Consultation statistique avec le logiciel R

C'est quoi la correction de continuité de Yates ?

Pr Jean R. Lobry

Quelques exercices de coloriage sous une courbe pour expliquer la correction de continuité de Yates – Où l'on démontre que Pierre-Simon Laplace est coupable de plagiat par anticipation.

1 Introduction

On rencontrait naguère[1] mention de la correction de continuité de Yates [3]. La source est disponible dans la figure 1. C'est assez simple à comprendre, mais il faut commencer par apprendre à colorier sous une courbe dans R.

2 Colorier sous une courbe

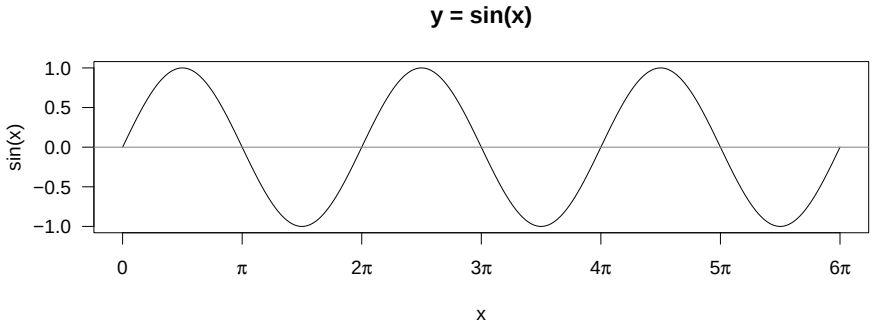
On représente la fonction $y = \sin x$ entre 0 et 6π :

```
x <- seq(from = 0, to = 3*2*pi, length = 1000)
y <- sin(x)
par(cex = 1.5)
plot(x, y, type = "l", las = 1, ylab = "sin(x)", xlab = "x",
main = "y = sin(x)", xaxt = "n")
abline( h = 0, col = grey(0.5))
axis(side = 1, at = (0:6)*pi, label = expression(0, pi,2*pi, 3*pi, 4*pi, 5*pi, 6*pi))
```

[1] ceci n'est plus vrai dans les versions actuelles de R, la qualité de la documentation d'un logiciel libre est inégalable.

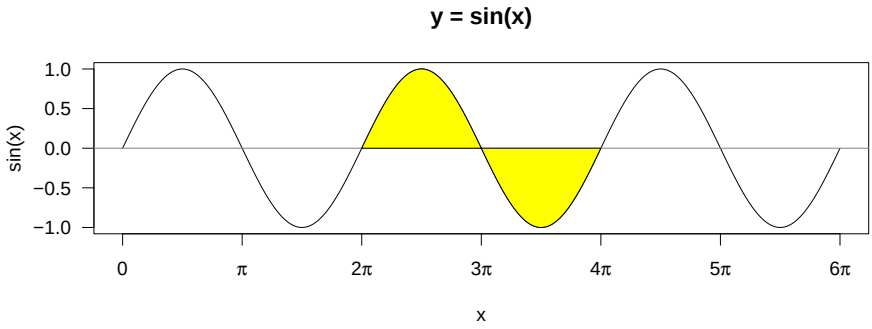
These discrepancies are primarily due to the fact that χ is a continuous distribution, whereas the distribution it is endeavouring to approximate is discontinuous. If we group the χ distribution, taking the half units of deviation from expectation as the group boundaries, we may expect to obtain a much closer approximation to the true distribution. This is equivalent to computing the values of χ^2 for deviations half a unit less than the true deviations, 8 successes, for example, being reckoned as $7\frac{1}{2}$, 2 as $2\frac{1}{2}$. This correction may be styled the *correction for continuity*, and the resultant value of χ denoted by χ'.

Figure 1: Copie d'écran d'une partie de la page 222 de l'article de Yates [3].



On voudrait maintenant colorier entre 2π et 4π pour mettre en évidence la période. On fait un premier essai avec la fonction `polygon()` en lui donnant les coordonnées x et y de la courbe telles que $2\pi \leq x \leq 4\pi$:

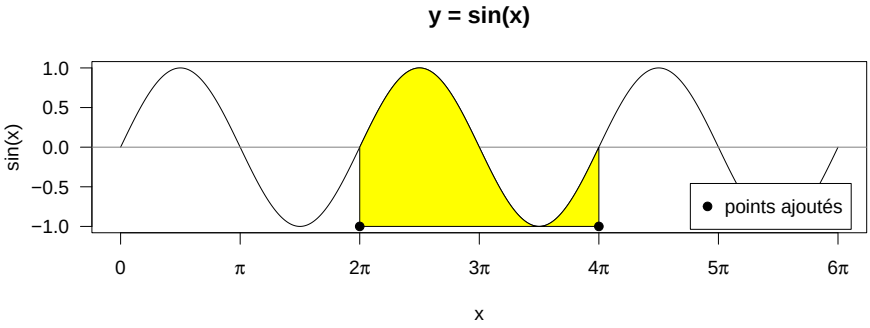
```
x <- seq(from = 0, to = 3*2*pi, length = 1000)
y <- sin(x)
par(cex = 1.5)
plot(x, y, type = "l", las = 1, ylab = "sin(x)", xlab = "x",
main = "y = sin(x)", xaxt = "n")
abline( h = 0, col = grey(0.5))
axis(side = 1, at = (0:6)*pi, label = expression(0, pi,2*pi, 3*pi, 4*pi, 5*pi, 6*pi))
bons <- which(x >= 2*pi & x <= 4*pi)
x.bons <- x[bons]
y.bons <- y[bons]
polygon(x = x.bons, y = y.bons, col = 'yellow')
```



Ooooups ! Ce n'est pas exactement ce que l'on voulait : on veut colorier sous la courbe. Pour celà[2] il faut compléter le polygone avec les points de coordonnées $(\min x, \min y)$ et $(\max x, \min y)$:

```
x <- seq(from = 0, to = 3*2*pi, length = 1000)
y <- sin(x)
par(cex = 1.5)
plot(x, y, type = "l", las = 1, ylab = "sin(x)", xlab = "x",
main = "y = sin(x)", xaxt = "n")
abline( h = 0, col = grey(0.5))
axis(side = 1, at = (0:6)*pi, label = expression(0, pi,2*pi, 3*pi, 4*pi, 5*pi, 6*pi))
polygon(x = c(min(x.bons), x.bons, max(x.bons)), y = c(min(y.bons), y.bons, min(y.bons)), col = 'yellow')
points(min(x.bons), min(y.bons), pch = 19)
points(max(x.bons), min(y.bons), pch = 19)
legend('bottomright', inset = 0.02, pch = 19, legend = 'points ajoutés', bg = 'white')
```

[2] Je m'autorise par licence scientifique à écrire « pour celà » en lieu et place de « pour faire ce là ». Que les les philologues m'excusent, mais ici cela me semble être justifié.

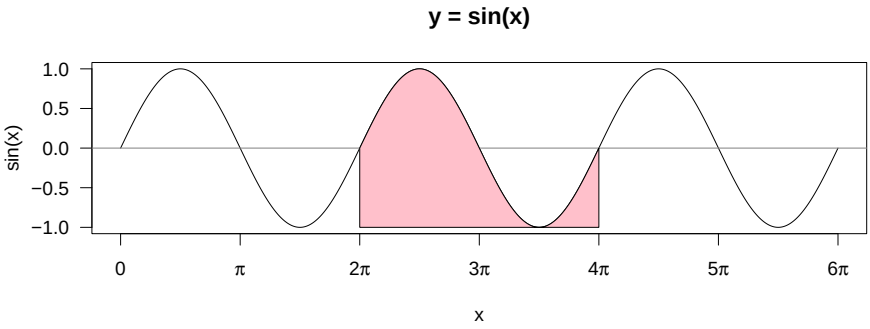


C'est bien mieux. Mais en fait on n'est pas obligé de vouloir avoir la base du rectangle au minimum de y. On écrit une petite fonction `polycurve()` pour automatiser la tâche, et ou l'ordonnée de la base du rectangle est un paramètre ajustable :

```
polycurve <- function(x, y, base.y = min(y), ...) {
  polygon(x = c(min(x), x, max(x)), y = c(base.y, y, base.y), ...)
}
```

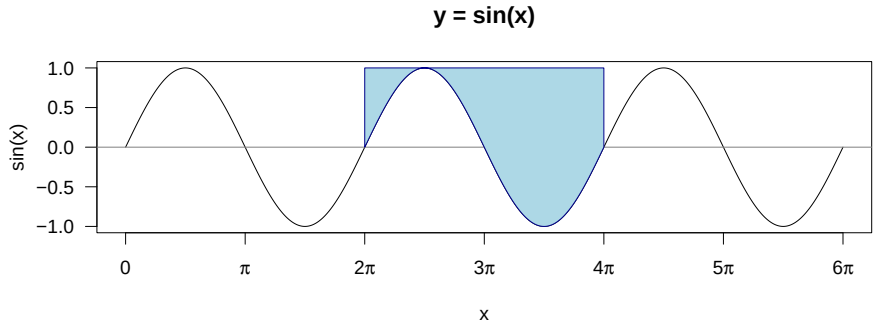
Essayons de voir si cela marche avec la valeur par défaut pour l'ordonnée de la base :

```
x <- seq(from = 0, to = 3*2*pi, length = 1000)
y <- sin(x)
par(cex = 1.5)
plot(x, y, type = "l", las = 1, ylab = "sin(x)", xlab = "x",
main = "y = sin(x)", xaxt = "n")
abline( h = 0, col = grey(0.5))
axis(side = 1, at = (0:6)*pi, label = expression(0, pi,2*pi, 3*pi, 4*pi, 5*pi, 6*pi))
polycurve(x.bons, y.bons, col = 'pink')
```



