TP2

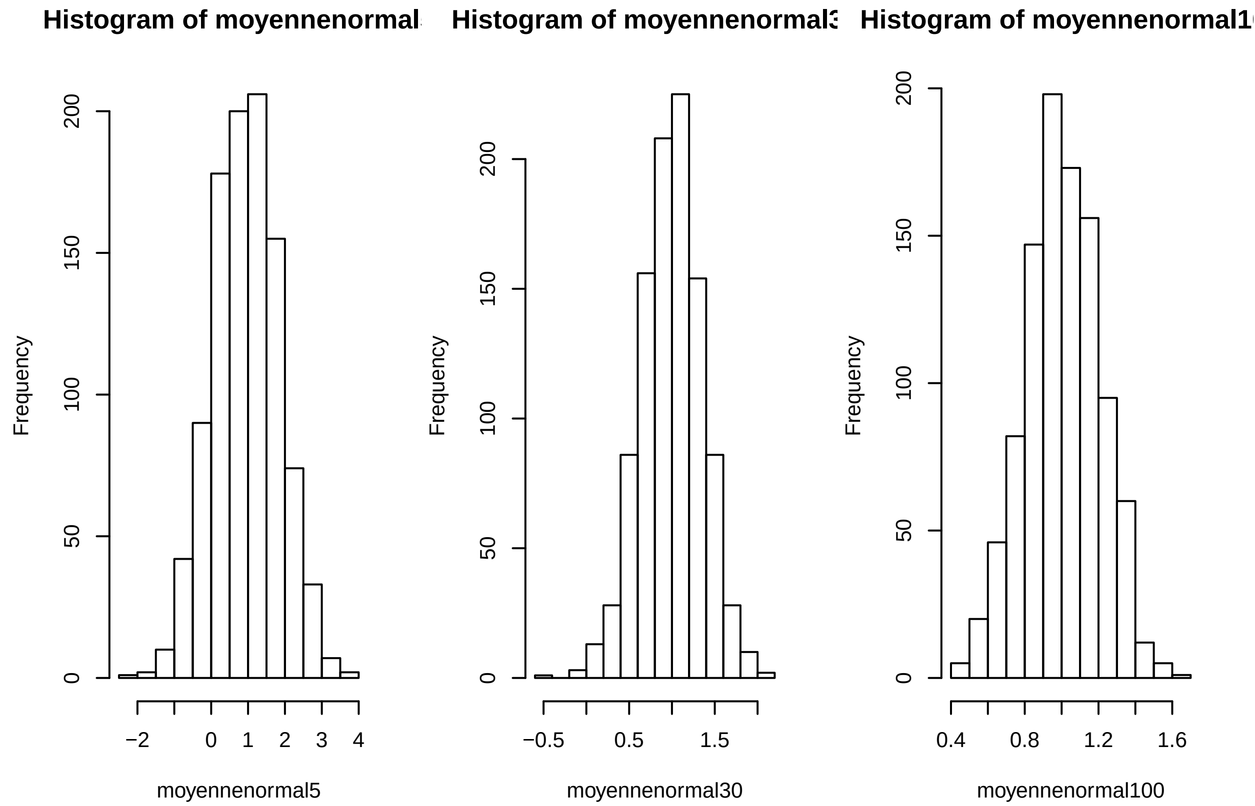
Gautier Poursin

28/02/2020

Echantillon, Théorème Central Limite, Estimation Monte Carlo

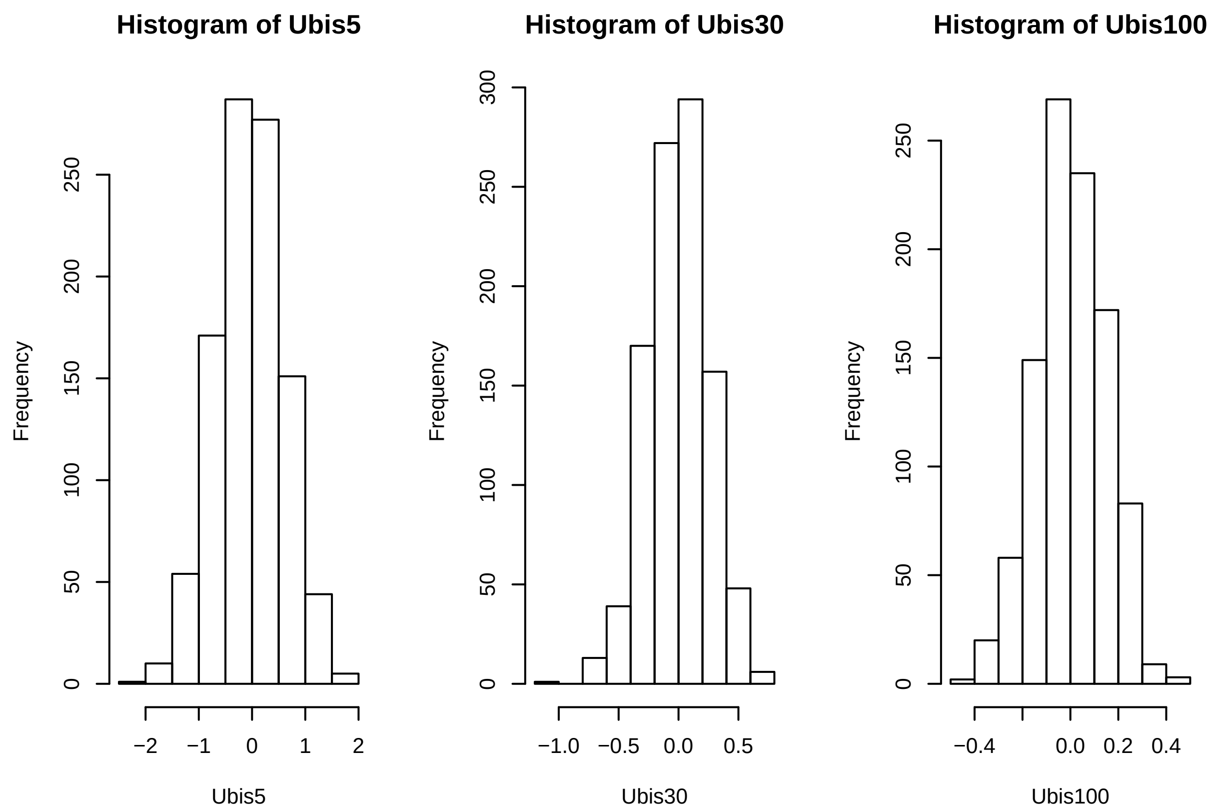
Loi Normale

```r
normal5<-c(rnorm(5000,mean=1, sd=2))
normal30<-c(rnorm(30000, mean=1, sd=2))
normal100<-c(rnorm(100000,mean=1, sd=2))

par(mfrow=c(1,3))
Normal5<-matrix(normal5,1000,5)
Normal30<-matrix(normal30,1000,30)
Normal100<-matrix(normal100,1000,100)

moyennenormal5= rep(1,1000)
for(i in 1:1000){
  moyennenormal5[i] = mean(Normal5[i,])}

hist(moyennenormal5)

variancenormal5=rep(1,1000)
for(i in 1:1000){
  variancenormal5[i]=var(Normal5[i,])
}

moyennenormal30= rep(1,1000)
for(i in 1:1000){
  moyennenormal30[i] = mean(Normal30[i,])
}
hist(moyennenormal30)

