

ENSTA — Filtrage bayésien et approximation particulière

Fonction MATLAB pour la redistribution multinomiale

Cette fonction MATLAB, baptisée `fct_multi.m`, prend en entrée les arguments suivants

- un tableau \mathbf{x} de dimension $d \times M$ contenant les positions des M particules, à valeurs d -dimensionnelles, avant redistribution,
- un tableau \mathbf{a} de dimension $1 \times M$ contenant les poids des M particules, non-nécessairement normalisés, avant redistribution,
- un entier N contenant la taille de l'échantillon désiré en sortie,

et elle rend en sortie

- un tableau $\mathbf{x1}$ de dimension $d \times N$ contenant les positions des N particules, à valeurs d -dimensionnelles, sélectionnées au sein de la population donnée en entrée dans le tableau \mathbf{x} et redistribuées selon les poids donnés en entrée dans le tableau \mathbf{a} .

L'algorithme utilisé est celui présenté dans le cours : on simule d'abord dans le tableau \mathbf{uu} une N -statistique d'ordre uniforme, qu'on interclasse ensuite avec la suite formée dans le tableau \mathbf{p} par la somme cumulée des M poids donnés en entrée dans le tableau \mathbf{a} .

```

function [x1,diff,ordre_k] = fct_multi(x,a,N)
%
u_tild = zeros(1,N);
expo = zeros(1,N);
alpha = zeros(1,N);
u_ord = zeros(1,N);
uu = zeros(1,N+1);
p = zeros(1,N);
ordre_k = zeros(1,N);
x1 = zeros(size(x,1),N);
kk = 0;
diff = 0;
%
a = a./sum(a);
p = cumsum(a);
u_tild = rand(1,N);
expo = N:-1:1;
expo = (ones(1,N)./expo);
alpha = u_tild.^expo;
u_ord = cumprod(alpha);
u_ord(N:-1:1) = u_ord;
uu = [u_ord,2];
i = 1;
for j=1:size(a,2)
    k = 0;
    while uu(i)<p(j)
        x1(:,i) = x(:,j);
        ordre_k(i) = j;
        i = i+1;
        k = 1;
    end
    if k==1
        kk = 1+kk;
    end
end
diff = kk;

```

Table 1: Redistribution multinomiale