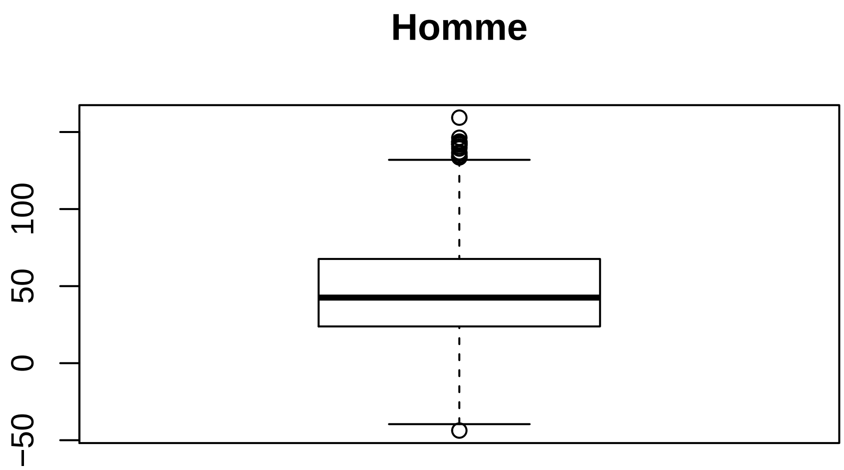
projet

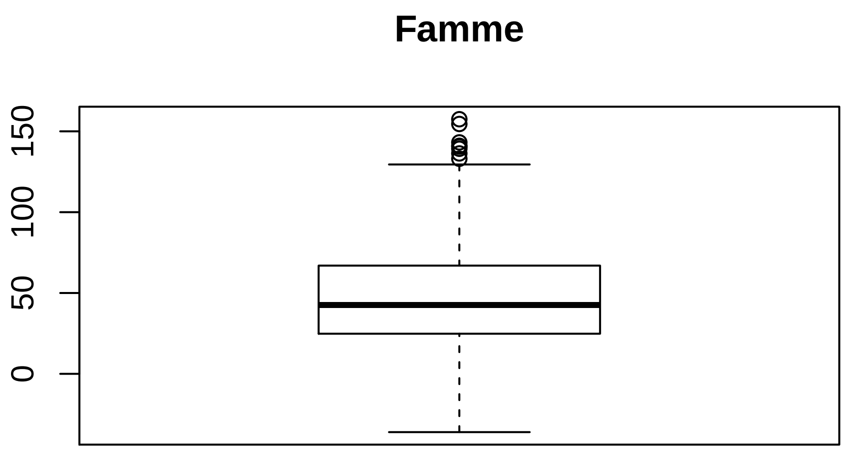
1 Statistique Descriptives

(a) Tracer les boîte des femmes et les hommes

```
rm(list = ls())
  data=read.csv("DB_binome_56.csv",header = TRUE)
  hommes <-subset(data,data['Sexe']==0)
  femmes<-subset(data,data['Sexe']==1)
par(fig=c(0,0.8,0,0.8))
  boxplot(hommes['Peche'],TRUE,main = "Homme")
```



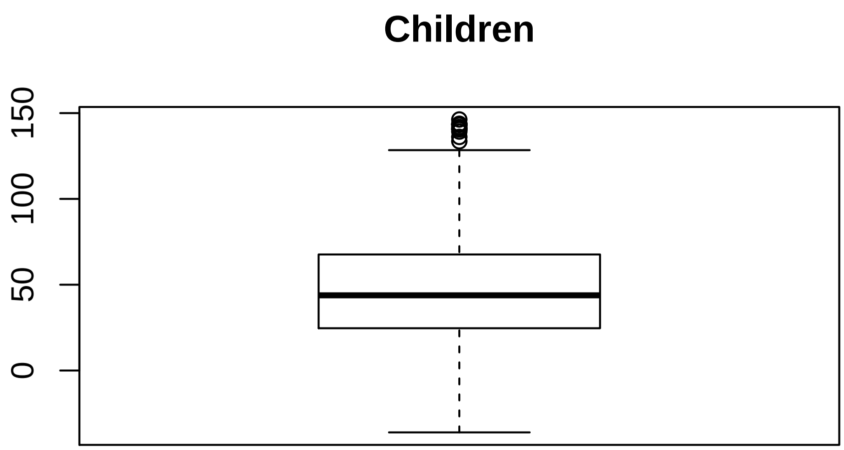

```
 par(fig=c(0,0.8,0,0.8))
  boxplot(femmes['Peche'],TRUE,main = "Famme")
```