variancenormal30=rep(1,1000)
for(i in 1:1000){
  variancenormal30[i]=var(Normal30[i,])
}

moyennenormal100= rep(1,1000)
for(i in 1:1000){
  moyennenormal100[i] = mean(Normal100[i,])
}
hist(moyennenormal100)
```




```r
variancenormal100=rep(1,1000)
for(i in 1:1000){
  variancenormal100[i]=var(Normal100[i,])
}
```

On choisi an=$\mathbb{E}[X]$, bn=$\sqrt{\mathbb{V}(X)}$ et on remarque que $Un, i \sim \mathcal{N}(0,1)$. On centre et on réduit la loi.



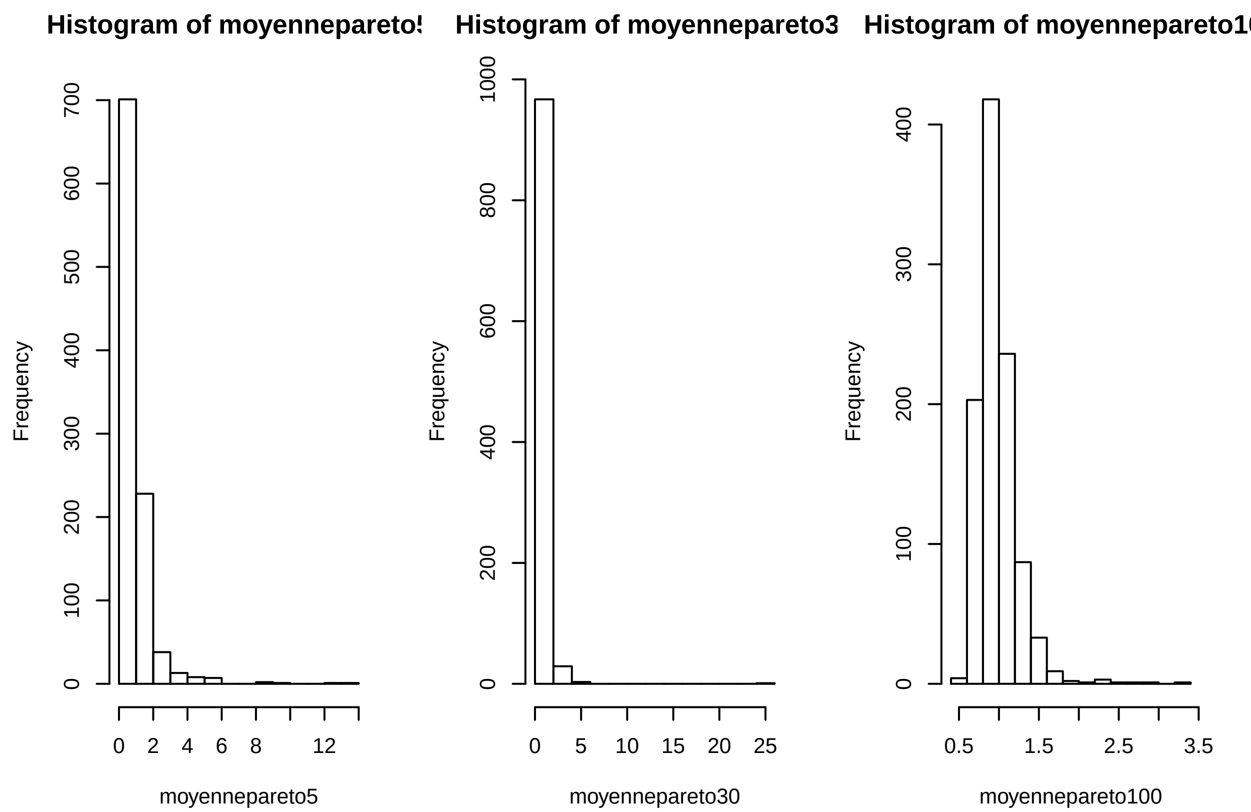
Loi de Pareto

```
library(rmutil)
```

```
##
## Attaching package: 'rmutil'

## The following object is masked from 'package:stats':
##
##     nobs

## The following objects are masked from 'package:base':
##
##     as.data.frame, units
```

