T.P. Variables aléatoires et inférence statistique (Labo 2)

201-9F6-ST : Statistiques appliquées à l'informatique

Dominique Goyette et Marc-André Désautels

2017-12-08

Instructions:

1. Le but de ce T. P. est de vous familiariser avec le langage `R`. Il vous faudra trouver et utiliser les commandes appropriées pour répondre aux questions. Vous devez vous aider de la documentation fournie dans le logiciel `RStudio` ou de la recherche `Google`.
2. Vous devez répondre aux questions directement dans ce document et vous assurez qu'il compile lorsque vous utilisez la commande `Knit`. Vous pouvez également compiler vos commandes au fur et à mesure dans ce document en appuyant sur la **flèche verte pointant vers la droite** en haut à droite de votre code `R`.

Installer `R` et `RStudio`

Vous pouvez télécharger `R` aux adresses suivantes:

- Pour Linux
- Pour (Mac) OS X
- Pour Windows

Une fois le logiciel `R` installé, vous pouvez télécharger et installer le logiciel `RStudio` à l'adresse suivante:

- Pour Linux, (Mac) OS X et Windows

Les lois de probabilités

Chaque distribution en `R` possède quatre fonctions qui lui sont associées. Premièrement, la fonction possède un *nom racine* (qui correspond au nom de la `loi`), par exemple le *nom racine* pour la distribution *binomiale* est `binom`. Cette racine est précédée par une de ces quatre lettre:

- `p` pour *probabilité*, qui représente la fonction de répartition
- `q` pour *quantile*, l'inverse de la fonction de répartition
- `d` pour *densité*, la fonction de densité de la distribution
- `r` pour *random* ou *simulation*, une variable aléatoire suivant la distribution spécifiée.

Pour la loi binomiale (*nom racine* `binom`) par exemple, ces fonctions sont `pbinom`, `qbinom`, `dbinom` et `rbinom`.

Nous avons donc:

Loi: `loi`	Densité	Fonction de répartition	Quantile	Simulation
Notations	$f(x)$ ou $P(X = x)$	$F(x)$	valeur liée à $F(x)$	$x_1, x_2, \ldots, x_n$
Commandes	`dloi`	`ploi`	`qloi`	`rloi`

Les noms de lois les plus célèbres sont : `norm` (pour la loi normale), `norm` (pour la loi binomiale), `unif` (pour la loi uniforme), `geom` (pour la loi géométrique), `pois` (pour la loi de Poisson), `t` (pour la loi de Student),

`chisq` (pour la loi du Chi-deux), `exp` (pour la loi exponentielle), `f` (pour la loi de Fisher)...

Commandes

Si la loi de X dépend d'un ou de plusieurs paramètres, disons `par1` et `par2`, alors la densité de X en x est donnée par la commande : `dloi(x, par1, par2)`

Quelques exemples sont décrits ci-dessous:

Loi	Binomiale	Géométrique	Poisson
Paramètres	$n \in \mathbb{N}$, $p \in]0,1[$	$p \in]0,1[$	$\lambda > 0$
$X \sim$	$B(n;p)$	$G(p)$	$Po(\lambda)$
Ch(X)	$\{0,1,\ldots,n\}$	$\mathbb{N}$	$\mathbb{N}$
$P(X=x)$	$C_x^n p^x q^{n-x}$	$p(1-p)^x$	$e^{-\lambda}\frac{\lambda^x}{x!}$
Commandes	`dbinom(x,n,p)`	`dgeom(x,p)`	`dpois(x,lambda)`

Loi	Uniforme	Exponentielle	Normale
Paramètres	$(a,b) \in \mathbb{R}^2$	$p \in]0,1[$	$\lambda > 0$
$X \sim$	$U([a,b])$	$E(\lambda)$	$N(\mu,\sigma^2)$
Ch(X)	$[a,b]$	$[0,\infty]$	$\mathbb{R}$
$P(X=x)$	$\frac{1}{b-a}$	$\lambda^{-\lambda x}$	$\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$
Commandes	`dunif(x,a,b)`	`dexp(x,lambda)`	`dnorm(x,mu,sigma)`

Exemples de calculs

Soit X une variable aléatoire telle que $X \sim B(8, 0.3)$.

1. Pour calculer $P(X = 4)$, nous devons utiliser la commande suivante:

```
dbinom(4,8,0.3)
```

```
## [1] 0.1361367
```

Ceci signifie que $P(X = 4) = 0.1361367$.

2. Pour calculer $P(X \leq 4)$, nous devons utiliser la commande suivante:

```
pbinom(4,8,0.3)
```

```
## [1] 0.9420324
```

Ceci signifie que $P(X \leq 4) = 0.9420324$.

3. Pour calculer $P(X > 4)$, nous pouvons utiliser une des commandes suivantes:

```
pbinom(4,8,0.3,lower.tail = FALSE)
```

```
## [1] 0.05796765
```

```
1-pbinom(4,8,0.3)
```

```
## [1] 0.05796765
```

Ceci signifie que $P(X > 4) = 0.0579676$.