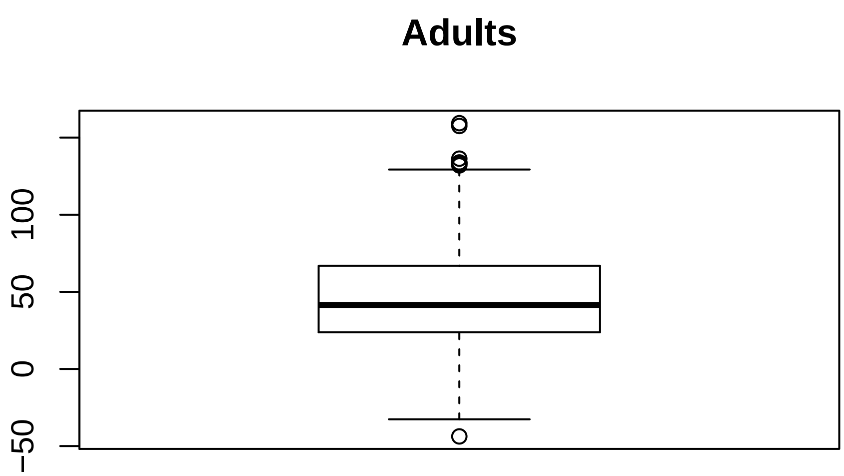
On remarque que il n'y a pasune grande defference entre les hommes et les femmes.

(b)Tracer la quantité de péche en fonction de la tranche d'age

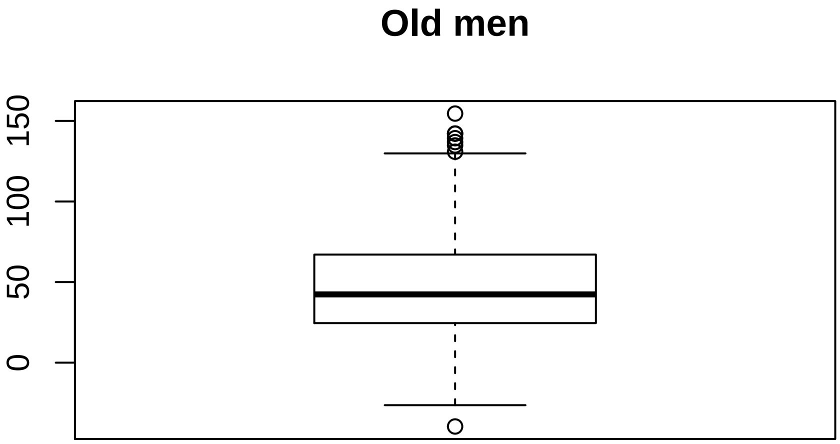
```r
rm(list = ls())
  data=read.csv("DB_binome_56.csv",header = TRUE)
  children<-subset(data,data['Age']==0)
  adults<-subset(data,data['Age']==1)
  oldmen<-subset(data,data['Age']==2)
 par(fig=c(0,0.8,0,0.8))
  boxplot(children['Peche'],TRUE,main = "Children")
```




```r
 par(fig=c(0,0.8,0,0.8))
  boxplot(adults['Peche'],TRUE,main = "Adults")
```