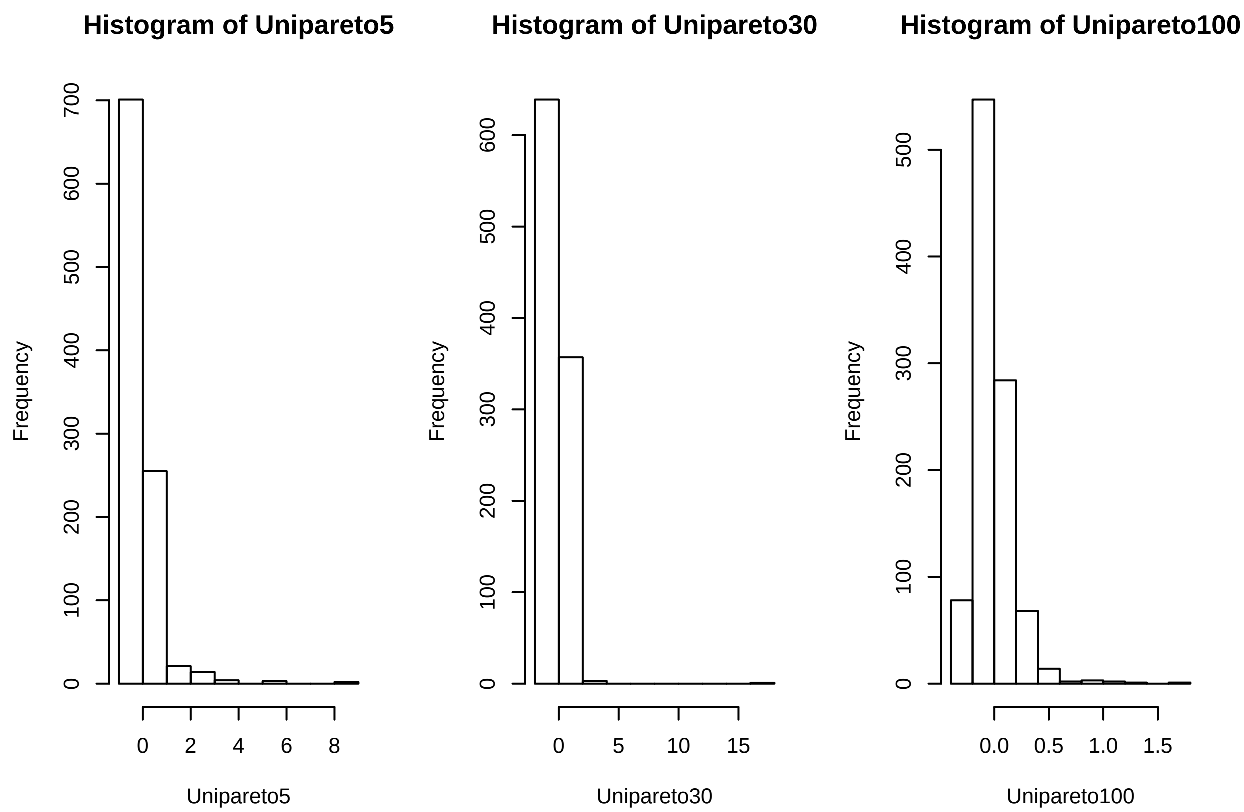
```
par(mfrow=c(1,3))
pareto5<-c(rpareto(5000,m=1, s=2))
pareto30<-c(rpareto(30000, m=1, s=2))
pareto100<-c(rpareto(100000,m=1, s=2))

Pareto5<-matrix(pareto5,1000,5)
Pareto30<-matrix(pareto30,1000,30)
```

```r
Pareto100<-matrix(pareto100,1000,100)

moyennepareto5= rep(1,1000)
for(i in 1:1000){
  moyennepareto5[i] = mean(Pareto5[i,])
}
hist(moyennepareto5)

variancepareto5=rep(1,1000)
for(i in 1:1000){
  variancepareto5[i]=var(Pareto5[i,])
}



moyennepareto30= rep(1,1000)
for(i in 1:1000){
  moyennepareto30[i] = mean(Pareto30[i,])
}
hist(moyennepareto30)

variancepareto30=rep(1,1000)
for(i in 1:1000){
  variancepareto30[i]=var(Pareto30[i,])
}

moyennepareto100= rep(1,1000)
for(i in 1:1000){
  moyennepareto100[i] = mean(Pareto100[i,])
}
hist(moyennepareto100)
```




```r
variancepareto100=rep(1,1000)
for(i in 1:1000){
  variancepareto100[i]=var(Pareto100[i,])
}
```