4. Pour calculer $P(X \geq 4) = 1 - P(X \leq 3)$, nous pouvons utiliser la commande suivante:

```
1-pbinom(3,8,0.3)
```

```
## [1] 0.1941043
```

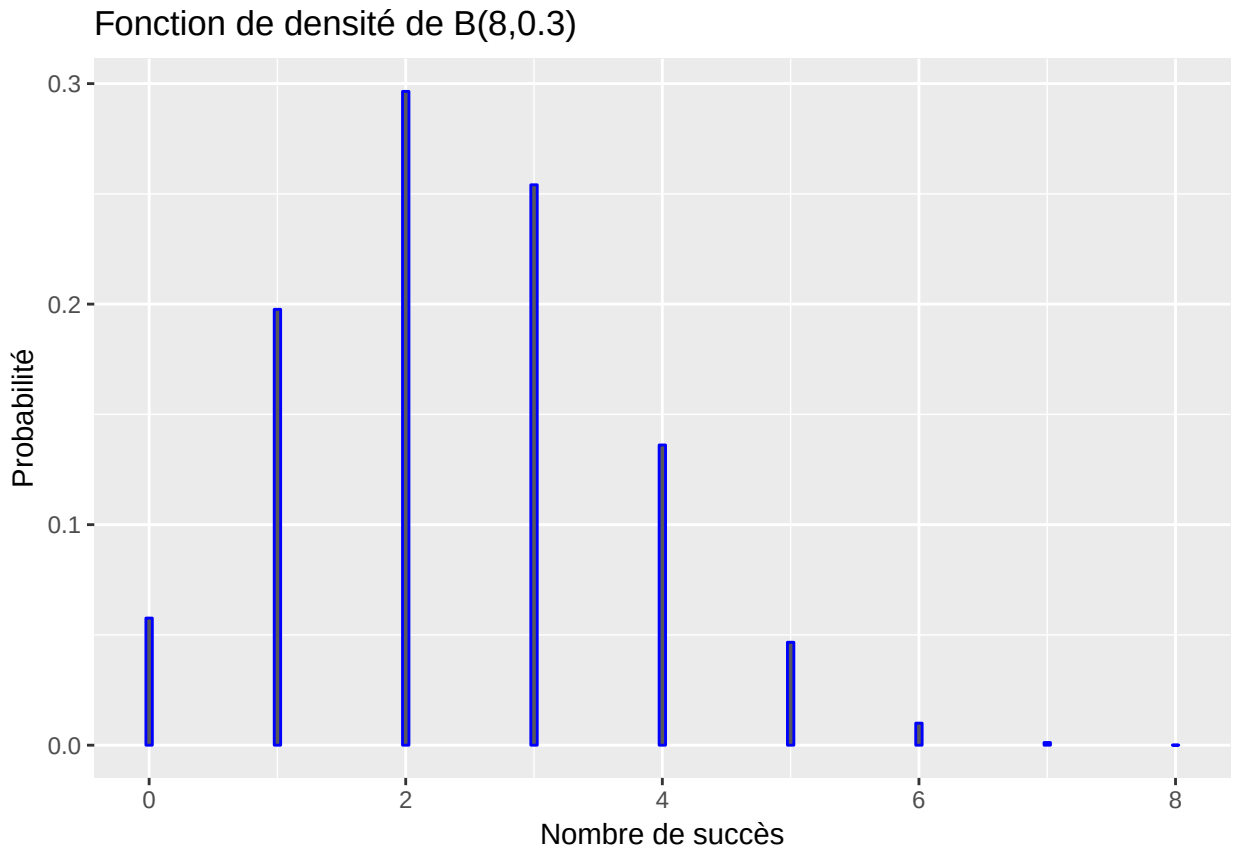
Ceci signifie que $P(X \geq 4) = 0.1941043$.

Représentation graphique

Les lois de probabilités discrètes

Nous pouvons représenter graphiquement la loi binomiale. Soit X $B(8, 0.3)$. Nous aurons:

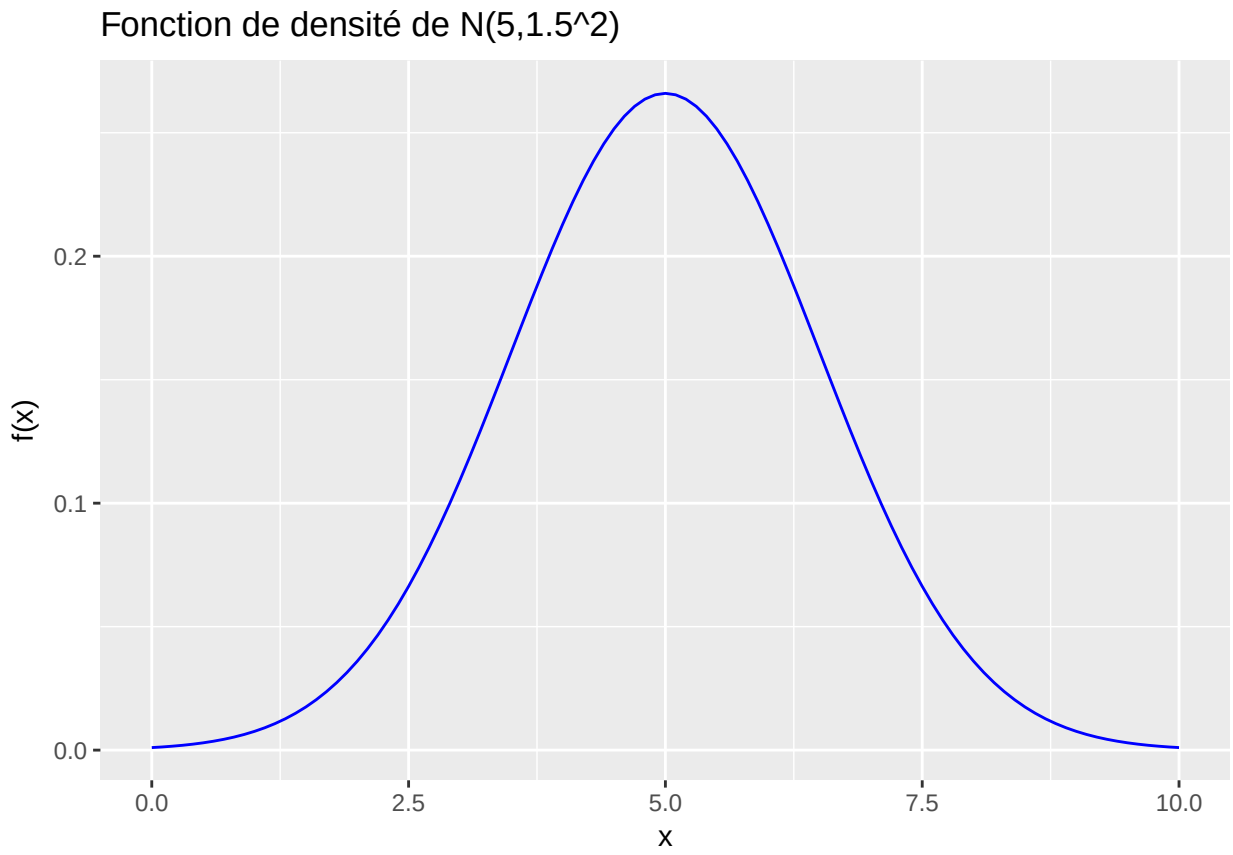
```
n <- 8
p <- 0.3
fbinom <- data.frame(x = 0:n, y = dbinom(0:n, n, p))
ggplot(fbinom, aes(x = x, y = y)) +
  geom_bar(width = 0.05, stat = "identity", colour = "blue") +
  labs(
    x = "Nombre de succès",
    y = "Probabilité",
    title = "Fonction de densité de B(8,0.3)"
  )
```



Les lois de probabilités continues

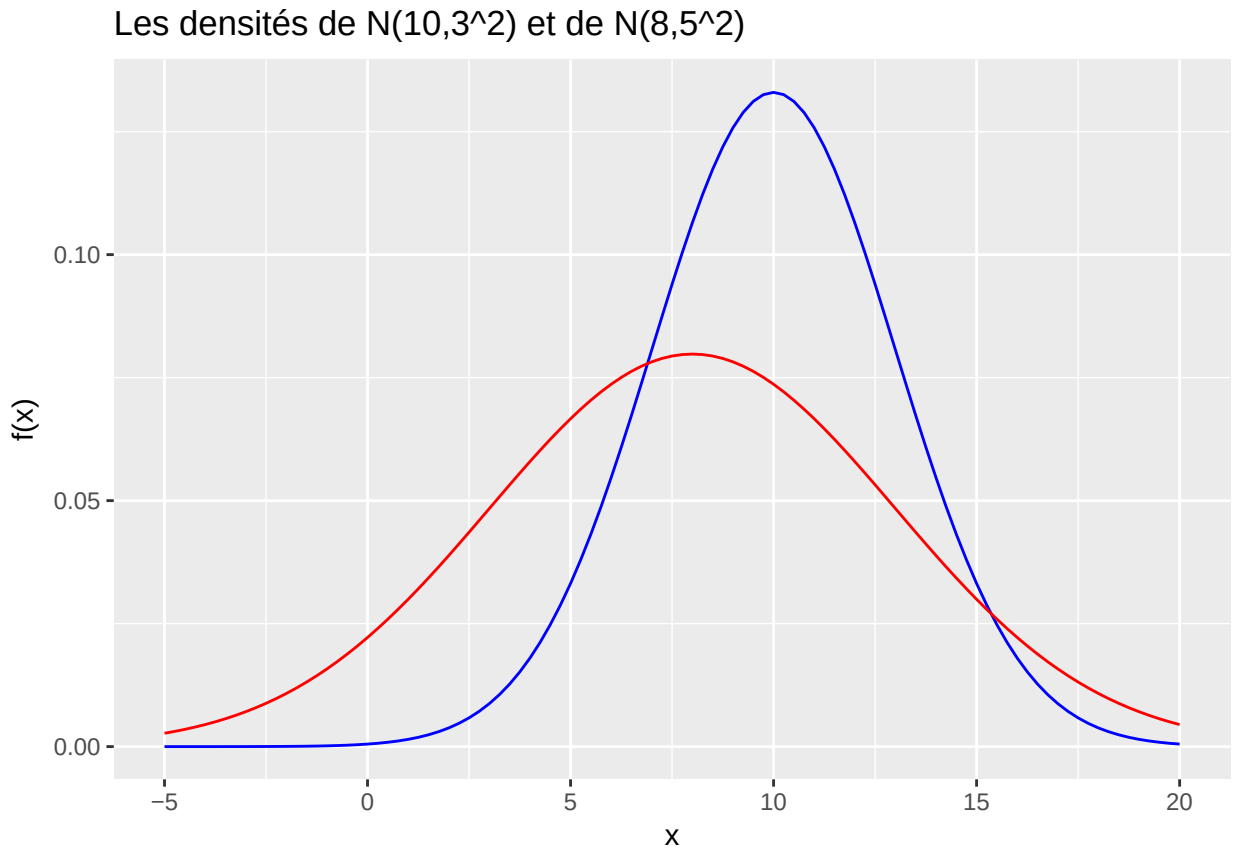
Nous pouvons représenter graphiquement la loi normale. Soit $X \sim N(5, 1.5^2)$. Nous aurons:

```
ggplot(data = data.frame(x = c(0, 10)), aes(x)) +
  stat_function(fun = dnorm, args = list(mean = 5, sd = 1.5), colour = "blue") +
  labs(
    x = "x",
    y = "f(x)",
    title = "Fonction de densité de N(5,1.5^2)"
  )
```