Essayons maintenant de colorier la courbe au dessus :

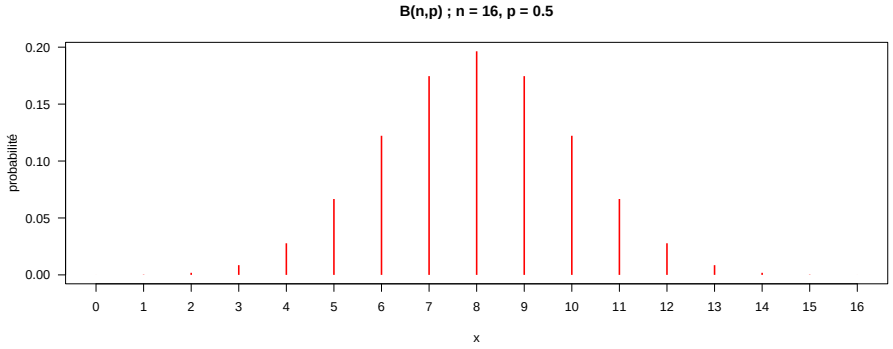
```
x <- seq(from = 0, to = 3*2*pi, length = 1000)
y <- sin(x)
par(cex = 1.5)
plot(x, y, type = "l", las = 1, ylab = "sin(x)", xlab = "x",
main = "y = sin(x)", xaxt = "n")
abline( h = 0, col = grey(0.5))
axis(side = 1, at = (0:6)*pi, label = expression(0, pi,2*pi, 3*pi, 4*pi, 5*pi, 6*pi))
polycurve(x.bons, y.bons, base.y = 1, col = 'lightblue', border = 'darkblue')
```



3 La correction de continuité de Yates

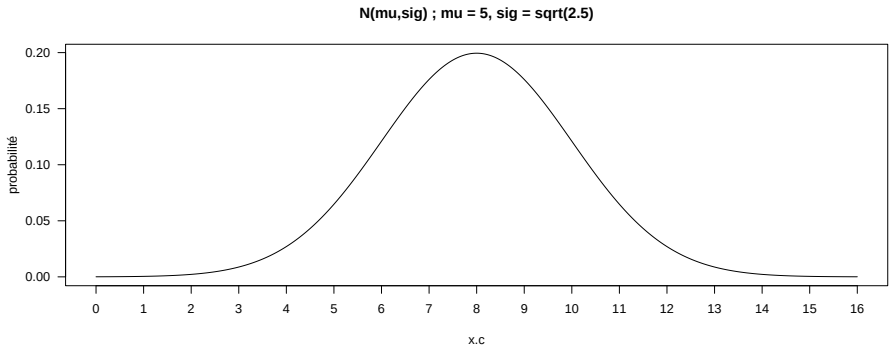
ON ILLUSTRE ICI la correction de continuité de YATES dans le cas de la distribution binomiale, mais ceci est valable pour toute approximation d'une distribution discrète par une distribution continue. La distribution binomiale est une distribution discrète : une variable aléatoire binomiale ne peut prendre que des valeurs entières entre 0 et n.

```
n <- 16
p <- 0.5
x <- 0:n
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "B(n,p) ; n = 16, p = 0.5",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
```



PRENONS comme ci-dessus une binomiale de paramètre $n = 16$ et $p = 0.5$. La moyenne, μ, vaut 8 et la variance σ^2 vaut 9. Comment est-ce qu'une loi normale de moyenne μ et d'écart type σ approxime cette loi binomiale ?

```
mu <- n*p
sd <- sqrt(n*p*(1-p))
x.c <- seq(from = 0, to = n, length = 1000)
plot(x.c, dnorm(x.c, mu, sd), main = 'N(mu,sig) ; mu = 5, sig = sqrt(2.5)',
ylab = "probabilité", las = 1, xaxt = "n", type = 'l')
axis(1,x,x)
```



