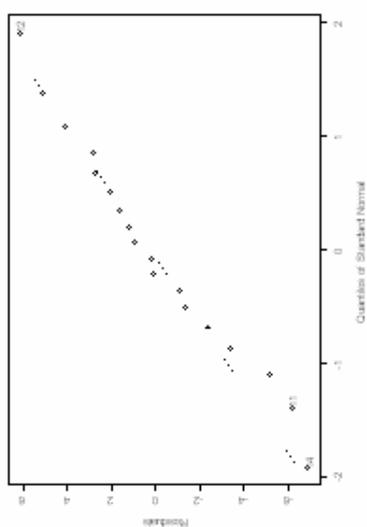
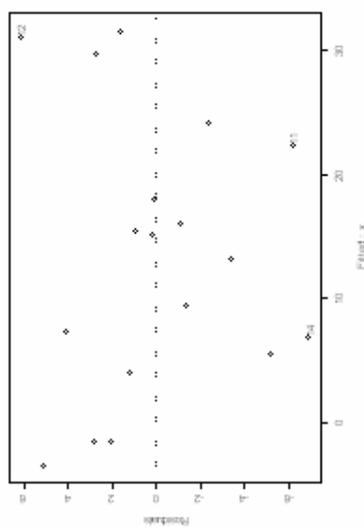
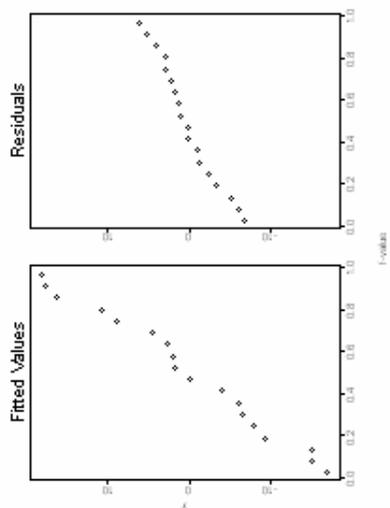
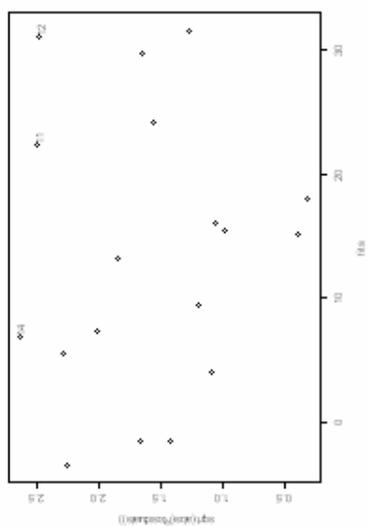
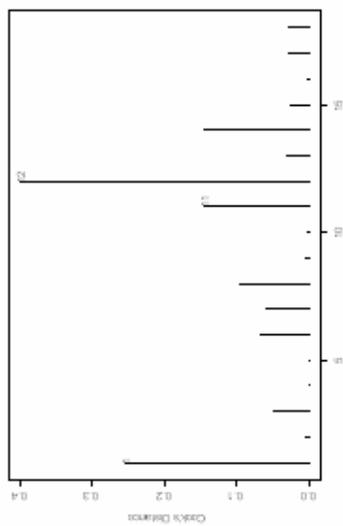
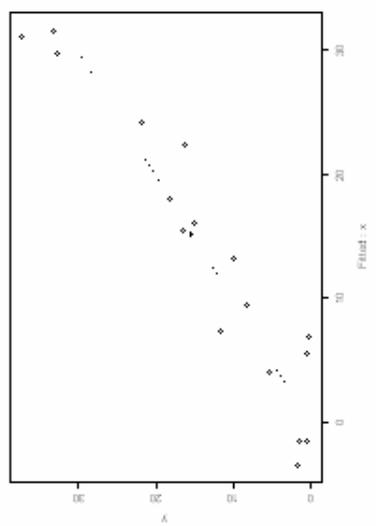
Coefficients:
            Value Std. Error t value Pr(>|t|)
(Intercept) -4.4042  1.7547  -2.5099  0.0232
            x   2.9958  0.2502  11.9717  0.0000

Residual standard error: 3.849 on 16 degrees of freedom
Multiple R-Squared: 0.8996
F-statistic: 143.3 on 1 and 16 degrees of freedom, the p-value is 2.129e-009

Correlation of Coefficients:
  (Intercept)
x -0.856
> |
```

dove l'intercetta è β_0 e la x è β_1

ora effettuiamo l'analisi dei residui confrontando i diversi grafici che ho costruito



Ora costruisco un intervallo di fiducia per la media , usando la funzione pointwise che permette di avere l'estremo inferiore e superiore e il valore considerato.