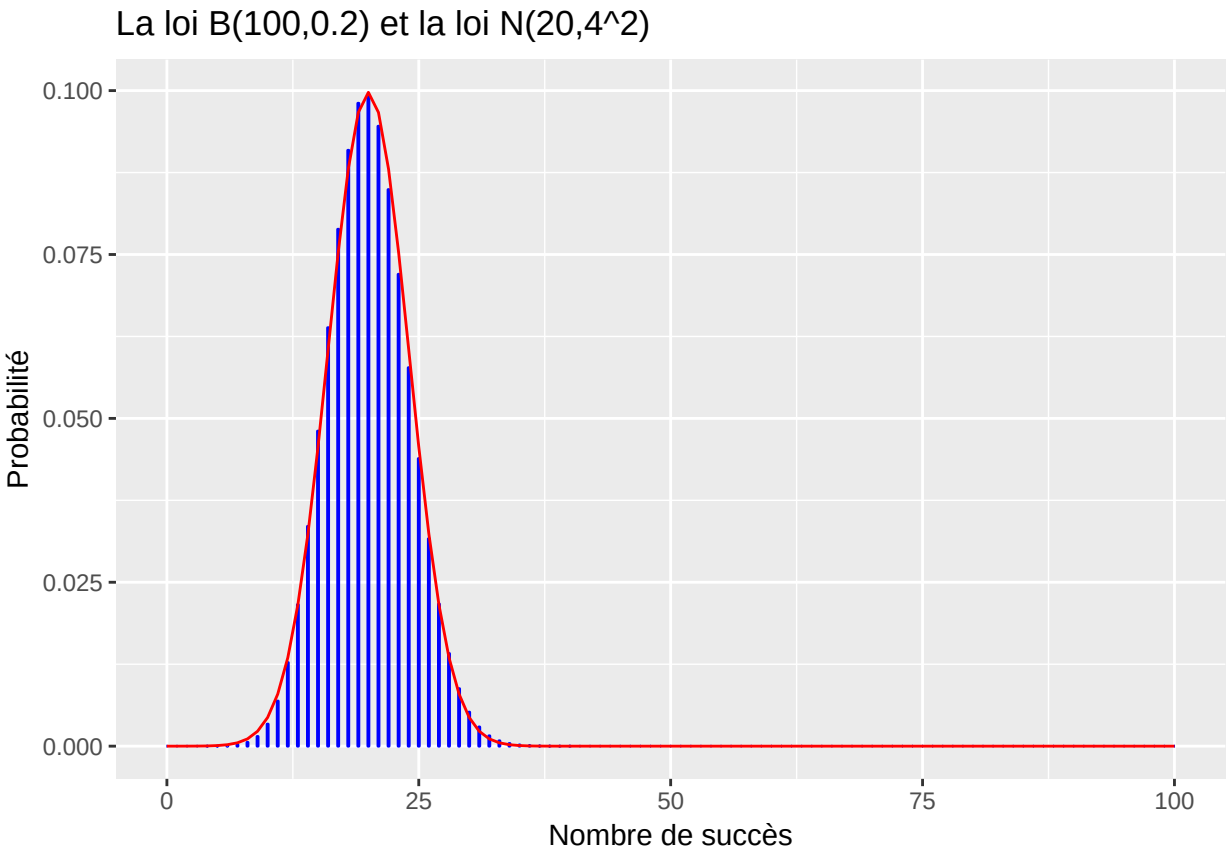
Nous pouvons également superposer plusieurs fonctions de densité. Par exemple, nous allons représenter la loi $N(10, 3^2)$ et la loi $N(8, 5^2)$ sur le même graphique.

```
ggplot(data = data.frame(x = c(-5, 20)), aes(x)) +
  stat_function(fun = dnorm, args = list(mean = 10, sd = 3), colour = "blue") +
  stat_function(fun = dnorm, args = list(mean = 8, sd = 5), colour = "red") +
  labs(
    x = "x",
    y = "f(x)",
    title = "Les densités de N(10,3^2) et de N(8,5^2)"
  )
```