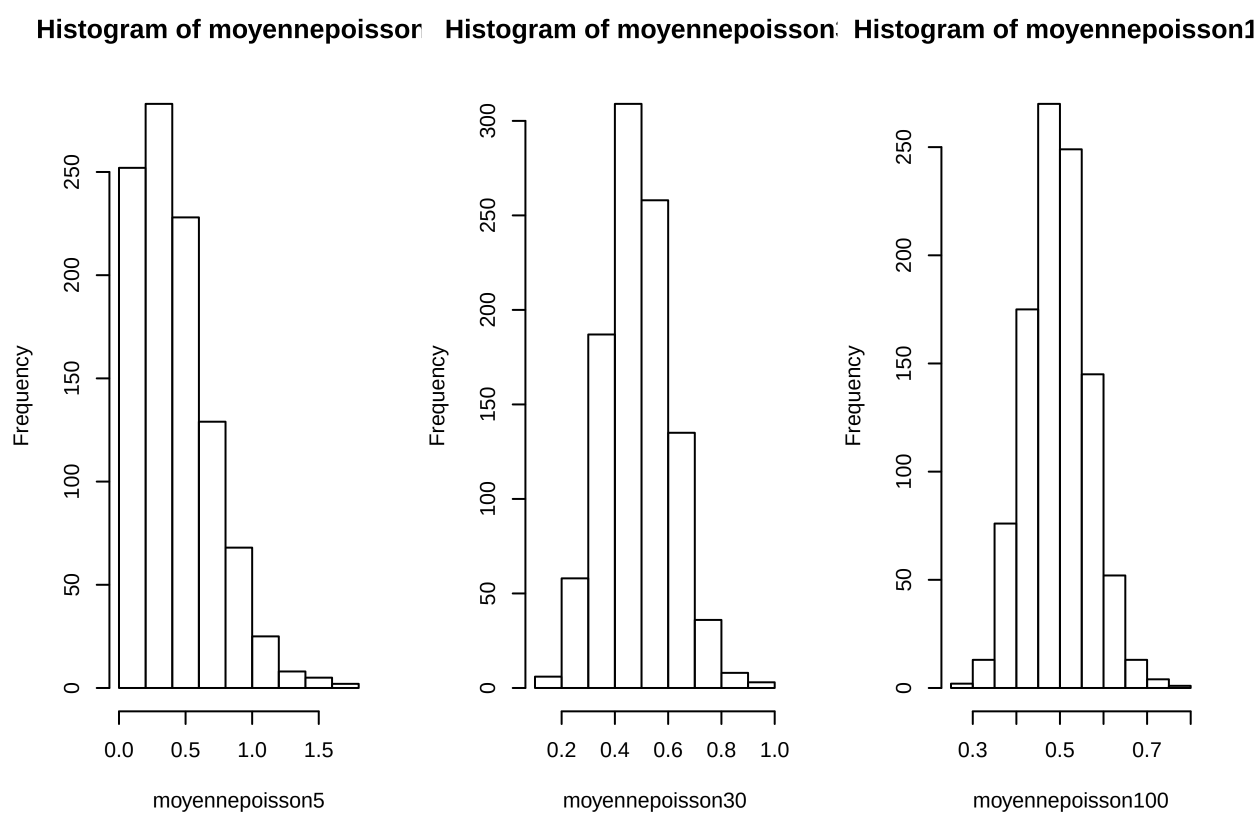
On choisit an=$\mathbb{E}[X]$, bn=$\sqrt{\mathbb{V}(X)}$ et on remarque que les $Un, i \sim \mathcal{N}(0,1)$. On centre et on réduit la loi.