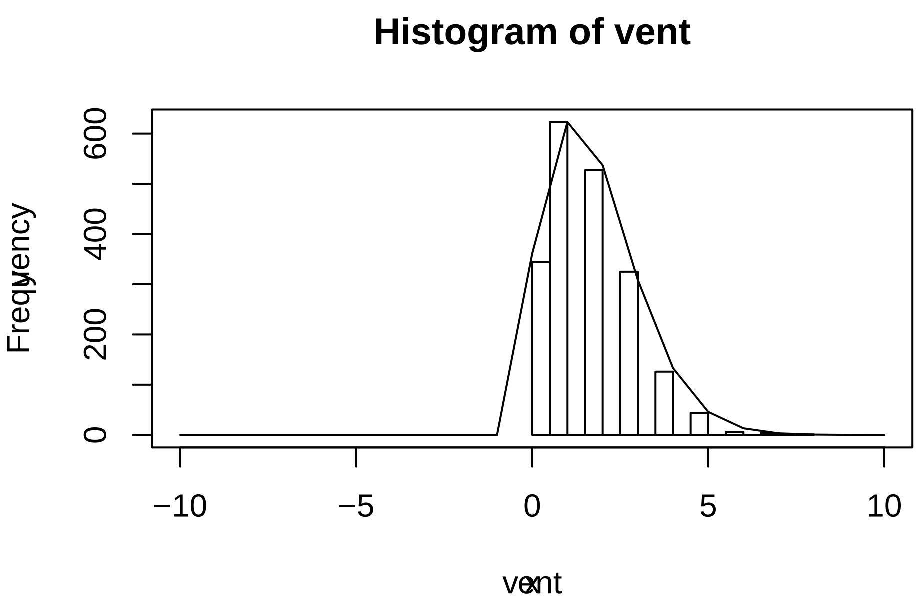

```r
 par(fig=c(0,0.8,0,0.8))
  boxplot(oldmen['Peche'],TRUE,main = "Old men")
```



On remarque que il n'y a pasune grande defference entre les gens qui ont les age defferents.

(c)Tracer l'histogramme l'intensité de vent

```
  vent<-data[,'Noeuds']
par(fig=c(0,0.8,0,0.8))
hist(vent,freq=TRUE,xlim=c(-10,10))
lambda=mean(vent)

x=seq(-10,10)
y<-dpois(x,lambda)
par(new=TRUE)
par(fig=c(0,0.8,0,0.8))
plot(x,y,type="l",xaxt="n",yaxt="n")
```

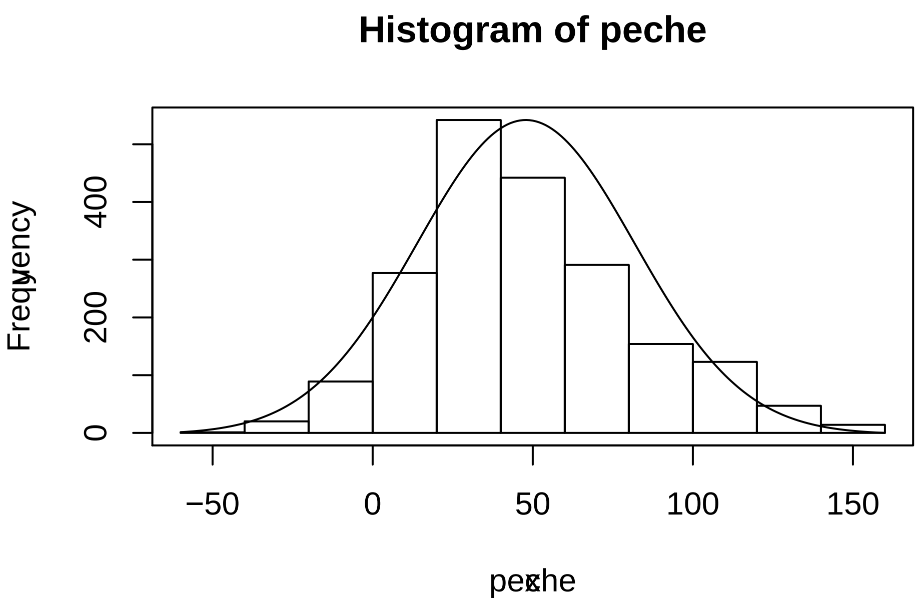


Elle suive la loi de Poisson .

On le justifie par la courbe de "dpoi"

(d)Tracer l'histogramme la quantité de péche

```
  peche<-data[,'Peche']
par(fig=c(0,0.8,0,0.8))
hist(peche,freq=TRUE,xlim=c(-60,160))

mu=mean(peche)
delta=sqrt(var(peche))
x=seq(-60,160)
y<-dnorm(x,mu,delta)
par(new=TRUE)
par(fig=c(0,0.8,0,0.8))
plot(x,y,type="l",xaxt="n",yaxt="n")
```



Elle suive la loi de Normale .

