**Statistique Numérique et Analyse de Données**
**Ecole des Ponts ParisTech, 2ème année**

# TP 3 : STATISTIQUE PARAMÉTRIQUE

La séance de TP se fait sous environnement Windows, sauf si vous avez une nette préférence pour Linux.

Pour commencer la séance

1. Créer un répertoire `TP3_Stat` sur le bureau.
2. Lancer un navigateur, par exemple Mozilla, et aller sur la page web suivante :
   `http://certis.enpc.fr/~dalalyan/Download/TP_ENPC_3.pdf`
3. Lancer ensuite `R` et modifier le répertoire de travail en allant dans
   `Fichier -> Changer le Répertoire Courant`
   et en choisissant le répertoire `Bureau/TP3_Stat` que vous avez créé.
4. Ouvrir une fenêtre d'éditeur `Fichier -> Nouveau Script`.
5. Sauvez le fichier dans le répertoire courant sous le nom `TP3.R`.
   `Fichier -> Sauver sous`
6. Pour les différentes questions, vous pouvez utiliser un «copier-coller» à partir de ce document. Il est fortement récommandé de saisir toutes les commandes dans la fenêtre de l'éditeur que vous avez ouverte. Pour exécuter les commandes saisies, il suffit de les selectionner avec la souris et d'appuyer simultanément sur les touches `Ctrl` et `R`.
7. Pour inclure des commentaires dans le programme, ce qui est fortement récommandé, vous devez utiliser le caractère `#`. Tout ce qui suit ce caractère `#` sera négligé lors de l'exécution.
8. Penser à sauvegarder régulièrement le contenu du fichier `TP3.R` en appuyant sur les touches `Ctrl` et `S`.

## 1. Comparaison des estimateurs du paramètre de position

Soit $X_1, \dots, X_n$ des variables aléatoires i.i.d. à densité. On suppose que cette densité, notée $p$, est symétrique par rapport à une valeur réelle $\theta$, c'est-à-dire que

$$p(\theta + x) = p(\theta - x), \qquad \forall x \in \mathbf{R}.$$

Le but de cet exercice est d'étudier les propriétés de trois estimateurs de $\theta$ : la moyenne, la médiane et le « mid-range », définis par

$$\hat{\theta}_1 = \frac{X_1 + \ldots + X_n}{n}, \qquad \hat{\theta}_2 = \mathrm{Med}_n, \qquad \hat{\theta}_3 = \frac{\min_i X_i + \max_i X_i}{2}.$$

1. On commence par poser $n = 1000$ et par générer $n$ réalisations indépendantes d'une v.a. de loi gaussienne $\mathcal{N}(5,4)$ :

```
n=1000;
X=randn(n,mean=5,sd=2);
```

2. On vérifie que la répartition des éléments de `X` est proche de la loi gaussienne :

```
par(bg="cornsilk",lwd=2,col="darkblue")
hist(X,breaks=20,freq=F,col="cyan")
curve(dnorm(x,mean=5,sd=2),add=T)
```

**Question :** Quelle est la valeur de $\theta$ dans ce cas ?

3. On veut savoir quelle est le comportement des estimateurs $\hat{\theta}_1$, $\hat{\theta}_2$ et $\hat{\theta}_3$ lorsque $n \to \infty$.

   (a) Pour cela, on écrit la fonction suivante :

```
location_estimator=function(U,theta)
{
        n=length(U)
        theta1=cumsum(U)/(1:n)
        theta2=1:n
        for (i in 1:n)
            theta2[i]=median(U[1:i])
        end
        par(mfrow=c(1,2),bg="cornsilk",lwd=2,col="darkblue")
        plot(1:n,theta1,type="l",main="moyenne")
        abline(h=theta,col="darkred")
        plot(1:n,theta2,type="l",main="mediane")
        abline(h=theta,col="darkred")
        return(matrix(c(theta1,theta2),n,2))
}
```

   **Questions :**
   1. Si `U` est un vecteur dont les coordonnées sont $(u_1, \ldots, u_n)$, que représente le vecteur `theta1` ? le vecteur `theta2` ?
   2. A quoi sert l'option `type="l"` dans la commande `plot` ?

