* **MEAN MEDIAN MODE**

> x=c(18,19,19,19,19,20,20,20,20,20,21,21,21,21,22,23,24,27,30,36)

> mean(x) #mean [1] 22

> median(x) #median [1] 20.5

> y=x[x md=median(y)

> md [1] 20

> xr=table(x) #mode

> mode=which(xr==max(xr))

> mode

20

3

* **MEAN MEDIAN MODE (Frequency Distribution)**

> mid=seq(147.5,182.5,5)

> mid [1] 147.5 152.5 157.5 162.5 167.5 172.5 177.5 182.5

> f=c(4,6,28,58,64,30,5,5)

> fr.distr=data.frame(mid,f)

> fr.distr

mid f

1. 147.5 4
2. 152.5 6
3. 157.5 28
4. 162.5 58
5. 167.5 64
6. 172.5 30
7. 177.5 5
8. 182.5 5

**Mean:--**

> mean=(sum(mid\*f))/sum(f)

> mean

[1] 165.175

Median

> midx=seq(147.5,182.5,5)

> frequency=c(4,6,28,58,64,30,5,5)

> fr.dist<-data.frame(midx,frequency)

> fr.dist

midx frequency

1 147.5 4

2 152.5 6

3 157.5 28

4 162.5 58

5 167.5 64

6 172.5 30

7 177.5 5

8 182.5 5

> cl=cumsum(frequency)

> cl

[1] 4 10 38 96 160 190 195 200

> n=sum(frequency)

> n

[1] 200

> ml=min(which(cl>=n/2)) # The serial number of the median class

> ml

[1] 5

> h=5

> h [1] 5

> f=frequency[ml] #frequency of the median class

> f [1] 64

> c=cl[ml-1] # cumulative frequency of the median class

> c [1] 96

> l=mid[ml]-h/2

> l [1] 165

> median=l+(((n/2)-c)/f)\*h #median

> median

[1] 165.3125

**Mode:-**

> m=which(frequency==max(frequency)) #serial number of the median class

> m

[1] 5

> fm=frequency[m] # frequency of the modal class

> fm

[1] 64

> f1=frequency[m-1] # frequency of the pre modal class

> f2=frequency[m+1] # frequency of the post modal class

> f1

[1] 58

> f2

[1] 30

> l=midx[m]-h/2

> l

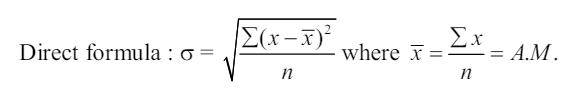
[1] 165

> mode=l+((fm-f1)/(2\*fm-f1-f2))\*h

> mode

[1] 165.75

* **STANDARD DEVIATION**

****

> x=c(1.2,1.4,1.3,1.6,1.0,1.5,1.7,1.1,1.2,1.3)

> x

[1] 1.2 1.4 1.3 1.6 1.0 1.5 1.7 1.1 1.2 1.3

> summary(x)

Min. 1st Qu. Median Mean 3rd Qu. Max.

1.000 1.200 1.300 1.330 1.475 1.700

> range=1.7-1.0 #range

> range

[1] 0.7

> var(x) #variance

[1] 0.049

> sd=sqrt(var(x)) #standard deviation

> sd

[1] 0.2213594 There is no separate command for Quartile deviation Mean deviation .We have to evaluate the expression

>cqd=(1.475-1.2)/(1.475+1.2) #coefficient of quartile deviation # Mean deviation about Mean

> y=(x-mean(x))

> y

[1] -0.13 0.07 -0.03 0.27 -0.33 0.17 0.37 -0.23 -0.13 -0.03

> y=abs(y)

> y [1] 0.13 0.07 0.03 0.27 0.33 0.17 0.37 0.23 0.13 0.03

> mdl=sum(y)/length(y)

> mdl [1] 0.176 #Mean deviation about Median

> z =abs(x-median(x))

> md2=sum(z)/length(z)

> md2

[1] 0.17 Mean deviation about Mode # in this Problem ,it is a bi-model series (Mode is not possible)

* **MEASURES OF SKEWNESS,KURTOSIS**

>x=c(15.9,16.2,16.0,15.6,16.2,15.9,16.0,15.6,15.6,16.0,16.2,15.6,15.9,16.2,15.6,16.2,15.8,16 .0,15.8,15.9,16.2,15.8,15.8,16.2,16.0,15.9,16.2,16.2,16.0,15.6)