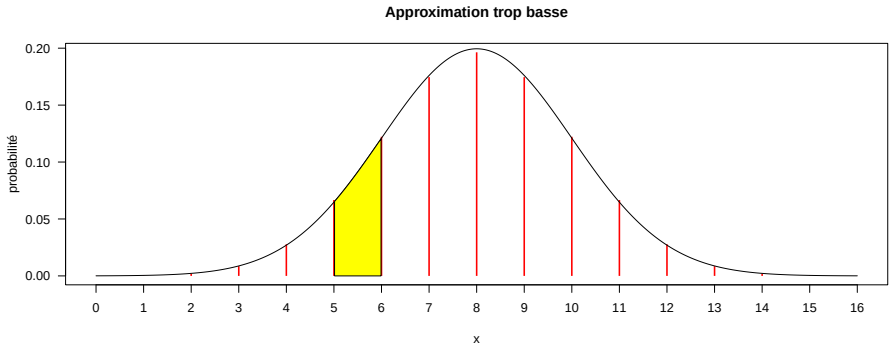
La loi normale est une distribution continue. Il nous faut donc approximer la hauteurs des bâtons de la binomiale par une surface sous la courbe de la loi normale. Essayons pour $P(X = 6)$ par exemple. La probabilité exacte donnée par la binomiale est 0.1221924 :

```
dbinom(6,n,p)
```

```
[1] 0.1221924
```

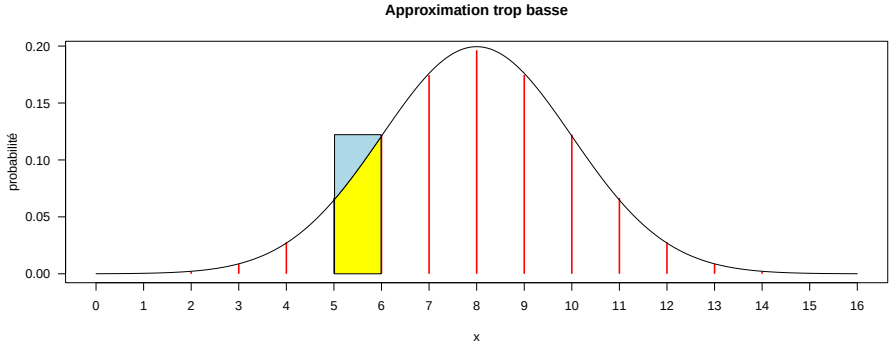
Si nous intégrons la fonction de densité de la loi normale de 5 à 6, on est toujours trop bas :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "Approximation trop basse",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 5 & x.c <= 6)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
```



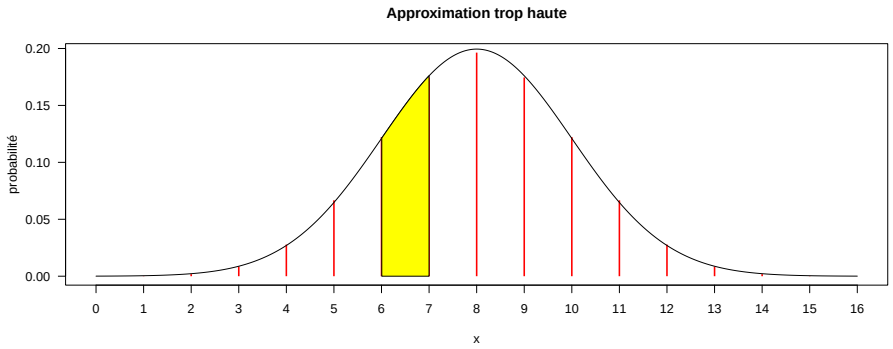
Pour atteindre la bonne valeur, il nous manque la région en bleu ci-dessous :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "Approximation trop basse",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 5 & x.c <= 6)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = dbinom(6,n,p), col = 'lightblue')
```



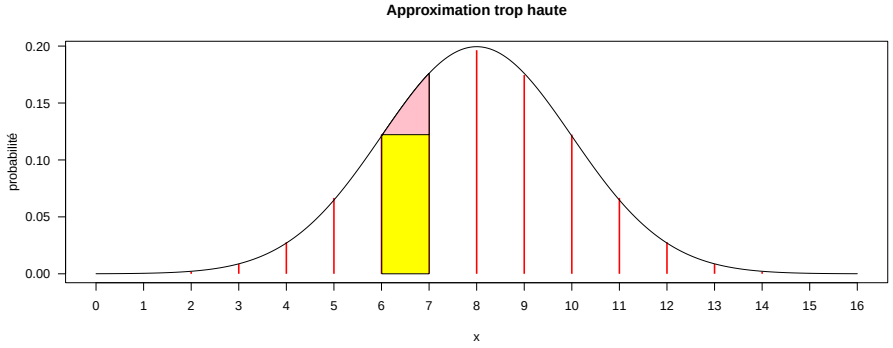
Si nous intégrons la fonction de densité de la loi normale de 6 à 7, on est toujours trop haut :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "Approximation trop haute",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 6 & x.c <= 7)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
```



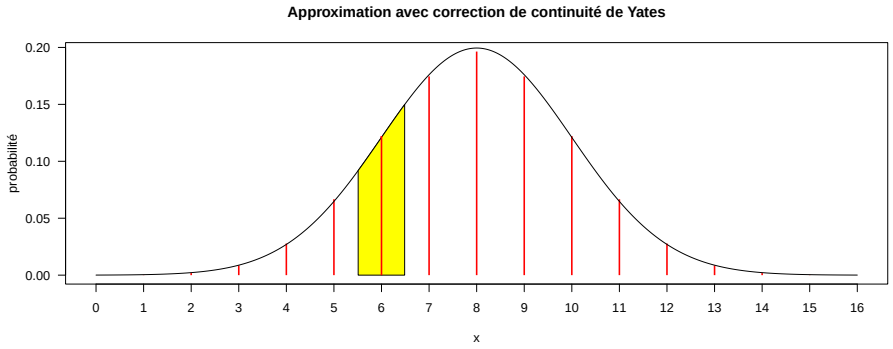
Pour atteindre la bonne valeur, il nous avons compté en trop la région en rose ci-dessous :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "Approximation trop haute",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 6 & x.c <= 7)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = dbinom(6,n,p), col = 'pink')
```