Loi de Poisson

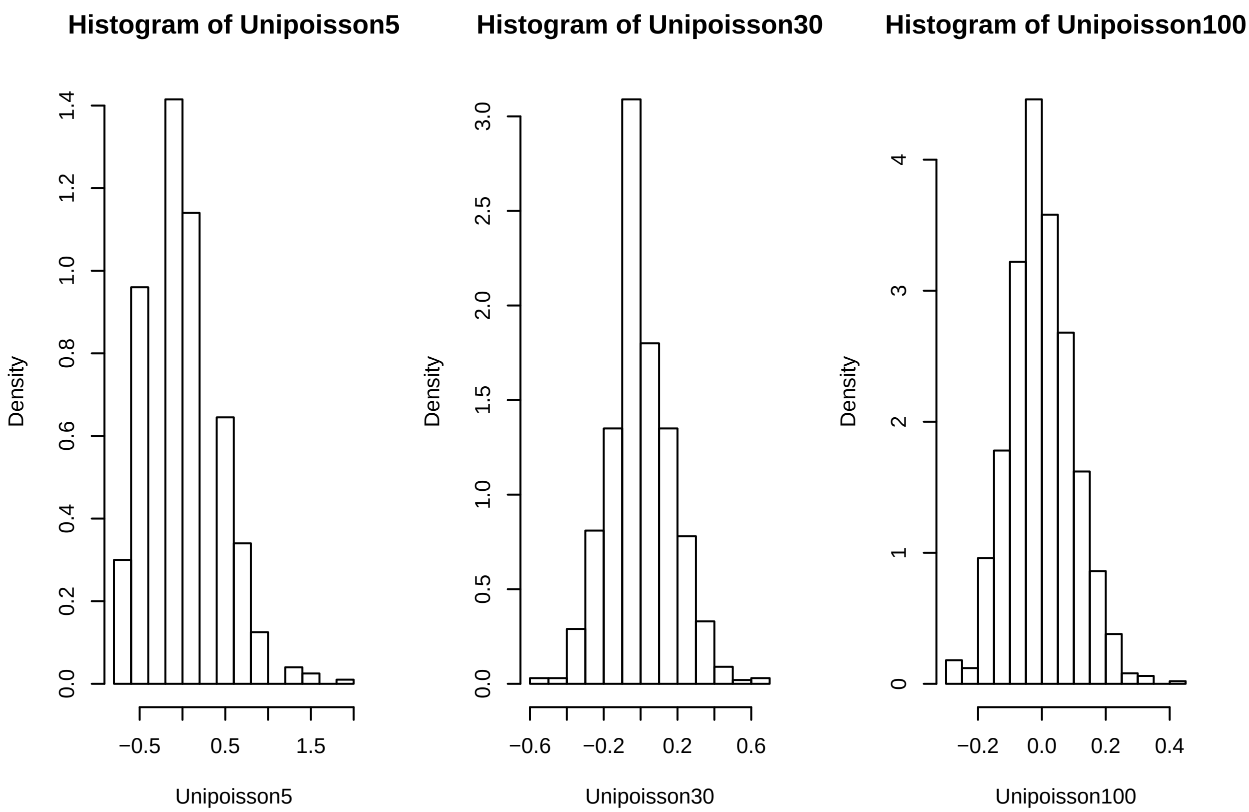
```
poisson5<-c(rpois(5000,0.5))
poisson30<-c(rpois(30000, 0.5))
poisson100<-c(rpois(100000,0.5))

Poisson5<-matrix(poisson5,1000,5)
Poisson30<-matrix(poisson30,1000,30)
Poisson100<-matrix(poisson100,1000,100)

par(mfrow=c(1,3))

moyennepoisson5= rep(1,1000)
for(i in 1:1000){
  moyennepoisson5[i] = mean(Poisson5[i,])
}
hist(moyennepoisson5)

variancepoisson5=rep(1,1000)
for(i in 1:1000){
  variancepoisson5[i]=var(Poisson5[i,])
}

moyennepoisson30= rep(1,1000)
```

```r
for(i in 1:1000){
  moyennepoisson30[i] = mean(Poisson30[i,])
}
hist(moyennepoisson30)

variancepoisson30=rep(1,1000)
for(i in 1:1000){
  variancepoisson30[i]=var(Poisson30[i,])
}

moyennepoisson100= rep(1,1000)
for(i in 1:1000){
  moyennepoisson100[i] = mean(Poisson100[i,])
}
hist(moyennepoisson100)
```




```r
variancepoisson100=rep(1,1000)
for(i in 1:1000){
  variancepoisson100[i]=var(Poisson100[i,])
}
```

On choisit an=$\mathbb{E}[X]$, bn=$\sqrt{\mathbb{V}(X)}$ et on remarque que les $Un, i \sim \mathcal{N}(0,1)$. On centre et on réduit la loi.



On peut donc approximer facilement une loi par une loi Normale centrée réduite. Plus la taille de l'échantillon est élevé (ie plus n est grand) et plus l'approximation sera proche de la réalité. N améliore la qualité de l'approximation

Moyenne et phénomène de concentration

Si $X \sim \mathcal{P}(\lambda)$, alors $\mathbb{E}(X) = \lambda$ et $\mathbb{V}(X) = \lambda$. Alors, $P(|X - \mathbb{E}(X)| > \epsilon)) \leq \frac{\lambda}{\epsilon^2}$

Si $X \sim \mathcal{N}(\mu, \sigma^2)$, alors $\mathbb{E}(X) = \mu$ et $\mathbb{V}(X) = \sigma^2$. Alors, $P(|X - \mathbb{E}(\mathbb{X})| \geq \epsilon) \leq (\frac{\sigma}{\epsilon})^2$

On pose $Z = 1_{\{|X-\mu| \leq \delta\}}$ .On a ainsi $P(|X - \mu| \geq \delta) = E[Z]$