   (b) On appelle cette fonction à l'aide des commandes

```
X=rnorm(n,mean=5,sd=2)
est=location_estimator(X,5)
```

   **Question :** Que remarque-t-on ? A quoi correspond la matrice `est` ?

   (c) Compléter la fonction `location_estimator` pour qu'elle renvoie également la valeur de $\hat{\theta}_3$. (On pourra utiliser les commandes `cummax` et `cummin`.) A votre avis, l'estimateur $\hat{\theta}_3$ est-il convergent ?

4. On veut maintenant comparer les 3 estimateurs.

   (a) Pour cela, on écrit une fonction qui génère $N$ échantillons dont chacun est de taille $n$ et est distribué suivant la loi normale $\mathcal{N}(5,4)$.
   Pour chaque échantillon, on calcule les 3 estimateurs. Cela nous donne 3 séries numériques de $N$ valeurs $\hat{\theta}_1^j$, $j = 1, \ldots, N$, $\hat{\theta}_2^j$, $j = 1, \ldots, N$ et $\hat{\theta}_3^j$, $j = 1, \ldots, N$.
   On trace ensuite les boxplots de ces 3 séries numériques.

```
compare_estimators=function(N,n)
{
        X=matrix(rnorm(N*n,mean=5,sd=2),N,n)

        theta1=apply(X,1,mean)
        theta2=apply(X,1,median)
        theta3=(apply(X,1,min)+apply(X,1,max))/2

        par(bg="cornsilk",lwd=2,col="darkblue")
        boxplot(theta1,theta2,theta3,col="cyan")
}
```

**Question :** Que fait la commande `apply` ?

(b) On appelle cette fonction :

```
compare_estimators(200,1000)
```

Au vue de ce résultat, lequel des 3 estimateurs préféreriez-vous ?

(c) On modifie la fonction `compare_estimators` en remplaçant la loi normale par la loi uniforme sur $[0, 10]$ : `runif(N*n,0,10)`. Lequel des 3 estimateurs préférez-vous dans ce cas-là ?

(d) On modifie encore la fonction `compare_estimators` en remplaçant la loi uniforme par la loi de Cauchy : `rcauchy(N*n,location=5,scale=2)`.

i. On observe d'abord que le troisième estimateur est incontestablement le moins bon des trois. (Le graphe correspondant doit être inclu dans le compte-rendu.)

ii. On supprime le calcul de $\hat{\theta}_3$ ainsi que l'affichage de son boxplot de la fonction `compare_estimators` et on relance la commande

```
compare_estimators(200,1000)
```

Le résultat obtenu est-il en faveur de $\hat{\theta}_1$ ou $\hat{\theta}_2$ ?

Résumer les réponses obtenues dans le tableau suivant :

| Loi | $\mathcal{N}(5,4)$ | $\mathcal{C}(5,2)$ | $\mathcal{U}([0,10])$ |
|---|---|---|---|
| Meilleur estimateur parmi $\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3$ | | | |

5. Si vous aviez de vraies données, de loi inconnue, réparties de façon symétrique par rapport à une valeur $\theta$, que feriez-vous pour estimer cette valeur ?

## 2. Intervalles de confiance (IC)

**IC pour la moyenne d'un échantillon gaussien :** on observe un $n$-échantillon, $x_1, \ldots, x_n$, distribué suivant la loi gaussien $\mathcal{N}(\mu, \sigma^2)$ où $\mu \in \mathbf{R}$ et $\sigma > 0$ sont des paramètres inconnus. Le but est de déterminer un IC de niveau 95% pour $\mu$ et étudier ses propriétés. On sait que $\mu$ et $\sigma^2$ sont bien estimées respectivement par la moyenne empirique $\bar{x} = \frac{1}{n}\sum_i x_i$ et par la variance empirique sans biais $s_n^2 = \frac{1}{n-1}\sum_i (x_i - \bar{x})^2$.