On le justifie par la courbe de "dnorm"

2

(a)Verifier les observation

```
nb = length(vent)
lambda = mean(vent)
echant =  rpois(nb , lambda)
ks.test(vent, echant)
```

```
## Warning in ks.test(vent, echant): p-value will be approximate in the
## presence of ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  vent and echant
## D = 0.014, p-value = 0.9895
## alternative hypothesis: two-sided
```

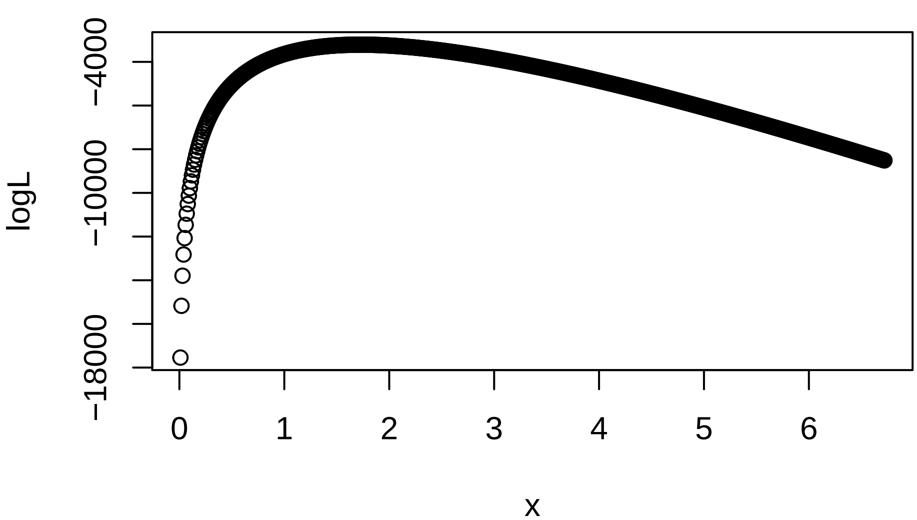
Donc on remarque la p-value est presque $1 > 0.05$, donc il suivre la loi poisson

(b)La vraisemblance

$L = e^{-n\lambda} \prod_{i=0}^{n} \frac{\lambda^{x_i}}{x_i!}$ $logL = -\lambda n + \sum_{i=0}^{n}(x_i log\lambda - log x_i!)$ $logL' = -n + \frac{1}{\lambda}\sum_{i=0}^{n}(x_i) = 0$ $\widehat{\lambda} = \frac{1}{n}\sum_{i=0}^{n} x_i = \bar{x}$

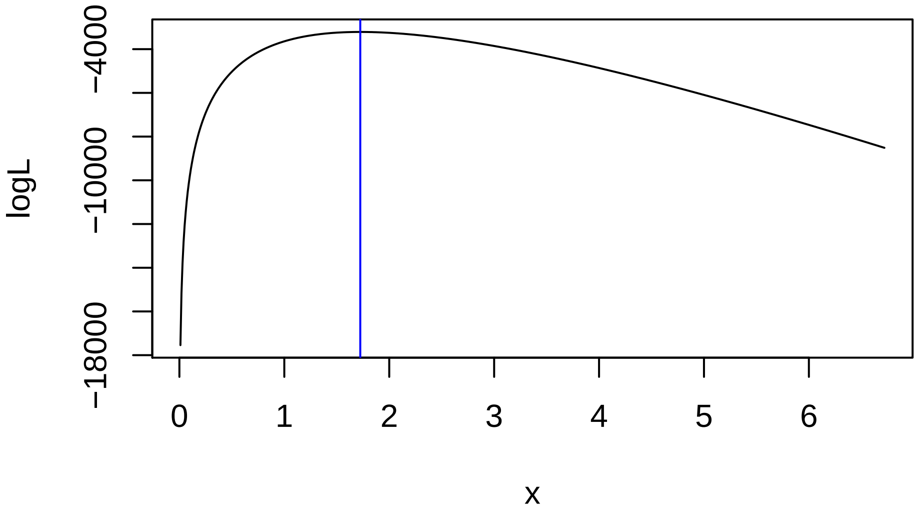
(c) Tracer La log-vraisemblance et déterminer le maximun de vraisemblance

```
lambda=mean(vent)
x<-seq(0.01,lambda+5,0.01)
logL=vector(length=length(x))
n<-length(vent)

max=-100000
maxX=0

for( i in seq(1,length(x))){
  sum=0
  for(j in seq(1,length(vent))){
  sum = sum + vent[j]*log(x[i]) - log(factorial(vent[j]))
  }

  logL[i]=-n*x[i] +sum

  if(logL[i]>max){
    max=logL[i]
    maxX=x[i]
  }
}
par(fig=c(0,0.8,0,0.8))
plot(x,logL,xaxt="lambda",yaxt="logL")
```

