

Séances n°3 : transformation affine de variable aléatoire

Module de Génie Informatique en Licence 3 d'électronique

Laboratoire LTSI - UMR INSERM 642 - Université de Rennes1

Nous allons lors de ce TP voir l'effet des paramètres de position et de dispersion sur une variable aléatoire donnée. Autrement dit, quel comportement statistique adopte la variable $Y(a, b) = aX + b$ en fonction du couple (a, b) de paramètres lorsque X suit une certaine loi? Nous considérerons à titre d'exemple que X suit tout d'abord une loi normale (d'espérance et de variance unité), puis une loi uniforme sur $[0, 1]$.

1) Tout d'abord, calculons théoriquement l'espérance et la variance de $Y(a, b)$ en fonction de a et b pour les deux lois de X considérées ci-dessus.

2) Puis, vérifions ce résultat par simulation en utilisant les estimateurs empiriques de l'espérance et de la variance d'une variable aléatoire définis au TP précédent.

2.1) Il nous faut dans un premier temps construire sous Matlab un vecteur x de N réalisations de la variable aléatoire X (on prendra $N = 1000$) pour chacune des deux lois considérées.

2.2) Il faut ensuite se donner deux vecteurs, vec_a et vec_b , de valeurs pour a et b . On pourra par exemple faire varier a entre 0 et 20 avec un pas unité. Quant à b , on pourra le faire varier entre 0 et 100 également avec un pas unité.

2.3) La troisième étape consiste à construire sous Matlab une fonction prenant en argument les vecteurs x , vec_a , vec_b et renvoyant comme résultats deux matrices, $mYab$ et $varYab$. La (n, p) -ième composante de la matrice $mYab$ contiendra l'estimée de l'espérance de $Y(a, b)$ avec pour valeurs respectives de a et b la n -ième composante de vec_a et à la p -ième composante de vec_b . De même, la (n, p) -ième composante de la matrice $varYab$ contiendra l'estimée de la variance de $Y(a, b)$ avec pour valeurs respectives de a et b la n -ième composante de vec_a et à la p -ième composante de vec_b . Les deux matrices, $mYab$ et $varYab$, seront donc de taille $(L_a \times L_b)$ si L_a et L_b désignent respectivement les longueurs des vecteurs vec_a et vec_b . En outre, on construira ces deux matrices de deux manières différentes, tout d'abord avec deux boucles *for* emboîtées, puis à l'aide d'une seule boucle *for*. Notons que sous Matlab toute fonction est sauvegardée dans un fichier distinct du programme principal en ajoutant l'extension ".m" au nom du fichier. Le corps de ce fichier, pour l'exemple présent, aura l'allure suivante :

```
function [mYab,varYab] = taffine(x,vec_a,vec_b)
```

```
    [
      :
      mYab = ...;
      varYab = ...;
```

Le nom de fonction "*taffine*" a été choisi de manière arbitraire. Il est alors recommandé de donner le même nom au fichier de la fonction lors de sa sauvegarde.

2.4) Enfin, on appellera cette fonction dans le programme principal et on représentera sur deux figures différentes les graphes correspondant aux matrices $mYab$ et $varYab$ renvoyées par la fonction. On utilisera pour cela la fonction *surf* de Matlab. On sera donc en mesure de décrire l'influence des paramètres a et b .