Nous pouvons aussi superposer une variable aléatoire discrète et une variable aléatoire continue. Dans l'exemple suivant, nous avons la loi $B(100, 0.2)$ et son approximation par la loi normale $N(20, 4^2)$.

```
n <- 100
p <- 0.2
m <- n*p
s <- sqrt(n*p*(1-p))
fbinom <- data.frame(x = 0:n, y = dbinom(0:n, n, p))
ggplot(fbinom, aes(x = x, y = y)) +
  geom_bar(width = 0.1, stat = "identity", colour = "blue") +
  stat_function(fun = dnorm, args = list(mean = m, sd = s), colour = "red") +
  labs(
    x = "Nombre de succès",
    y = "Probabilité",
    title = "La loi B(100,0.2) et la loi N(20,4^2)"
  )
```



Exercices

Vous devez répondre aux questions suivantes dans les espaces prévus à cette fin.

1. Soit $X \sim B(15, 0.4)$.

   a) Calculez la probabilité $P(X = 4)$.

```
# Écrivez votre réponse ici.
dbinom(4,15,0.4)
```

```
## [1] 0.1267758
```

   b) Calculez la probabilité $P(X \leq 4)$.

```
# Écrivez votre réponse ici.
pbinom(4,15,0.4)
```

```
## [1] 0.2172777
```

   c) Calculez la probabilité $P(X > 8)$.

```
# Écrivez votre réponse ici.
1-pbinom(8,15,0.4)
```

```
## [1] 0.09504741
```

   d) Calculez la probabilité $P(X \geq 8)$.

```r
# Écrivez votre réponse ici.
1-pbinom(7,15,0.4)
```

```
## [1] 0.2131032
```

2. Soit $X \sim N(0, 1^2)$.

   a) Calculez la probabilité $P(X < -0.5)$.

```r
# Écrivez votre réponse ici.
pnorm(-0.5,0,1)
```

```
## [1] 0.3085375
```

   b) Calculez la probabilité $P(X > 1.5)$.

```r
# Écrivez votre réponse ici.
1-pnorm(1.5,0,1)
```

```
## [1] 0.0668072
```

3. Soit $X \sim N(15, 3^2)$.

   a) Calculez la probabilité $P(16 \leq X \leq 20)$.

```r
# Écrivez votre réponse ici.
pnorm(20,15,3)-pnorm(16,15,3)
```

```
## [1] 0.321651
```

   b) Calculez la probabilité $P(X > 18)$.

```r
# Écrivez votre réponse ici.
1-pnorm(18,15,3)
```

```
## [1] 0.1586553
```

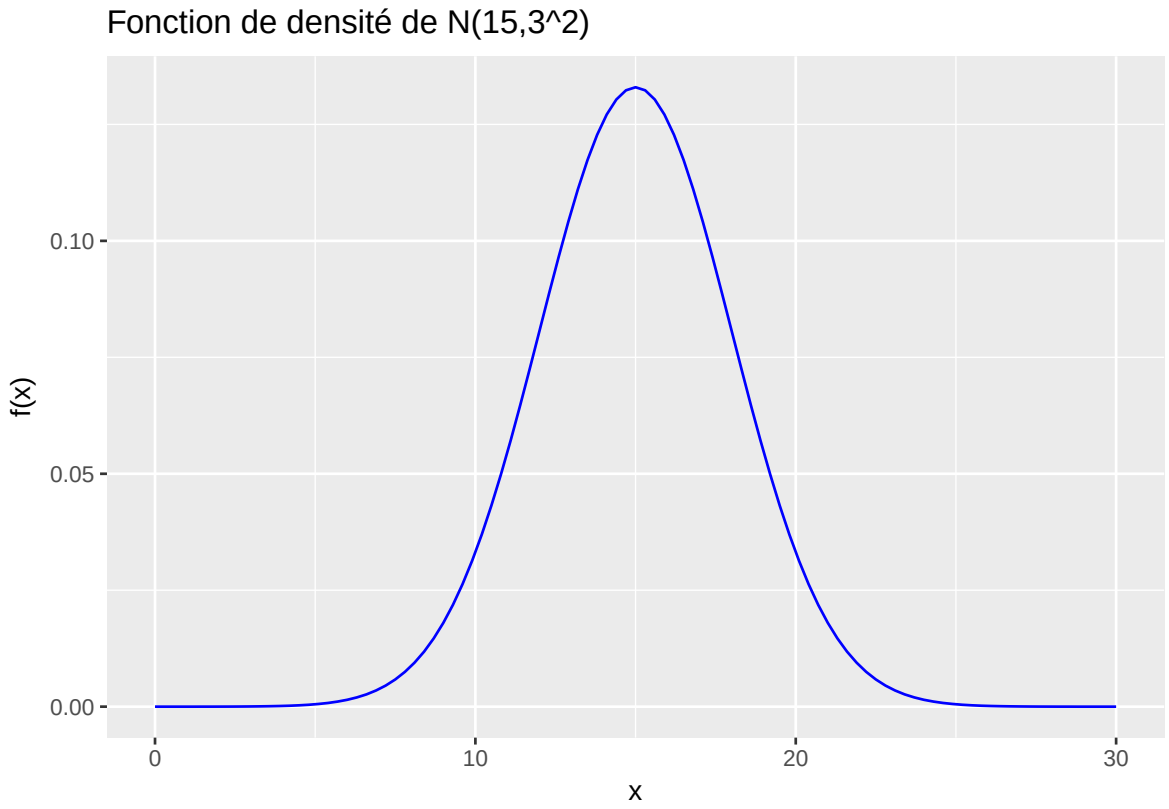
   c) Calculez la probabilité $P(X < 6)$.

```r
# Écrivez votre réponse ici.
pnorm(6,15,3)
```

```
## [1] 0.001349898
```