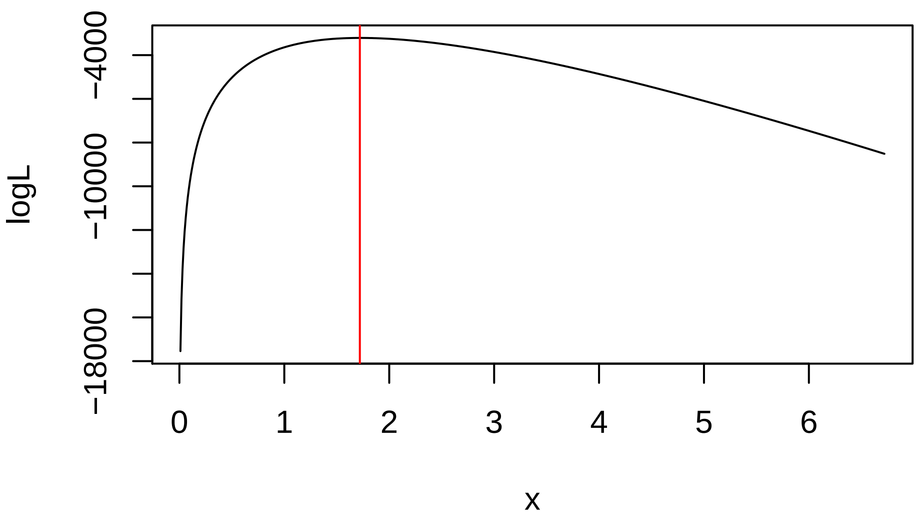


#(d) donner une estimation du parametreλ ,et faire apparaitre sur le graphique. je utilise deux facos pour estimer le parametre

```r
par(fig=c(0,0.8,0,0.8))
plot(x,logL,type="l",xaxt="lambda",yaxt="logL")
par(new=TRUE)
par(fig=c(0,0.8,0,0.8))
abline(v=lambda,col = "blue")
```




```r
par(fig=c(0,0.8,0,0.8))
plot(x,logL,type="l",xaxt="lambda",yaxt="logL")
par(new=TRUE)
par(fig=c(0,0.8,0,0.8))
abline(v=maxX,col = "red")
```




```r
cat("1er",lambda,"2eme",maxX)
```

```
## 1er 1.724 2eme 1.72
```

(e)écrire la vraisemblance théorique du modèle de la quantité de péche

$$f(x_i,\lambda_i) = \frac{1}{\sqrt{2\pi}\sigma} exp(\frac{-(x_i - \frac{100}{1+\lambda_i})^2}{2\sigma^2})$$

$$f_\delta(x) = \sum_{i=0}^{m} P_{pois}(y=\lambda_i) f(x,\lambda_i)$$

pour lambda = 1.724 , ici les

$$\lambda_i > 11$$

sont omises du calcul .

$$L = \prod_{i=0}^{n} f_\sigma(x_i)$$

$$logL = \sum_{i=0}^{n} log(f_\sigma(x_i))$$

```
repartition<-function(xi,sigma){
  m<-qpois(p=0.999999,lambda=lambda)
  proba<-dpois(seq(0,m,1),lambda)
   f=vector(length = m)
    sum<-0
  for(j in seq(1,m)){

       norm<- 1/sqrt(2*pi)/sigma*exp(-((xi-100/(j))/sigma)^2/2)

    f[j]=proba[j]*norm
    sum=sum+f[j]
  }
    return(sum)
}
```

(f)TCL

On pose X= $\sum_{i=0}^{\infty} log f(x)$ donc $\frac{X-E(X)}{\sqrt{V(X)}}$ ~$N(0,1)$