Loi Normale

```
normalmontecarlo<-c(rnorm(1e5,mean=0, sd=1))

meansNMC<-0
delta<- 0.5

normaleZN=rep(1:1e5)
for (i in 1:1e5){
if (abs(normalmontecarlo[i]-meansNMC)>=delta){
  normaleZN[i]<-1
}
```

```r
else normaleZN[i]<-0}

meansnormaleZN<- mean(normaleZN)
print(paste("Espérance Loi normale:", meansnormaleZN, sep=""))
```

```
## [1] "Espérance Loi normale:0.61621"
```

Loi de Poisson

```r
poissonmontecarlo<-c(rpois(1e5,lambda=1))

meansPMC<-1
delta<- 0.5

poissonZN=rep(1:1e5)
for (i in 1:1e5){
if (abs(poissonmontecarlo[i]-meansPMC)>=delta){
  poissonZN[i]<-1
}
else poissonZN[i]<-0}

meanspoissonZN<-mean(poissonZN)
print(paste("Espérance Loi de Poisson de paramètre lambda=1:", meanspoissonZN, sep=""))
```

```
## [1] "Espérance Loi de Poisson de paramètre lambda=1:0.63085"
```

Loi de Pareto

```r
paretomontecarlo<-c(rpareto(1e5,m=1, s=2))
meansParetoMC<-2*1/(2-1)
delta<- 0.5

paretoZN=rep(1:1e5)
for (i in 1:1e5){
if (abs(paretomontecarlo[i]-meansParetoMC)>=delta){
  paretoZN[i]<-1
}
else paretoZN[i]<-0}

meansparetoZN<-mean(paretoZN)
print(paste("Espérance Loi de Pareto de paramètre m=1 et s=2:", meansparetoZN, sep=""))
```

```
## [1] "Espérance Loi de Pareto de paramètre m=1 et s=2:0.92185"
```

Bienaymé Tchebitchev

```r
for (delta in c(0.01,0.5,1)){
  for (sigma in c(0.2,0.4,2)){
    normalBT<-c(rnorm(1e5,mean=1, sigma))
    meansBT=1
    varNMC<-sigma**2

    normaleZN=rep(1:1e5)
    for (i in 1:1e5){
      if (abs(normalBT[i]-meansBT)>=delta){
        normaleZN[i]<-1
        }
      else normaleZN[i]<-0}
    Borne<-varNMC/(delta**2)
    meansnormaleBT<- mean(normaleZN)
    print(paste("Espérance Loi normale pour delta = ", delta, ", sigma = ",sigma, " vaut ", meansnormale
  }
}
```

```
## [1] "Espérance Loi normale pour delta = 0.01, sigma = 0.2 vaut 0.95981, borne =400"
## [1] "Espérance Loi normale pour delta = 0.01, sigma = 0.4 vaut 0.9804, borne =1600"
## [1] "Espérance Loi normale pour delta = 0.01, sigma = 2 vaut 0.99597, borne =40000"
## [1] "Espérance Loi normale pour delta = 0.5, sigma = 0.2 vaut 0.01338, borne =0.16"
## [1] "Espérance Loi normale pour delta = 0.5, sigma = 0.4 vaut 0.20992, borne =0.64"
## [1] "Espérance Loi normale pour delta = 0.5, sigma = 2 vaut 0.80188, borne =16"
## [1] "Espérance Loi normale pour delta = 1, sigma = 0.2 vaut 0, borne =0.04"
## [1] "Espérance Loi normale pour delta = 1, sigma = 0.4 vaut 0.01252, borne =0.16"
## [1] "Espérance Loi normale pour delta = 1, sigma = 2 vaut 0.61616, borne =4"
```

```r
for (delta in c(1,5,10)){
  for(lambda in c(0.5,5,10)){
  poissonBT<-c(rpois(1e5,lambda))
  meansPBT<-lambda

  varPMC<-lambda
  poissonZN=rep(1:1e5)
  for (i in 1:1e5){
    if (abs(poissonBT[i]-meansPBT)>=delta){
         poissonZN[i]<-1
      }
    else poissonZN[i]<-0}

  Borne2<-lambda/(delta**2)
    meanspoissonZN<-mean(poissonZN)
   print(paste("Espérance Loi de Poisson de paramètre lambda=",lambda, ", delta= ", delta, " vaut ", mea

  }}
```

```
## [1] "Espérance Loi de Poisson de paramètre lambda=0.5, delta= 1 vaut 0.08967, borne= 0.5"
## [1] "Espérance Loi de Poisson de paramètre lambda=5, delta= 1 vaut 0.8234, borne= 5"
## [1] "Espérance Loi de Poisson de paramètre lambda=10, delta= 1 vaut 0.87362, borne= 10"
## [1] "Espérance Loi de Poisson de paramètre lambda=0.5, delta= 5 vaut 1e-05, borne= 0.02"
## [1] "Espérance Loi de Poisson de paramètre lambda=5, delta= 5 vaut 0.03808, borne= 0.2"
```

```
## [1] "Espérance Loi de Poisson de paramètre lambda=10, delta= 5 vaut 0.15056, borne= 0.4"
## [1] "Espérance Loi de Poisson de paramètre lambda=0.5, delta= 10 vaut 0, borne= 0.005"
## [1] "Espérance Loi de Poisson de paramètre lambda=5, delta= 10 vaut 0.00029, borne= 0.05"
## [1] "Espérance Loi de Poisson de paramètre lambda=10, delta= 10 vaut 0.00312, borne= 0.1"
```

```r
for(delta in c(1,5,10)){
  for (alpha in c(3,4,6)){
    paretoBT<-c(rpareto(1e5,m=1, alpha))
    varPBT<-alpha**2/((alpha-1)**2)*alpha/(alpha-2)
    meansPBT<-alpha/(alpha-1)
    paretoZN=rep(1:1e5)
    for (i in 1:1e5){
      if (abs(paretomontecarlo[i]-meansPBT)>=delta){
        paretoZN[i]<-1
      }
      else paretoZN[i]<-0}
    Borne3<-varPBT/(delta**2)
      meansparetoZN<-mean(paretoZN)
   print(paste("Espérance Loi de Pareto de paramètre a=1, alpha=", alpha, ", delta= ", delta, " vaut ",


  }
}
```

```
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=3, delta= 1 vaut 0.63757, borne = 6.75"
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=4, delta= 1 vaut 0.52712, borne = 3.55555555555
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=6, delta= 1 vaut 0.40323, borne = 2.16"
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=3, delta= 5 vaut 0.01756, borne = 0.27"
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=4, delta= 5 vaut 0.01831, borne = 0.14222222222
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=6, delta= 5 vaut 0.01902, borne = 0.0864"
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=3, delta= 10 vaut 0.0061, borne = 0.0675"
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=4, delta= 10 vaut 0.00625, borne = 0.0355555555
## [1] "Espérance Loi de Pareto de paramètre a=1, alpha=6, delta= 10 vaut 0.00633, borne = 0.0216"
```