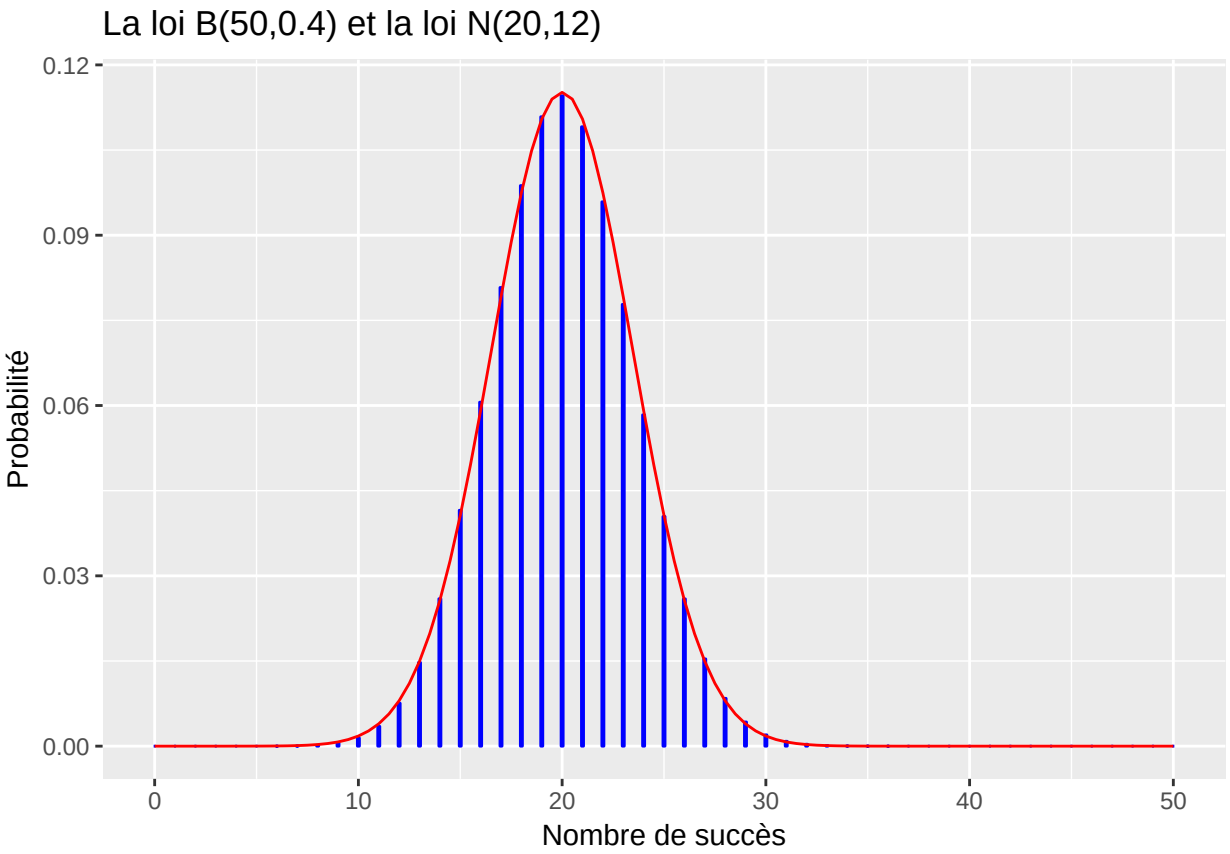
   d) Tracez la fonction de densité de la variable continue X.

```r
# Écrivez votre réponse ici.
ggplot(data = data.frame(x = c(0, 30)), aes(x)) +
stat_function(fun = dnorm, args = list(mean = 15, sd = 3), colour = "blue") +
labs(
  x = "x",
  y = "f(x)",
  title = "Fonction de densité de N(15,3^2)"
)
```