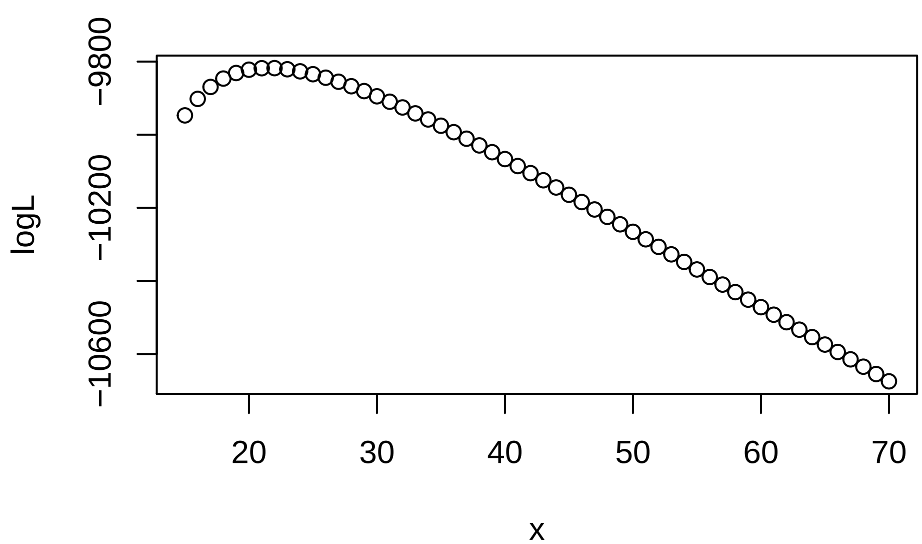
(g)Déterminer le maximum de vraissemblance du parametre

δ, tracer la log-vraissemblance de l'échantillon en fonction de la valeur du δ avec chaqu'une des méthodes

$$logL = \sum_{i=0}^{n} log(f_\delta(x_i))$$

$$logL' = \sum_{i=0}^{n} \frac{f'_\delta(x)}{f_\delta(x)}$$

```r
sum<-0
x<-seq(15,70,1)
logL=vector(length=length(x))
  max=-100000
maxX=0
second=-100000
secondX=0

for(j in seq(1,length(x))){
  sum<-0
  for(i in seq(1,length(peche))){
      sum=sum+log(repartition(peche[i],x[j]))
      }

  logL[j]=sum

  if(logL[j]>max){
    max=logL[j]
    maxX=x[j]
  }else if (logL[j]>second){
    second=logL[j]
    secondX=x[j]
  }
}
par(fig=c(0,0.8,0,0.8))
plot(x,logL)
```



#(h)Donner une estimation du parametre δ avec les deux approches et la faire apparaitre sur les graphiques

```r
cat("les deux approches",maxX,secondX)
```

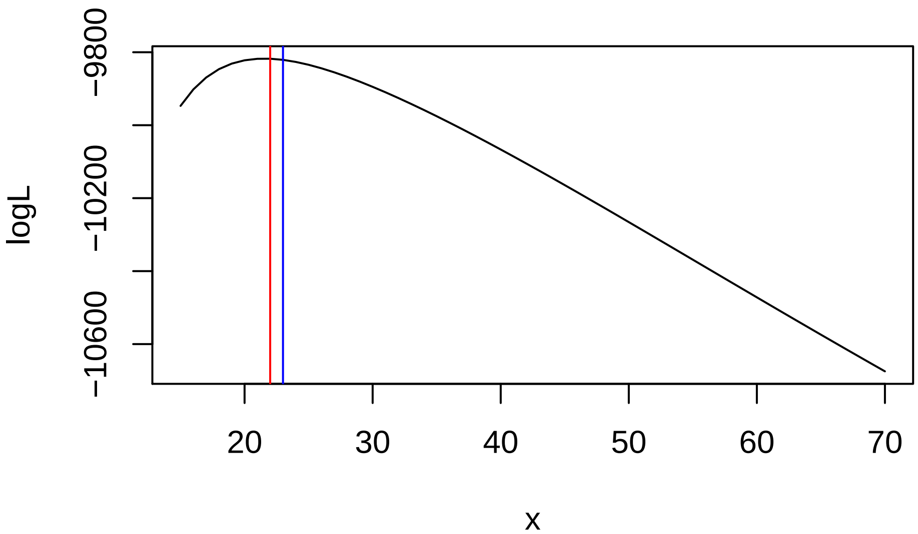
```
## les deux approches 22 23
```

```r
cat("estimateur=",maxX/2+secondX/2)
```

```
## estimateur= 22.5
```

```r
par(fig=c(0,0.8,0,0.8))
plot(x,logL,type="l",xaxt="lambda",yaxt="logL")
```

```r
par(new=TRUE)
par(fig=c(0,0.8,0,0.8))
abline(v=maxX,col = "red")
par(new=TRUE)
par(fig=c(0,0.8,0,0.8))
abline(v=secondX,col = "blue")
```