```

S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
Linear
> plot(lm.nome)
> med<-c(0.32,2.85,3.94,6.52,7.51,11.43,0.96,3.32,4.62,6.65,8.96,11.88,0.96,3.80,5.9
> pointwise(predict(lm.nome,data.frame(nomeX=med),se.fit =T)
+ )
$upper:
      1      2      3      4      5      6      7      8      9
1.480838 7.645079 10.44761 17.80493 20.96391 34.60833 3.010772 8.837906 12.27211 18

      16      17      18
18.77494 27.94807 36.74003

$fit:
      1      2      3      4      5      6      7      8      9
-3.44552 4.133828 7.399238 15.12838 18.09421 29.8377 -1.528215 5.541849 9.436375 15

      16      17      18
16.05707 24.23558 31.60522

$lower:
      1      2      3      4      5      6      7      8      9
-8.371878 0.6225758 4.350862 12.45182 15.2245 25.06708 -6.067202 2.245791 6.600644

      16      17      18
13.3392 20.52308 26.47041

```

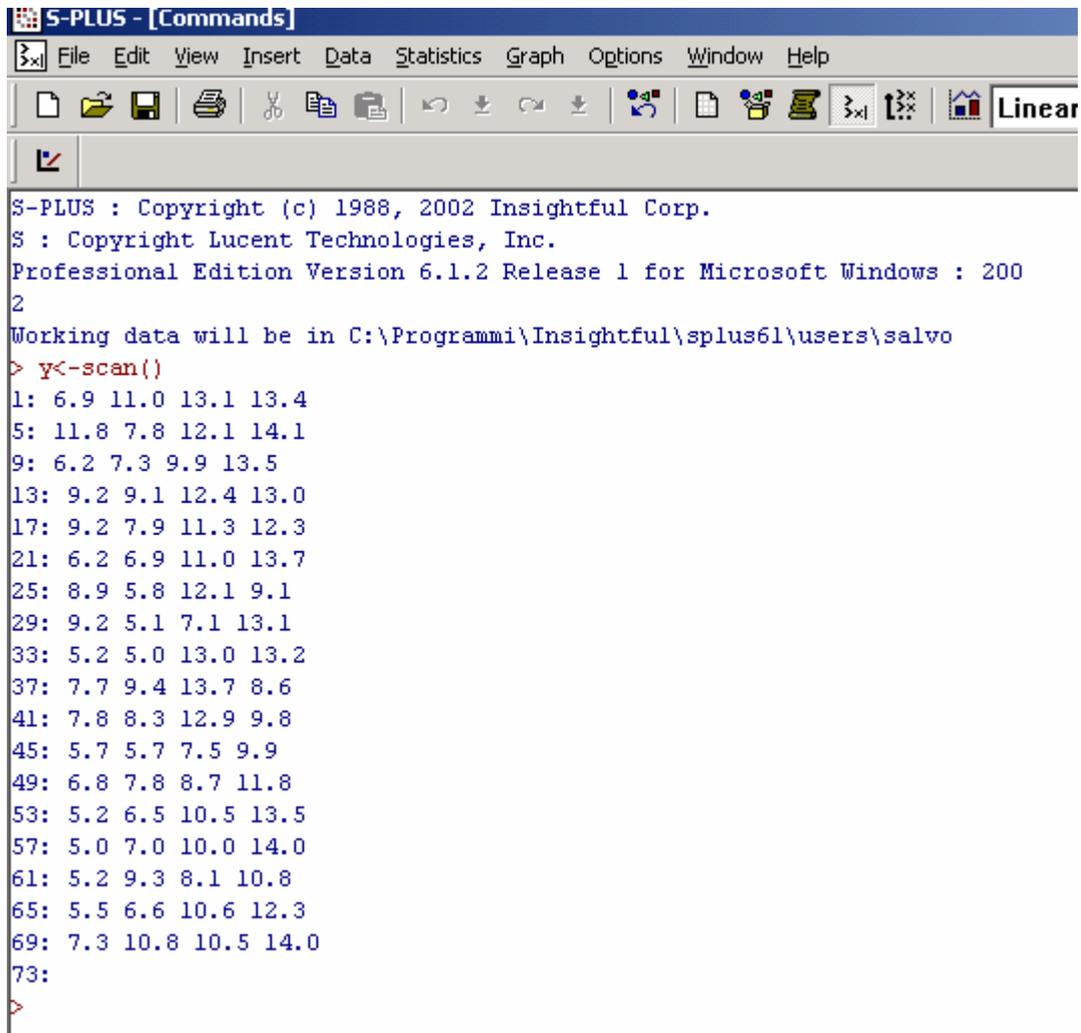
Ora costruiamo un intervallo di fiducia per μ_1

Sappiamo che l'intervallo di fiducia considerato è $\hat{\mu}_1 \pm t_{\frac{\alpha}{2}}(n-2) * (S \hat{\mu}_1)$, quindi si ha

$\hat{\mu}_1 = 2,9958$ $S \hat{\mu}_1 = 0,25502$ e $t_{\frac{\alpha}{2}}(n-2)$ è il quantile della t-student , che nel nostro caso , ha 16 gradi di libertà = 2,1199 . L'intervallo sarà quindi [2,4654;3,5262]

Esercizio n°3

Nella figura sottostante vi è l'acquisizione dei dati.