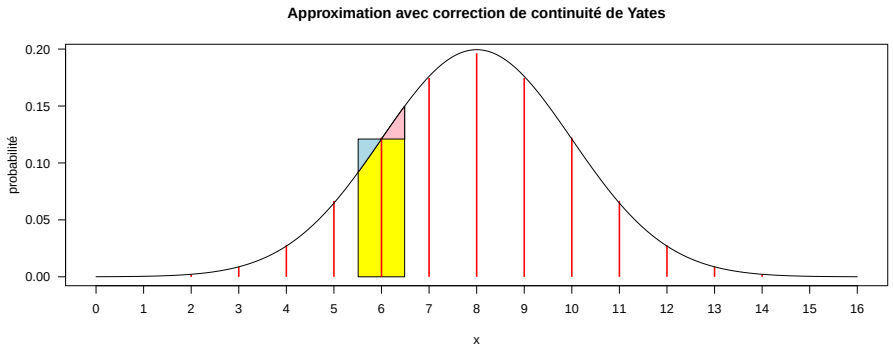
Mais si nous intégrons la fonction de densité de la loi normale de 5.5 à 6.5 nous allons avoir une bien meilleure approximation :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "Approximation avec correction de continuité de Yates",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 5.5 & x.c <= 6.5)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
segments(6, 0, 6, dbinom(6,n,p), col = 'red', lwd = 2, lend = 'butt')
```



On se doute que l'approximation va être meilleure puisque que nous avons omis de compter la région en bleu et que nous avons compté en trop la région en rose :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "Approximation avec correction de continuité de Yates",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 5.5 & x.c <= 6.5)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
segments(6, 0, 6, dbinom(6,n,p), col = 'red', lwd = 2, lend = 'butt')
bons.bas <- which(x.c >= 5.5 & x.c < 6)
polycurve(x.c[bons.bas], dnorm(x.c, mu, sd)[bons.bas], base.y = dnorm(6, mu, sd), col = 'lightblue')
bons.haut <- which(x.c > 6 & x.c <= 6.5)
polycurve(x.c[bons.haut], dnorm(x.c, mu, sd)[bons.haut], base.y = dnorm(6, mu, sd), col = 'pink')
```

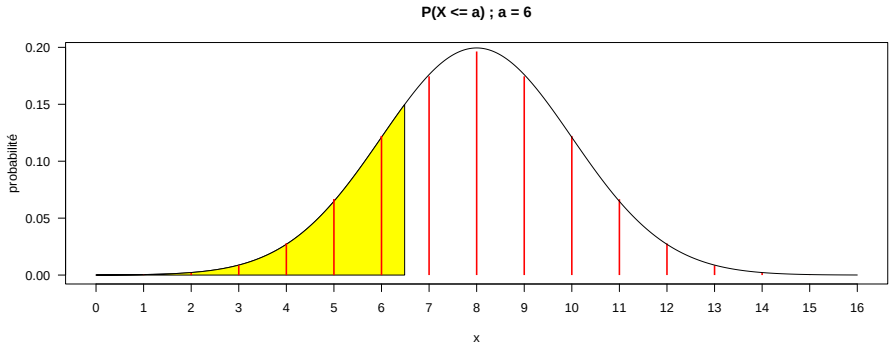


Bien entendu, nous n'allons pas compenser exactement la région rose par la région bleu, mais c'est quand même mieux que nos approximations hautes et basses précédentes. C'est tout, vous avez compris l'essentiel de l'idée de la correction de continuité de Yates [3]. Vous pouvez quitter ce document et reprendre une activité normale.