3

(1)

On a deja $\widehat{\lambda} = \frac{1}{n}\sum_{i=0}^{n} x_i = \bar{x}$ donc on a $\frac{\frac{1}{n}\sum_{i=0}^{n} x_i - E[\frac{1}{n}\sum_{i=0}^{n} x_i]}{\sqrt{D[\frac{1}{n}\sum_{i=0}^{n} x_i]}} \sim N(0,1)$ donc $\sqrt{n}\frac{\widehat{\lambda}-E[x]}{D[x]} \sim N(0,1)$ avec $E[x] = \lambda$ $D[x] = \lambda$ on pose que α, $P(\frac{\alpha}{2} < \sqrt{n}\frac{\widehat{\lambda}-E[x]}{D[x]} < 1-\frac{\alpha}{2}) = 1-\alpha$, donc on a $\widehat{\lambda} \in [\lambda+\frac{\sqrt{\lambda}}{\sqrt{n}}F^{-1}(\frac{\alpha}{2}), \lambda+\frac{\sqrt{\lambda}}{\sqrt{n}}F^{-1}(1-\frac{\alpha}{2})]$

(2)

```r
Intervalle=function(lambda, alpha, n) {
  theta1 = lambda + sqrt(lambda/n) * qnorm(alpha/2, mean = 0, sd = 1, log = FALSE)
  theta2 = lambda + sqrt(lambda/n) * qnorm(1-alpha/2, mean = 0, sd = 1, log = FALSE)
  paste("Intervalle de confiance : [ ", theta1, ", ", theta2, "]", sep = "")
}
Intervalle(lambda, 0.05, 2000)
```

```
## [1] "Intervalle de confiance : [ 1.66645577784488, 1.78154422215512]"
```

4 Tests

(a)

On pose que H0:la quantité de pêche ne dépend pas du sexe H1:la quantité de péche dépend du sexe

```r
quantiteH <-subset(data,data['Sexe']==1)[,'Peche']

quantiteF <-subset(data,data['Sexe']==0)[,'Peche']

t.test(quantiteH,quantiteF,paired = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  quantiteH and quantiteF
## t = -0.13155, df = 1817.6, p-value = 0.8954
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.219464  2.814720
## sample estimates:
## mean of x mean of y
##  47.73338  47.93575
```

On a p-value = 0.8953 > 0.05, donc on ne rejette pas l'hypothèse 0. ###(b) Test :

$$H_0 : \sigma = \sigma_0 = 20$$

et

$$H_1 : \sigma \neq \sigma_0$$

$$P_{H_0}(\sum_{i=1}^{n}(x_i - \mu_i)^2 > K_\alpha) = P_{H_0}(\frac{\sum_{i=1}^{n}(x_i - \mu_i)^2}{\sigma_0^2} > \frac{K_\alpha}{\sigma_0^2}) = 1 - F_{\mathcal{X}^2}(\frac{K_\alpha}{\sigma_0^2}) = \alpha$$

si

$$\sum_{i=1}^{n}(x_i - \mu_i)^2 > \delta_0^2 F_{\mathcal{X}^2}^{-1}(1 - \alpha)$$

, on rejette le Hypothese

```r
trois_Noeuds <-subset(data,data['Noeuds']==3)[,'Peche']
  sum=0
  for(i in seq(1,length(trois_Noeuds))){
   sum=sum+(trois_Noeuds[i]-100/(3+1))^2
  }
   cat(sum,"inferieur a",20^2*qchisq(p=0.999, df=length(trois_Noeuds)),"superieur a",20^2*qchisq(p=0.00
```

```
## 161348.9 inferieur a 163806 superieur a 100749.8
```

```r
  if(sum>20^2*qchisq(p=1, df=length(trois_Noeuds))){
  print("on rejette H0")
}else if(sum<20^2*qchisq(p=0.005, df=length(trois_Noeuds))){
  print("on rejette H0")
}else{
  print("on ne rejette pas H0")
}
```

```
## [1] "on ne rejette pas H0"
```

On remarque que pour ces datas , H0 n'est pas rejet quand alpha = 0.001