> x

[1] 15.9 16.2 16.0 15.6 16.2 15.9 16.0 15.6 15.6 16.0 16.2 15.6 15.9 16.2 15.6

[16] 16.2 15.8 16.0 15.8 15.9 16.2 15.8 15.8 16.2 16.0 15.9 16.2 16.2 16.0 15.6

> n=length(x)

> n

[1] 30

> mean=mean(x)

> mean

[1] 15.93667

> m4=sum((x-mean)^4)/n

> m4

[1] 0.004062022

> m2=var(x)

> m2

[1] 0.0486092

> beta2=m4/(m2^2)

> beta2

[1] 1.719117

> gam2=beta2-3

> gam2

[1] -1.280883

* **CORRELATION**

> x=c(23,27,28,28,29,30,31,33,35,36)

> y=c(18,20,22,27,21,29,27,29,28,29)

> var(x)

[1] 15.33333

> var(y)

[1] 18.22222

> var(x,y)

[1] 13.66667

> r=var(x,y)/sqrt(var(x)\*var(y))

> r

[1] 0.8176052

Or

> cor(x,y)

[1] 0.8176052

Or

> cor.test(x,y)

Or

> cor.test(x,y,method="pearson") Pearson's product-moment correlation data: x and y

t = 4.0164, df = 8, p-value = 0.003861

alternative hypothesis: true correlation is not equal to 0 95 percent confidence interval: 0.3874142 0.9554034

sample estimates: cor 0.8176052

* **REGRESSION**

> Y=c(110,80,70,120,150,90,70,120)

> X1=c(30,40,20,50,60,40,20,60)

> X2=c(11,10,7,15,19,12,8,14)

> input\_data=data.frame(Y,X1,X2)

> input\_data

Y X1 X2

1 110 30 11

2 80 40 10

3 70 20 7

4 120 50 15

5 150 60 19

6 90 40 12

7 70 20 8

8 120 60 14

> RegModel <- lm(Y~X1+X2, data=input\_data)

> RegModel

Call:

lm(formula = Y ~ X1 + X2, data = input\_data)

Coefficients:

(Intercept) X1 X2

16.8314 0.2442 7.8488

> summary(RegModel)

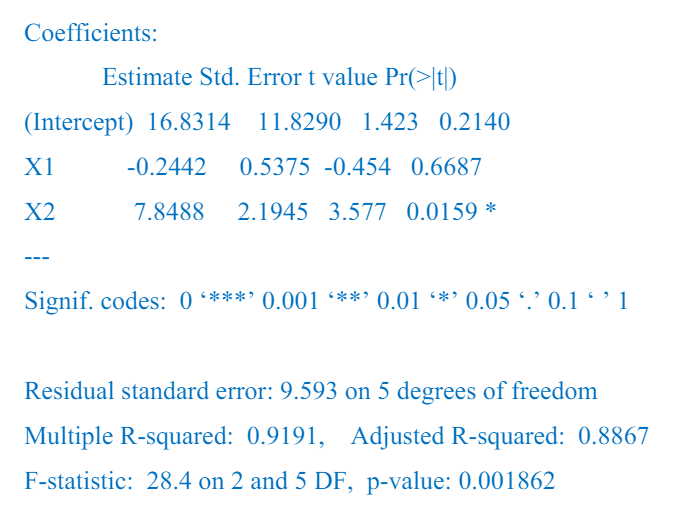
Call:

lm(formula = Y ~ X1 + X2, data = input\_data)

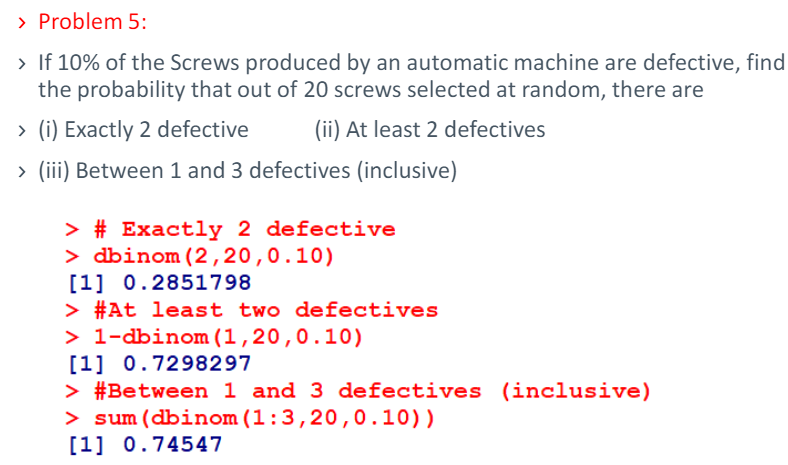
Residuals:

1 2 3 4 5 6 7 8

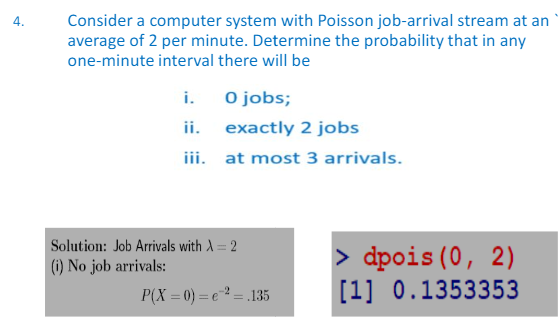
14.157 -5.552 3.110 -2.355 -1.308 -11.250 -4.738 7.936

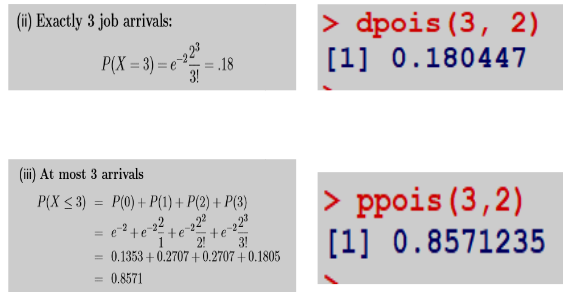
****

* **BINOMIAL DISTRIBUTION**

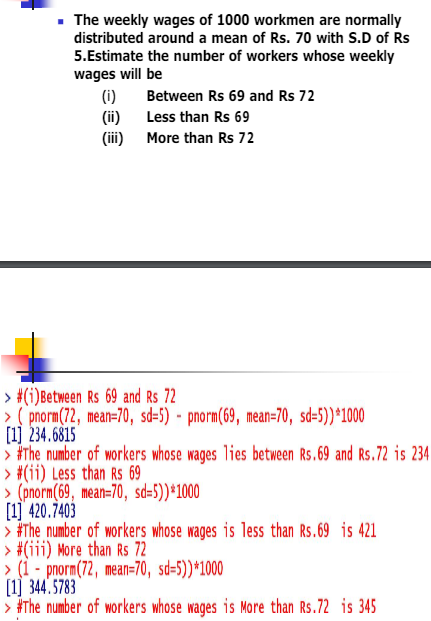
****

* **POISSON DISTRIBUTION**

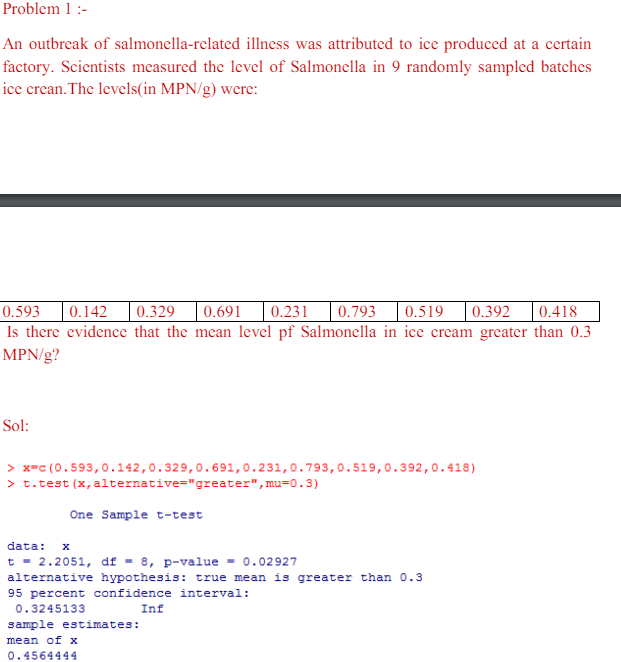
****

****

* **NORMAL DISTRIBUTION**

****

* **T-test**

****