4 Du point de vue des fonctions de répartition

Plus généralement, si nous voulons approximer la probabilité binomiale $P(X \leq a)$, nous allons intégrer la fonction de densité de la loi normale de $-\infty$ à $a+0.5$. Dans l'exemple ci-dessous on a $a = 6$, la probabilité exacte avec la loi binomiale donne 0.2272491, et l'approximation avec la loi normale avec correction de continuité 0.2266274 :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "P(X <= a) ; a = 6",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c <= 6.5)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
segments(0:6, 0, 0:6, dbinom(0:6,n,p), col = 'red', lwd = 2, lend = 'butt')
```

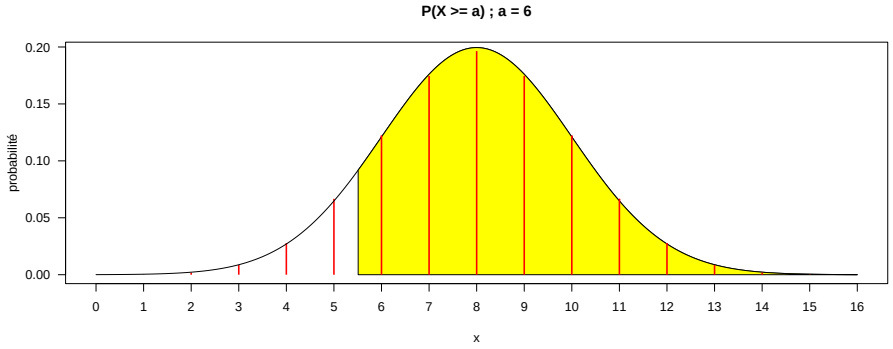



```
c(sum(dbinom(0:6, n, p)),
pbinom(6,n,p),
pnorm(6.5,mu,sd))
```

```
[1] 0.2272491 0.2272491 0.2266274
```

Si nous voulons approximer la probabilité binomiale $P(X \geq a)$, nous allons intégrer la fonction de densité de la loi normale de $a - 0.5$ à $+\infty$. Dans l'exemple ci-dessous on a $a = 6$, la probabilité exacte avec la loi binomiale donne 0.8949432, et l'approximation avec la loi normale avec correction de continuité 0.8943502 :

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "P(X >= a) ; a = 6",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(x.c >= 5.5)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
segments(6:16, 0, 6:16, dbinom(6:16,n,p), col = 'red', lwd = 2, lend = 'butt')
```

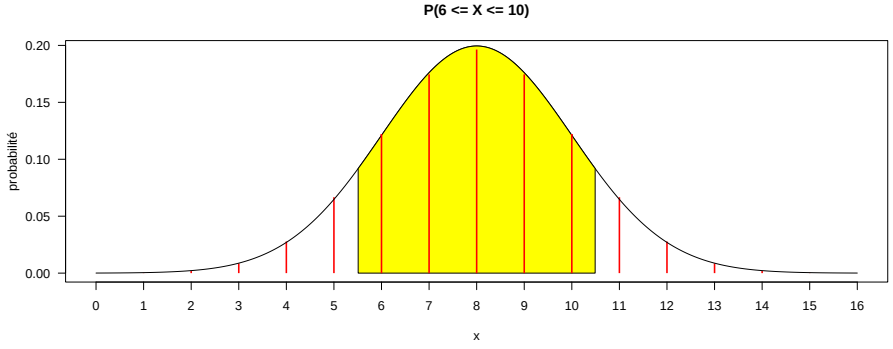



```
c(sum(dbinom(6:16, n, p)),
1 - pbinom(5,n,p),
1 - pnorm(5.5,mu,sd))
```

```
[1] 0.8949432 0.8949432 0.8943502
```

Un petit exercice pour ceux qui suivent. En utilisant uniquement la fonction de répartition de la loi binomiale, `pbinom()`, et la fonction de répartition de la loi normale, `pnorm()`, on veut calculer $P(6 \leq X \leq 10)$:

```
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 2, lend = "butt",
main = "P(6 <= X <= 10)",
ylab = "probabilité", las = 1, xaxt = "n")
axis(1,x,x)
lines(x.c, dnorm(x.c, mu, sd))
bons <- which(5.5 <= x.c & x.c <= 10.5)
polycurve(x.c[bons], dnorm(x.c, mu, sd)[bons], base.y = 0, col = 'yellow')
segments(6:10, 0, 6:10, dbinom(6:10,n,p), col = 'red', lwd = 2, lend = 'butt')
```



Calculer la probabilité exacte avec la binomiale[3] :

```
[1] 0.7898865
```

Donner l'approximation avec la loi normale sans correction de continuité :

```
[1] 0.6826895
```