1. La théorie dit que la variable aléatoire $T_n = \frac{(\bar{X}-\mu)\sqrt{n}}{S_n}$ suit la loi de Student à $n-1$ degrés de liberté (cf. Proposition 4.1. du polycopié). On cherche à avoir une confirmation empirique de ce résultat. Pour cela, on génère $N$ échantillons de taille $n$ de loi $\mathcal{N}(\mu, \sigma^2)$, pour des valeurs de $\mu$ et de $\sigma^2$ données. Pour chacun de ces $N$ échantillons, on calcule la valeur de $T_n$. Cela nous donne une série numérique $t_n^1, \ldots, t_n^N$.

```
check_student=function(N,n,mu,sigma)
{
    X=matrix(rnorm(N*n,mean=mu,sd=sigma),N,n)
    t=sqrt(n)*(apply(X,1,mean)-mu)/apply(X,1,sd)
    return(t)
}
```

   Si cette série est vraiment distribuées suivant la loi de student $t_{n-1}$, alors son histogramme devrait être proche de la densité de la loi $t_{n-1}$. Pour vérifier cela, on exécute les commandes :

```
N=5000; n=30;
t=check_student(N,n,2,3)
par(bg="cornsilk",lwd=2,col="darkblue")
hist(t,breaks=30,freq=F,col="cyan")
curve(dt(x,n-1),add=T,lwd=2)
```

2. **Questions :**
   (a) Déterminer les quantiles d'ordre 2.5% et 97.5% de la série numérique $t_n^1, \ldots, t_n^N$. (On lira l'aide en ligne de la commande `quantile`.)
   (b) Faire varier les valeurs de $\mu$ et de $\sigma$. Cela influence t-il le résultat ?
   (c) Augmenter $N$ jusqu'à 40.000 (si les capacité de la machine le permettent) et refaire l'expérience. Quelles sont les valeurs obtenus pour les deux quantiles ?
   (d) Soit $a$ et $b$ les valeurs obtenues dans la question précédente. Si $T$ est une v.a. distribuée suivant la loi $t_{n-1}$, quelle est la probabilité $\mathbb{P}(T \in [a, b])$ ?
   (e) Vérifier par calcul que $T_n \in [a, b]$ équivaut à

$$\mu \in \left[\bar{X} - \frac{S_n b}{\sqrt{n}}, \bar{X} - \frac{S_n a}{\sqrt{n}}\right]. \tag{1}$$

   Quel est le pourcentage des échantillons (parmi les $N$ échantillons générés) pour lesquels (1) est satisfaite ?

3. **Application :** On mesure la force de compression d'un ciment en moulant de petits cylindres et en mesurant la pression $X$, mesurée en kg/cm$^2$, à partir de laquelle ils se cassent. Pour 10 cylindres utilisés on observe les pressions suivantes :

```
19.6 19.9 20.4 19.8 20.5 21.0 18.5 19.7 18.4 19.4
```

   et on suppose que la variable aléatoire $X$ obéit à une loi gaussienne. On veut déterminer un IC de niveau 95% pour la moyenne de $X$. Pour cela,

(a) Rentrer les données :
`X=c(19.6, 19.9, 20.4, 19.8, 20.5, 21.0, 18.5, 19.7, 18.4, 19.4)`.

(b) Déterminer la valeur de $n$ et les valeurs de $a$ et de $b$ correspondant à cette valeur de $n$.

(c) En déduire l'IC demandé par la formule (1).

## 3. Calcul de l'EMV à l'aide de la commande `mle`

Le but de cet exercice est de vous apprendre à vous servir de la commande `mle` afin de calculer l'estimateur du maximum de vraisemblance (EMV) dans un modèle paramétrique spécifié. On va illustrer l'utilisation de cette commande sur l'exemple de la loi gamma. Rappelons qu'une variable aléatoire $X$ suit une loi gamma de paramètre de forme $a > 0$ et de paramètre d'échelle $\sigma > 0$ si $X$ est à densité et sa densité est donnée par :

$$p_X(x, a, \sigma) = \frac{x^{a-1}}{\sigma^a \Gamma(a)} e^{-x/\sigma} \mathbb{1}_{]0,\infty[}(x), \tag{2}$$

où $\Gamma(a) = \int_0^\infty t^{a-1} e^{-t}\, dt$ désigne la fonction gamma.

