

```

# Approximation de pi
# un cercle de rayon 1, aire = pi*1*1 = pi = 3.1415926...
symbols(0, 0, circles = 1, inches = F, fg = 2)
# un carre (circonscrit) de cote 2, aire = 2*2 = 4
symbols(0, 0, squares = 2, inches = F, add = T, fg = 4)
# On tire un point au hasard dans le carre. Quelle est la
# proba qu'il soit aussi dans le disque ? pi/4 ~ 0.785
pi/4
# tirage d'un point
a = 2*runif(1)-1; b = 2*runif(1)-1
points(a, b);
# tirage de 1000 points
a = 2*runif(1000)-1; b = 2*runif(1000)-1
points(a, b, col = "blue");
# combien de points dans le disque ?
# On s'attend a ce que la proportion de points dans le
# disque soit egale au rapport de l'aire du disque par
# l'aire du carre, c.a.d. pi/4 ~ 0.785
c = (a^2+b^2<1);
sum(c); sum(c)/1000
# colorer ces points
d = which(a^2+b^2<1)
points(a[d], b[d], col = "red")
# estimation de pi
sum(c)*4/1000
# creation d'une serie de 400 valeurs approchees de pi
N = 400; p = 1:N
for (i in 1:N){
  a = 2*runif(1000)-1; b = 2*runif(1000)-1
  c = (a^2+b^2<1);
  p[i] = sum(c)*4/1000
}
# statistique descriptive
mean(p); sd(p); min(p); max(p)
# Pour visualiser la repartition des valeurs des p[i], on
# decoupe le segment [min(p), max(p)] en 20 intervalles
# et calcule pour chacun de ces intervalles le nombre
# d'indice i tel que p[i] tombe dans l'intervalle. On
# affiche le resultat (les 20 nombres obtenus) sous une
# forme graphique qu'on appelle histogramme.
hist(p, breaks = 20)
# proprietes de l'estimateur de pi
# On observe des fluctuations des hauteurs des batons
# autour des valeurs predites par le graphe de dnorm().
hist(p, breaks = 20, probability = T)
s = 4/1000*sqrt(1000*pi/4*(1-pi/4))
x = seq(min(p), max(p), 0.01)
lines(x, dnorm(x, pi, s), col = "red")
# controle de la regle des 95%
# L'intervalle [pi-2s, pi+2s] est la plage de normalite
# au niveau de confiance 95%.
sum(p > pi-2*s & p < pi+2*s)/N
sum(p > pi-3*s & p < pi+3*s)/N
sum(p > pi-s & p < pi+s)/N

```