Donner l'approximation avec la loi normale avec correction de continuité :

```
[1] 0.7887005
```

5 Application

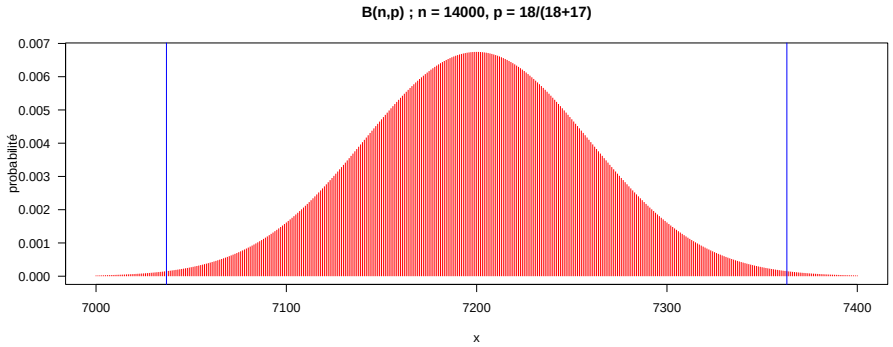
Pierre-Simon Laplace (1749-1827).

Pierre-Simon Laplace, Pair de France ; Grand Officier de la Légion d'honneur ; l'un des quarante de l'Académie française ; de l'Académie des Sciences ; membre du Bureau des Longitudes de France ; des Sociétés royales de Londres et de Göttingue ; des Académies des Sciences de Russie, de Danemark, de Suède, de Prusse, des Pays-Bas, d'Italie, etc.[4], nous propose [1] l'exercice suivant :

> Supposons, par exemple, que les facilités des naissances des garçons et des filles soient dans le rapport de 18 à 17, et qu'il naisse dans une année 14000 enfants ; on demande la probabilité que le nombre des garçons ne surpassera pas 7363, et ne sera pas moindre que 7037.

L'auteur donne $P(7037 \leq X \leq 7363) \approx 0.994303$ (*cf.* figure 2). Représentons le problème graphiquement :

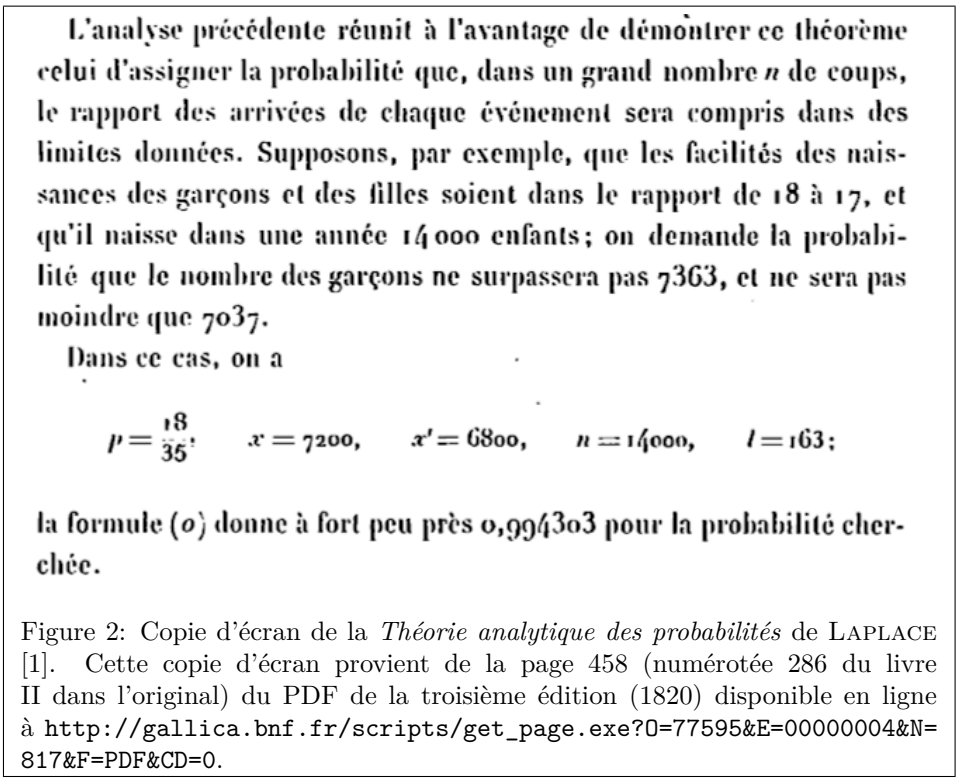
```
n <- 14000
p <- 18/(18+17)
x <- 7000:7400
plot(x, dbinom(x, n, p), type = "h", col = "red", lwd = 1, lend = "butt",
main = "B(n,p) ; n = 14000, p = 18/(18+17)",
ylab = "probabilité", las = 1)
abline(v = 7037, col = "blue")
abline(v = 7363, col = "blue")
```



Solution exacte avec la binomiale :

[3] Ce n'est pas du jeu que de faire : `sum(dbinom(6:10, n, p))`, on n'a droit qu'à la fonction de répartition `pbinom()` ici.