```
S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
[Icons]
S-PLUS : Copyright (c) 1988, 2002 Insightful Corp.
S : Copyright Lucent Technologies, Inc.
Professional Edition Version 6.1.2 Release 1 for Microsoft Windows : 200
2
Working data will be in C:\Programmi\Insightful\splus61\users\salvo
> y<-scan()
1: 6.9 11.0 13.1 13.4
5: 11.8 7.8 12.1 14.1
9: 6.2 7.3 9.9 13.5
13: 9.2 9.1 12.4 13.0
17: 9.2 7.9 11.3 12.3
21: 6.2 6.9 11.0 13.7
25: 8.9 5.8 12.1 9.1
29: 9.2 5.1 7.1 13.1
33: 5.2 5.0 13.0 13.2
37: 7.7 9.4 13.7 8.6
41: 7.8 8.3 12.9 9.8
45: 5.7 5.7 7.5 9.9
49: 6.8 7.8 8.7 11.8
53: 5.2 6.5 10.5 13.5
57: 5.0 7.0 10.0 14.0
61: 5.2 9.3 8.1 10.8
65: 5.5 6.6 10.6 12.3
69: 7.3 10.8 10.5 14.0
73:
>
```

Attenzione l'acquisizione dei dati non è corretta , consultare gli appunti per ulteriori chiarimenti

In questa immagine vediamo la determinazione del fattore A e del fattore B , la chiamata alle funzioni che disegnano il plot design , il plot factor . Dopo ciò vi è l'analisi della varianza e la visualizzazione dell'istogramma e del qqnorm

```

S-PLUS - [Commands]
File Edit View Insert Data Statistics Graph Options Window Help
[Icons]
>
> livelli <- list(fattA=c("W","X","Y","Z"),fattB=c("A","B","C"))
> livelli <- list(fattB=c("W","X","Y","Z"),fattA=c("A","B","C"))
> livelli
$fattB:
[1] "W" "X" "Y" "Z"

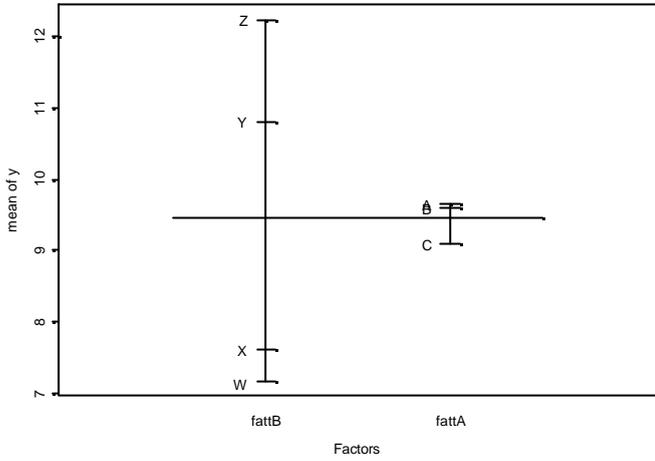
$fattA:
[1] "A" "B" "C"

> varianza.design<-fac.design(c(4,3),livelli)
> varianza.df<-data.frame(varianza.design,y)
> plot.design(varianza.df)
>
>
> plot.design(varianza.df,fun=median)
> plot.factor(varianza.df)
>
> aov.varianza<-aov(y~fattA*fattB,varianza.df)
>
> summary(aov.varianza)
          Df Sum of Sq  Mean Sq  F Value    Pr(F)
fattA     2    4.5211   2.2606   0.69214 0.5044540
fattB     3   325.5749  108.5250  33.22842 0.0000000
fattA:fattB 6    36.7189    6.1198   1.87378 0.1001698
Residuals 60   195.9617    3.2660