Les bornes de Bienaymé Tchebitchev sont donc bien respectées.

Chernoff

Si $X \sim \mathcal{N}(0, \sigma^2)$, alors $\mathcal{P}(X > \epsilon) \leq exp(-\frac{\epsilon^2}{2*\sigma^2})$

```r
sigma=2
ech_normale=rnorm(n=20,mean=1,sigma)
epsilon=0.5

echexpborne=rep(1:20)
for (i in 1:20){
  if (ech_normale[i]>epsilon) echexpborne[i]=1
  else echexpborne[i]=0
}

borneChernoff20=exp(-epsilon**2/(2*2**2))
print(paste("espérance loi normale (sigma=2) pour n=20 =", mean(echexpborne), "la borne est ", borneChe
```

```
## [1] "espérance loi normale (sigma=2) pour n=20 = 0.7 la borne est  0.969233234476344"
```

```r
sigma=2
ech_normale=rnorm(n=100,mean=1,sigma)
epsilon=0.5

echexpborne=rep(1:100)
for (i in 1:100){
  if (ech_normale[i]>epsilon) echexpborne[i]=1
  else echexpborne[i]=0
}

borneChernoff100=exp(-epsilon**2/(2*2**2))
print(paste("espérance loi normale (sigma=2 pour n=100 =", mean(echexpborne), "la borne est ", borneChe
```

```
## [1] "espérance loi normale (sigma=2 pour n=100 = 0.58 la borne est  0.969233234476344"
```

```r
ech_normale=rnorm(n=1000,mean=1,sigma)
epsilon=0.5

echexpborne=rep(1:1000)
for (i in 1:1000){
  if (ech_normale[i]>epsilon) echexpborne[i]=1
  else echexpborne[i]=0
}

borneChernoff1000=exp(-epsilon**2/(2*2**2))
print(paste("espérance loi normale (sigma=2) pour n=1000 =", mean(echexpborne), "la borne est ", borneC
```

```
## [1] "espérance loi normale (sigma=2) pour n=1000 = 0.594 la borne est  0.969233234476344"
```

Si $X \sim \mathcal{P}(\lambda)$, alors $\mathcal{P}(X > \epsilon) \leq exp(-\frac{\epsilon^2}{2*\lambda^2})$

```r
lambda=0.5
ech_poisson=rpois(n=20,lambda)
epsilon=0.3

echpoissonborne=rep(1:20)
for (i in 1:20){
  if (ech_poisson[i]>epsilon) echpoissonborne[i]=1
  else echpoissonborne[i]=0
}

borneChernoff20=exp(-epsilon**2/(2*lambda**2))
print(paste("espérance loi poisson (lambda=0.5) pour n=20 =", mean(echpoissonborne), "la borne est ", b
```

```
## [1] "espérance loi poisson (lambda=0.5) pour n=20 = 0.3 la borne est  0.835270211411272"
```

```r
ech_poisson=rpois(n=100,lambda)

echpoissonborne=rep(1:100)
```

```
for (i in 1:100){
  if (ech_poisson[i]>epsilon) echpoissonborne[i]=1
  else echpoissonborne[i]=0
}

borneChernoff100=exp(-epsilon**2/(2*lambda**2))
print(paste("esperance loi poisson (lambda=0.5) pour n=100 =", mean(echpoissonborne), "la borne est ", b
```

```
## [1] "esperance loi poisson (lambda=0.5) pour n=100 = 0.36 la borne est  0.835270211411272"
```

```
ech_poisson=rpois(n=1000,lambda)

echpoissonborne=rep(1:1000)
for (i in 1:1000){
  if (ech_poisson[i]>epsilon) echpoissonborne[i]=1
  else echpoissonborne[i]=0
}

borneChernoff1000=exp(-epsilon**2/(2*lambda**2))
print(paste("espérance loi poisson (lambda=0.5) pour n=1000 =", mean(echpoissonborne), "la borne est ",
```

```
## [1] "espérance loi poisson (lambda=0.5) pour n=1000 = 0.376 la borne est  0.835270211411272"
```