[4] Non, vous n'êtes pas en train de lire la biographie de S.A.S. Malko Linge de Gérard de Villiers (`http://www.sasmalko.com`), tout ceci a réellement existé.



L'analyse précédente réunit à l'avantage de démontrer ce théorème celui d'assigner la probabilité que, dans un grand nombre n de coups, le rapport des arrivées de chaque événement sera compris dans des limites données. Supposons, par exemple, que les facilités des naissances des garçons et des filles soient dans le rapport de 18 à 17, et qu'il naisse dans une année 14000 enfants; on demande la probabilité que le nombre des garçons ne surpassera pas 7363, et ne sera pas moindre que 7037.

Dans ce cas, on a

$$p = \frac{18}{35}, \quad x = 7200, \quad x' = 6800, \quad n = 14000, \quad l = 163;$$

la formule (o) donne à fort peu près 0,994303 pour la probabilité cherchée.

Figure 2: Copie d'écran de la *Théorie analytique des probabilités* de Laplace [1]. Cette copie d'écran provient de la page 458 (numérotée 286 du livre II dans l'original) du PDF de la troisième édition (1820) disponible en ligne à `http://gallica.bnf.fr/scripts/get_page.exe?O=77595&E=00000004&N=817&F=PDF&CD=0`.

```
(exact <- pbinom(7363,n,p) - pbinom(7036,n,p))
```

```
[1] 0.9943058
```

Solution approximative avec la loi normale sans correction de continuité :

```
mu <- n*p
sd <- sqrt(n*p*(1-p))
(approx1 <- pnorm(7363,mu,sd)-pnorm(7037,mu,sd))
```

```
[1] 0.9941546
```

Solution approximative avec la loi normale avec correction de continuité :

```
(approx2 <- pnorm(7363+0.5,mu,sd)-pnorm(7037-0.5,mu,sd))
```

```
[1] 0.9943039
```

La fonction `all.equal()` nous permet de tester dans R l'égalité de valeurs numériques en tenant compte de l'imprécision numérique. Entre la valeur exacte et l'approximation normale nous avons :

```
all.equal(exact, approx1)
```

```
[1] "Mean relative difference: 0.0001521415"
```

Entre la valeur exacte et l'approximation normale avec correction de continuité nous avons :

```
all.equal(exact,approx2)
```

```
[1] "Mean relative difference: 1.918929e-06"
```

Entre la valeur exacte et l'approximation de LAPLACE nous avons :

```
Laplace <- 0.994303
all.equal(exact, Laplace)
```

```
[1] "Mean relative difference: 2.859076e-06"
```

LAPLACE devait très certainement utiliser une correction de continuité à la YATES :

```
all.equal(Laplace, approx2)
```

```
[1] "Mean relative difference: 9.401495e-07"
```

Cela n'a rien d'étonnant, PEARSON faisait remarquer page 147 de [2] que cette correction était utilisée depuis fort longtemps par les statisticiens :

> 26. *The correction for continuity.* In the 2×2 table connexion, the improvement obtained by taking the normal integral (i) from $x = a - \frac{1}{2}$ if $a > \bar{a}$ or (ii) from $x = a + \frac{1}{2}$ if $a < \bar{a}$ (so that we are summing for the lower tail), was pointed out by Yates (1934) and has often been termed 'Yates's correction for continuity'. It is, however, the natural adjustment to make on the basis of the Euler-Maclaurin theorem, when approximating to a sum of ordinates by an integral and without wishing to detract from the value of Yates's suggestion in this particular problem, it should be pointed out that the adjustment was used by statisticians well before 1934, when employing a normal or skew curve to give the sum of terms of a binomial or hypergeometric series.*
>
> * The method was in use in the Department of Applied Statistics when I joined the staff in 1921, and may have been current many years before that.

Pierre-Simon LAPLACE est donc bien coupable de plagiat par anticipation...

References

[1] P.-S. Laplace. *Théorie analtique des probabilités*. Veuve Courcier, Paris, France, 1812.

[2] E.S. Pearson. The choice of statistical tests illustrated on the interpretation of data classed in a 2 x 2 table. *Biometrika*, 34:139–167, 1947.

[3] F. Yates. Contingency table involving small numbers and the χ^2 test. *Supplement to the Journal of the Royal Statistical Society*, 1(2):217–235, 1934.