4. Représentez le graphe de la densité d'une variable $X \sim B(50, 0.4)$, puis ajoutez par dessus ce graphe celui de la densité d'une variable $Y \sim N(20, 12)$ (cela illustrera le fait que, lorsque $n > 30$, $np > 5$ et $n(1-p) > 5$, on peut approximer la loi binomiale $B(n, p)$ par la loi normale $N(np, np(1-p))$.

```
# Écrivez votre réponse ici.
n <- 50
p <- 0.4
m <- n*p
s <- sqrt(n*p*(1-p))
fbinom <- data.frame(x = 0:n, y = dbinom(0:n, n, p))
ggplot(fbinom, aes(x = x, y = y)) +
  geom_bar(width = 0.1, stat = "identity", colour = "blue") +
  stat_function(fun = dnorm, args = list(mean = m, sd = s), colour = "red") +
  labs(
    x = "Nombre de succès",
    y = "Probabilité",
    title = "La loi B(50,0.4) et la loi N(20,12)"
  )
```



5. On sait que la probabilité qu'une personne soit allergique à un certain médicament est égale à 0.001. On s'intéresse à un échantillon de 1000 personnes. On appelle X la variable aléatoire dont la valeur est le nombre de personnes allergiques dans l'échantillon.

   a) Déterminez la loi de probabilité de X. > Écrivez votre réponse ici. $X \sim B(1000, 0.001)$

   b) Calculez la probabilité qu'il y ait exactement deux personnes allergiques dans l'échantillon.

```
# Écrivez votre réponse ici.
dbinom(2,1000,0.001)
```

```
## [1] 0.1840317
```

   c) Calculez la probabilité qu'il y ait au moins deux personnes allergiques dans l'échantillon.

```
# Écrivez votre réponse ici.
1-pbinom(1,1000,0.001)
```

```
## [1] 0.2642411
```

6. Dans une entreprise de confection, il y a en moyenne 0,3 accident par semaine.

   a) Quelle est la probabilité qu'il y ait aucun accident au cours d'une semaine ?

```
# Écrivez votre réponse ici.
dpois(0,0.3)
```

```
## [1] 0.7408182
```

   b) Quelle est la probabilité qu'il y ait au moins deux accidents en trois semaines ?

```r
# Écrivez votre réponse ici.
1-ppois(1,0.3*3)
```

```
## [1] 0.2275176
```

c) Quelle est la probabilité qu'il y ait 38 semaines sans accident dans une année ?

```r
# Écrivez votre réponse ici.
p <- dpois(0,0.3)
dbinom(38,52,p)
```

```
## [1] 0.1222433
```

7. On a établi que le temps d'étude personnel hebdomadaire pour les étudiants du programme de techniques de l'informatique obéit à une loi normale de moyenne 4,58 heures et d'écart-type 1,31 heure.

   a) Quelle la probabilité qu'un étudiant étudie plus de trois heures par semaine ?

```r
# Écrivez votre réponse ici.
1-pnorm(3,4.58,1.31)
```

```
## [1] 0.8861119
```

   b) On choisit au hasard et avec remise 120 étudiants du programme techniques de l'informatique. Quelle est la probabilité que pour moins de 100 d'entre eux, le temps d'étude dépasse 3 heures?

```r
# Écrivez votre réponse ici.
1-pnorm(3,4.58,1.31)
```

```
## [1] 0.8861119
```

8. Dans un port d'une grande ville, il arrive en moyenne 10 bateaux avec une cargaison de pétrole par jour. Les infrastructures du port ne peuvent traiter qu'au maximum 15 bateaux avec une cargaison de pétrole en une journée.

   a) Quelle est la probabilité qu'une journée donnée qu'il y ait des bateaux qui ne puissent délivrer leur chargement de pétrole?

```r
# Écrivez votre réponse ici.
1-ppois(15,10)
```

```
## [1] 0.0487404
```

   b) Quelle est la probabilité que durant une semaine de 7 jours, il y ait 2 journées où les bateaux ne peuvent délivrer leur chargement de pétrole ?

```r
# Écrivez votre réponse ici.
p <- 1-ppois(15,10)
dbinom(2,7,p)
```

```
## [1] 0.0388591
```

Créer un tableau de fréquences

Pour créer un tableau de fréquences pour une variable aléatoire qualitative ou quantitative discrète, nous utilisons la commande `freq` de la librairie `questionr`.

Variable aléatoire qualitative ou quantitative discrète

Par exemple, pour représenter le tableau de fréquences pour la variable `Sex` pour les données du Titanic, on pourrait utiliser la commande suivante:

```
kable(freq(titanic$Sex,
     exclud = NA,
     cum = FALSE,
     total = TRUE))
```

	n	%
female	314	35.2
male	577	64.8
Total	891	100.0

L'option `exclud = NA` permet d'exclure les valeurs manquantes, l'option `cum = FALSE` permet de ne **pas** afficher les pourcentages cumulés et l'option `total = TRUE` permet d'ajouter le total.