Les 50 valeurs suivantes ont été générées par une loi gamma : `77.551 45.195 50.626 39.878 29.137 57.321 39.140 66.776 48.028 42.325 31.200 38.632 42.914 60.969 22.076 52.446 45.257 42.626 62.504 22.684 69.196 42.383 61.339 45.803 74.707 33.048 72.423 43.670 65.279 42.714 59.785 101.742 59.641 44.749 44.161 58.488 46.448 25.280 67.619 66.846 80.208 98.492 41.149 40.395 22.220 34.628 77.768 48.161 48.909 66.267`
On cherche

- à estimer les paramètres $a$ et $\sigma$ qui ont été utilisés pour générer ces valeurs,
- et à fournir un IC pour chacun de ces paramètres.

1. Pour cela, on commence par saisir les données dans une variable qu'on appelle $X$ :
```
X=c(77.551, 45.195, 50.626, 39.878, 29.137, 57.321, 39.140,  66.776,
    48.028, 42.325, 31.200, 38.632, 42.914, 60.969, 22.076,  52.446,
    45.257, 42.626, 62.504, 22.684, 69.196, 42.383, 61.339,  45.803,
    74.707, 33.048, 72.423, 43.670, 65.279, 42.714, 59.785, 101.742,
    59.641, 44.749, 44.161, 58.488, 46.448, 25.280, 67.619,  66.846,
    80.208, 98.492, 41.149, 40.395, 22.220, 34.628, 77.768,  48.161,
    48.909, 66.267)
```
2. On définit une fonction qui calcule la log-vraisemblance négative de la loi gamma :
```
ll=function(a=1, sigma=1)
{
  if(a > 0 && sigma > 0)
       -sum(dgamma(X, shape=a, scale=sigma, log=TRUE))
  else
       NA
}
```

(a) Que fait la commande `dgamma` appelée avec l'option `log=TRUE` ?

(b) Etant donnée un vecteur $X = (x_1, \ldots, x_n)$ et deux valeurs réelles $a$ et `sigma`, quelle est la valeur retournée par `ll(a,sigma)` ?

(c) Faut-il minimiser ou maximiser la fonction `ll` pour obtenir l'EMV ?

3. Afin d'obtenir l'EMV à l'aide de la commande `mle` (Maximum Likelihood Estimator), il faut d'abord charger le package `stats4` :

```
library(stats4)
fit = mle(ll)
summary(fit)
```

Quelles sont les estimations obtenues pour $a$ et $\sigma$ ?

4. La commande `mle` ne fait pas que calculer l'EMV, elle peut aussi être utilisée pour déterminer des intervalles de confiance pour les paramètres ainsi que pour calculer la matrice de covariance limite :

```
vcov(fit)
par(mfrow=c(2,1),bg="cornsilk",col="blue",lwd=2)
plot(profile(fit), absVal=FALSE)
confint(fit,level=0.95)
```

Quel est l'IC de niveau 95% pour le paramètre $a$ ? 85% pour le paramètre $\sigma$ ?

5. On peut même afficher la surface représentée par la log-vraisemblance négative. Cela peut se faire en utilisant les commandes `persp`, `image` et `contour` comme dans l'exemple suivant :

```
# Calcul des valeurs de la log-vraisemblance
  K=80
  x=(1:K)/4;  y=(1:K)/4;  z=c();
  for (i in 1:length(x))
    for (j in 1:length(y))
      z=c(z,ll(x[i],y[j]))
# Transformation des valeurs calculées
  z=matrix(z,length(x),length(y))
  z=log(0.001+((z-min(z))/(max(z)-min(z))))
# Le contenu des 7 lignes suivantes peut etre utilisé comme une
# boîte noire
  nrz <- nrow(z)
  ncz <- ncol(z)
  jet.colors <- colorRampPalette( c("blue", "green") )
  nbcol <- 100
  color <- jet.colors(nbcol)
  zfacet <- z[-1, -1] + z[-1, -ncz] + z[-nrz, -1] + z[-nrz, -ncz]
  facetcol <- cut(zfacet, nbcol)
# Visualisation des résultats
  par(bg="cornsilk",lwd=1,mfrow=c(1,2))
  image(x,y,z,col = cm.colors(50))
  contour(x,y,z,add=T,col="darkred")
  persp(x, y, z,ticktype="detailed",expand=0.5,col=color[facetcol],shade=0.4)
```