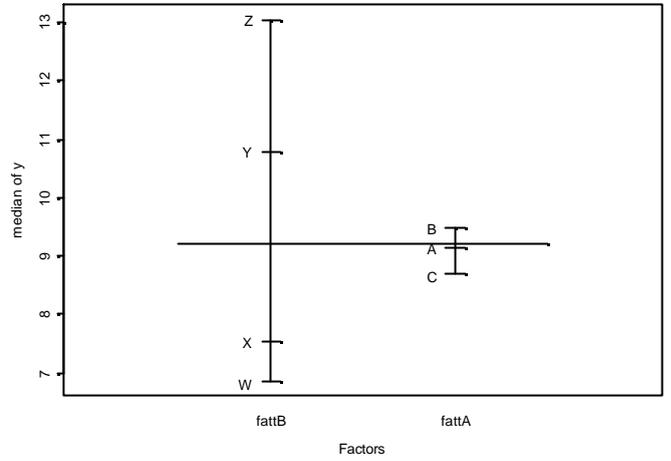
>
> hist(resid(aov.varianza))
> qqnorm(resid(aov.varianza))
> qqline(resid(aov.varianza))

```

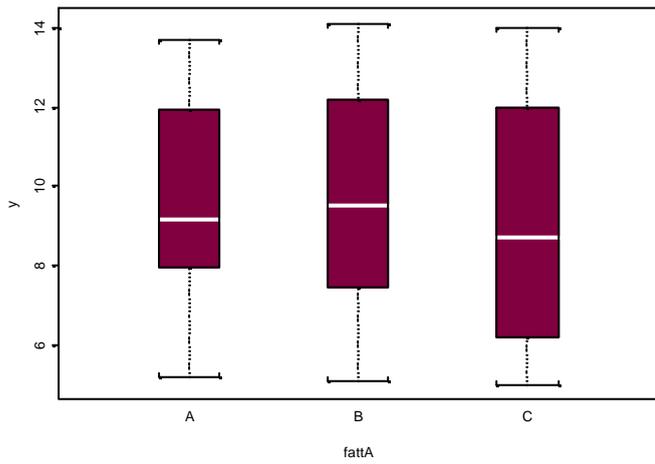
Ora mostriamo i grafici sopraelencati



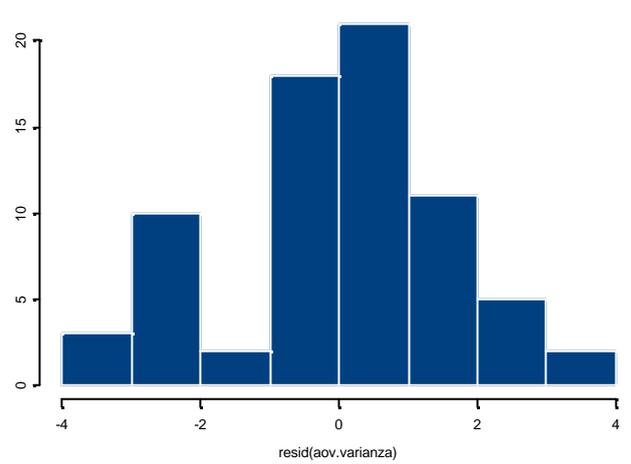
Il plot design (il valore della y è la media)



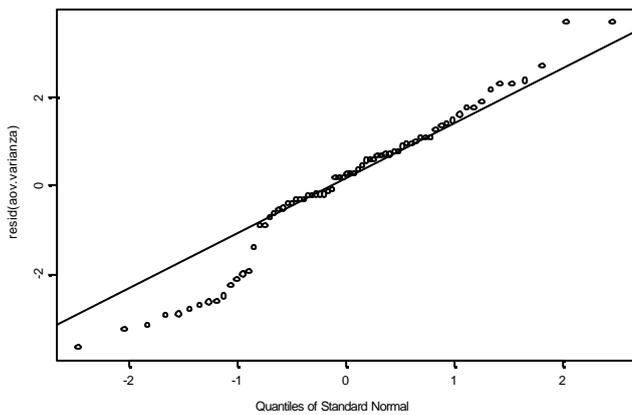
Il plot design (il valore della y è la mediana)



Il plot factor



L'istogramma



Il qqnorm