Variable aléatoire quantitative continue

Par exemple, pour représenter le tableau de fréquences de la variable `Fare` (prix payé pour la croisière sur le Titanic) il faut tout d'abord créer le vecteur `Prix` et créer de nouvelles classes (pour se faire, nous utilisons la commande `cut`). Ensuite, nous utilisons la commande `levels` pour renommer nos classes.

```
Prix <- cut(titanic$Fare,breaks=c(0,20,40,60,80,600))
levels(Prix) <- c("0$ à 20$","20$ à 40$","40$ à 60$","60$ à 80$","80$ à 600$")
kable(freq(Prix,
     exclud = NA,
     cum = FALSE,
     total = TRUE))
```

	n	%
0$ à 20$	500	57.1
20$ à 40$	200	22.8
40$ à 60$	54	6.2
60$ à 80$	48	5.5
80$ à 600$	74	8.4
Total	876	100.0

Tableau de contingence

Par exemple, pour représenter le tableau de contingence pour les variables `Fare` (prix payé pour la croisière sur le Titanic) et `Survived` (Survie au naufrage (0 = Non ; 1 = Oui)) il faut tout simplement se créer un vecteur pour la variable `Survived` en utilisant la commande suivante:

```
Survie <- titanic$Survived
```

On peut afficher le tableau des fréquences obervées:

```
addmargins(table(Survie,Prix))
```

```
##       Prix
```

```
## Survie 0$ à 20$ 20$ à 40$ 40$ à 60$ 60$ à 80$ 80$ à 600$ Sum
##    0        358       114        23        23         17 535
##    1        142        86        31        25         57 341
##    Sum      500       200        54        48         74 876
```

ou alors le tableau des fréquences relatives:

```
prop(table(Survie,Prix),digits=2)
```

```
##        Prix
## Survie  0$ à 20$ 20$ à 40$ 40$ à 60$ 60$ à 80$ 80$ à 600$ Total
##   0      40.87    13.01      2.63      2.63      1.94      61.07
##   1      16.21     9.82      3.54      2.85      6.51      38.93
##   Total  57.08    22.83      6.16      5.48      8.45     100.00
```

Estimation de paramètres et tests d'hypothèses

Estimation de paramètres

Test d'hypothèses

Test d'indépendance

9. Nous voulons faire un test d'hypothèses pour savoir s'il y a une dépendance entre les variables `PClass` (qui indique la classe où se trouvait le passager à bord du Titanic) et `Survived` (qui indique si le passager survit (1) au naufrage ou meurt (0)).

   a) Créez le vecteur `Classe` qui indique dans quelle classe se situait les passagers à bord du Titanic.

   ```
   Classe <- titanic$Pclass
   ```

   b) Créez le vecteur `Survie` qui indique si le passager est mort ou si il a survécu au naufrage du Titanic.

   ```
   Survie <- titanic$Survived
   ```

   c) Créez un tableau de fréquences observées des variables `Classe` et `Survie`.

   ```
   Tableau <- table(Classe,Survie)
   ```

   d) Affichez le tableau de fréquences observées des variables `Classe` et `Survie`.

   ```
   Tableau
   ```

   ```
   ##       Survie
   ## Classe   0   1
   ##      1  80 136
   ##      2  97  87
   ##      3 372 119
   ```

   e) Dans cet échantillon, quel est le taux de mortalité en première classe?

   ```
   PropTableau <- prop(Tableau)
   PropTableau
   ```

   ```
   ##        Survie
   ## Classe  0     1     Total
   ##   1       9.0  15.3  24.2
   ```

```
##   2      10.9   9.8  20.7
##   3      41.8  13.4  55.1
##   Total  61.6  38.4 100.0
```

```
PropTableau[1,3]
```

```
## [1] 24.24242
```

f) Dans cet échantillon, quel est le taux de mortalité en première classe?

```
PropTableau <- prop(Tableau)
PropTableau
```

```
##        Survie
## Classe  0     1     Total
##   1       9.0  15.3  24.2
##   2      10.9   9.8  20.7
##   3      41.8  13.4  55.1
##   Total  61.6  38.4 100.0
```

```
PropTableau[3,3]
```

```
## [1] 55.10662
```

g) Faites un test d'indépendance du Khi-deux (test d'hypothèses) au seuil de signification de 0.5% avec le tableau, donnez votre décision et interprétez le résultat.(remarque : La commande pour faire un test d'indépendance du Khi-deux est `chisq.test(Tableau)`

```
chisq.test(Tableau)
```

```
##
##  Pearson's Chi-squared test
##
## data:  Tableau
## X-squared = 102.89, df = 2, p-value < 2.2e-16
```

Puisque la p-value est plus petite que $4.5492517 \times 10^{-23}$, alors nous rejettons H_0 et nous acceptons H_1. Ceci signifie qu'il y a un lien entre le fait qu'un passager survive et sa classe.

10.