La surface affichée représente-t-elle la log-vraisemblance négative ou une transformation de celle-ci ? Donner la formule de la transformation le cas échéant et justifier son utilisation (on essaira d'exécuter les commandes ci-dessus sans utiliser la transformation).

6. Afin de vérifier que les commandes ci-dessus fournissent vraiment une estimation des paramètres $(a, \sigma)$ de la loi gamma, effectuer le test suivant.
   (a) Générer $n = 1000$ valeurs aléatoires de loi gamma de paramètres $a = 5$ et $\sigma = 3$.
   (b) Déterminer une estimation de $a$ et de $\sigma$.
   (c) Déterminer un IC de niveau 95% pour $a$ et pour $\sigma$.
   (d) Vérifier que les estimations obtenues sont proches des vraies valeurs.
7. Lors de la définition de la fonction `ll` dans la question 2, nous avons utilisé les valeurs initiales $a = 1$ et $\sigma = 1$. La nécessité de donner des valeurs initiales aux paramètres vient du fait que le calcul (approximatif) de l'EMV est fait par une méthode d'optimisation du type «descente de gradient». La log-vraisemblance négative de la loi gamma n'étant pas convexe, le point de minimum trouvé pourrait dépendre de la valeur initiale. Etudier la dépendance des valeurs initiales de l'EMV calculé ci-dessus en faisant varier les valeurs initiales et en observerant ce qui se passe avec l'EMV.
8. **Question pour le compte-rendu :** nous avons un échantillon de $n = 99$ valeurs réelles qui semblent être distribuées suivant une loi de Cauchy de paramètre de position $a$ et de paramètre d'échelle $s$ inconnus. Ces valeurs sont

```
 -7.54,  82.51,  14.27,   3.96,  189.98,   17.20,  -20.07,   52.66,   93.47,
-33.57,  13.13,  -1.26,  12.69,   53.33,    2.85,   -7.25,   13.30,   -5.67,
-38.99,  24.24,   4.17,  12.30,   21.59,   -6.70,    1.24,   13.91,   30.24,
  3.35,   6.45, -26.22,  72.65,   10.12,   -1.64,   21.49,  391.11,   26.53,
146.60,   2.11,   5.84,  14.25,    7.17,    4.96,   -9.55,    7.89,   -2.31,
 91.11,   8.39,   6.23,  25.45,    9.36,  102.44,   -7.28,  -40.02,   -8.86,
 14.11,   6.84, -11.15,  -6.67,  -84.82, -241.41,   -0.14,  -72.95,   21.09,
 53.47,  -3.80, -10.64,  19.71,   45.89, -124.30,   -2.02,   -1.67,    7.81,
 -9.76,   6.25,  16.68,   8.88,   32.14,    1.29,  -10.00,   -5.03,  -66.77,
 12.85,  15.32,  31.27,   6.59,    3.92,    8.61,   15.38,   -1.34,   14.11,
 10.53,   2.35, -94.19,  16.45,    2.97,   12.26,    4.15,   10.63,    5.47
```

   (a) Ecrire une fonction qui calcule l'EMV pour un échantillon distribué selon la loi de Cauchy $\mathcal{C}(a, s)$.
   (b) Donner une estimation des paramètres $a$ et $s$, ainsi que des IC de niveau 95% pour ces deux paramètres.
   (c) Les valeurs obtenues, dépendent-elles des valeurs initiales données à la log-vraisemblance ? Expliquer le résultat.
   (d) Afficher le surface de la log-vraisemblance (éventuellement transformée par une transformation croissante qui améliore le résultat visuel).