Loi de Cauchy

```
for (n in c(20,100,1000,10000,1e6)){
  cauchy<-rcauchy(n, location=0, scale=1)
  mean<-mean(cauchy)
  print(paste("pour n= ", n, " la moyenne empirique est ", mean, sep=""))
}
```

```
## [1] "pour n= 20 la moyenne empirique est 2.67611743243391"
## [1] "pour n= 100 la moyenne empirique est 0.967826031009799"
## [1] "pour n= 1000 la moyenne empirique est 2.11222952544155"
## [1] "pour n= 10000 la moyenne empirique est -0.0811572344572787"
## [1] "pour n= 1e+06 la moyenne empirique est -0.0665233795260554"
```

On remarque que la moyenne empirique diverge grossièrement. Plus n augmente et plus elle diverge. L'espérance ne converge donc pas.

Plus il y a de valeurs et plus on risque de s'éloigné de la médiane donc la moyenne diverge.

Estimateur

```
for (theta in c(-3,-2,-1,0,1,2,3)){
for (n in c(20,100,1000)){
  cauchy<-rcauchy(n, location=theta, scale=1)
  rangé<-sort(cauchy)
  print(paste("pour n= ", n, " theta= ",theta, " la médiane est ", (rangé[n/2]+rangé[n/2+1])/2, sep=""))
```

```
## [1] "pour n= 20 theta= -3 la médiane est -3.36874678846855"
## [1] "pour n= 100 theta= -3 la médiane est -3.07246690962389"
## [1] "pour n= 1000 theta= -3 la médiane est -2.93086785634714"
## [1] "pour n= 20 theta= -2 la médiane est -2.12851281112521"
## [1] "pour n= 100 theta= -2 la médiane est -2.2361586515686"
## [1] "pour n= 1000 theta= -2 la médiane est -2.00043317308806"
## [1] "pour n= 20 theta= -1 la médiane est -1.21554151017249"
## [1] "pour n= 100 theta= -1 la médiane est -1.11785384468747"
## [1] "pour n= 1000 theta= -1 la médiane est -0.889845823273945"
## [1] "pour n= 20 theta= 0 la médiane est 0.248648597534507"
## [1] "pour n= 100 theta= 0 la médiane est 0.0567788982970068"
## [1] "pour n= 1000 theta= 0 la médiane est -0.0375767621747419"
## [1] "pour n= 20 theta= 1 la médiane est 1.6555176220884"
## [1] "pour n= 100 theta= 1 la médiane est 0.977146599176847"
## [1] "pour n= 1000 theta= 1 la médiane est 0.995494651030649"
## [1] "pour n= 20 theta= 2 la médiane est 2.04974552042804"
## [1] "pour n= 100 theta= 2 la médiane est 1.97844038775041"
## [1] "pour n= 1000 theta= 2 la médiane est 1.9548348206361"
## [1] "pour n= 20 theta= 3 la médiane est 2.80182835890733"
## [1] "pour n= 100 theta= 3 la médiane est 2.73595565235928"
## [1] "pour n= 1000 theta= 3 la médiane est 3.08015326097838"
```

Le meilleur estimateur ici est Theta=0. En effet, les médianes sont proches des espérances calculées précédemment. Cela est logique car ces espérance ont été calculés pour theta=0.

Conclusion

Dans ce TP, nous avons tous d'abord travailler sur le théorème central limite. En réalisant différents échantillons de différentes lois, nous avons calculer leur moyenne empirique puis nous avons réaliser une renormalisation adéquate qui nous permet d'obtenir une loi normale centrée réduite. Il sufffit de prendre an=$\mathbb{E}(X)$ et bn=$\sqrt{\mathbb{V}(X)}$. La valeur de N permet d'améliorer la qualité de l'observation (plus N est grand et mieux est l'observation). Ensuite, nous nous sommes intéressés à différentes inégalités (Bienaymé-Tchebitchev,Chernoff…). Nous avons simuler plusieurs échantillons différents puis nous avons regarder les bornes des inégalités. On remarque que certains bornes sont absurdes (>1). Cela montre que certains majorations sont bien trop grossières. Enfin, nous avons déduis un estimateur de théta d'une loi de cauchy en observant la médiane. Le meilleur estimateur est celui pour lequel la médiane se rapproche le plus du thétha de l'observation.