

```
#####
# two-sided Pareto distribution #
# simulation et estimation      #
# exemple de la semaine 9      #
#####
chi2.pareto = fonction(x, intervals)
{ nx = length(x); k = length(intervals)-1
  n.empi = 1:k; n.theo = 1:k
  est = pareto.MLE(x)
  for(i in 1: k)
  { n.empi[i] = sum(x >= intervals[i] & x < intervals[i+1])
    n.theo[i] = nx*(ppareto(intervals[i+1], 1, est[2])-ppareto(intervals[i], 1, est[2]))
  }
  if(any(n.theo<5))
  { printf("WARNING : effectif theorique trop petit");
    break
  }
  D = sum((n.empi-n.theo)^2/n.theo)
  list(p.value = 1 - pchisq(D, k-1-1))
}

pareto.MLE <- fonction(X)
{
  n <- length(X)
  m <- min(X)
  a <- n/sum(log(X)-log(m))
  return( c(m,a) )
}

library(VGAM)
dsim.r = rpareto(5000, 1, 1.5)
dsim.l = rpareto(5000, 1, 1.5)
dsim = c(dsim.r-1, 1-dsim.l)
hist(dsim, br = 500)

dl = 1-dsim[dsim<0]; dr = dsim[dsim>=0]+1
hist(dl); hist(dr)
est.l = pareto.MLE(dl); est.r = pareto.MLE(dr)
ks.test(dl,"ppareto", est.l[1], est.l[2])$p.value
ks.test(dr,"ppareto", est.r[1], est.r[2])$p.value

m = median(dl); l = m/4;
intervals = c(seq(1, 2*m-1, length.out = 8), Inf)
chi2.pareto (dl, intervals)$p.value

m = median(dr); l = m/4;
intervals = c(seq(1, 2*m-1, length.out = 8), Inf)
chi2.pareto (dr, intervals)$p.value
```