Séance n°2 : estimateurs empiriques de statistiques

Module de Génie Informatique en Licence 3 d'électronique

Laboratoire LTSI - UMR INSERM 642 - Université de Rennes1

Il est possible sous Matlab de générer des variables de loi uniforme. Ceci se fait à l'aide de la fonction "*rand(.)*" qui s'utilise comme la fonction "*randn(.)*".

Ainsi dans le domaine des télécommunications radios, bien que l'information soit lors de son transport sous forme analogique, lors de la réception une conversion analogique/numérique est mise en oeuvre. Le traitement d'information qui suit cette étape se fait alors sur des signaux plus communément appelés modulations numériques. En pratique il existe un certain nombre de modulations numériques, chacune d'elle possédant des caractéristiques différentes (temporelles, spectrales, etc...). Une modulation couramment utilisée est la QPSK filtrée NRZ. Du point de vue du récepteur, elle peut être considérée comme une séquence de variables aléatoires i.i.d. (indépendantes et identiquement distribuées) prenant leur valeur de manière équiprobable dans l'ensemble $\{1, \mathrm{i}, -1, -\mathrm{i}\}$. Nous allons sous Matlab générer un tel processus.

Tout d'abord construisons un vecteur $\boldsymbol{x}$ représentant une réalisation (ou un tirage) de N variables aléatoires uniformément distribuées dans l'intervalle $[0, 1]$. Soit g l'application définie par :

$$g(x_j) = \begin{cases} 1 \text{ si } x_j \in [0, 1/4[\\ \mathrm{i} \text{ si } x_j \in [1/4, 1/2[\\ -1 \text{ si } x_j \in [1/2, 3/4[\\ -\mathrm{i} \text{ si } x_j \in [3/4, 1[\end{cases}$$

Appliquons à chaque composante de $\boldsymbol{x}$ la fonction g afin d'obtenir le vecteur $\boldsymbol{y}$. On pourra utiliser le fait que le script suivant :

$$v = (x >= 0.25 \ \& \ x < 0.5);$$

a pour effet d'affecter à la j-ième composante de $\boldsymbol{v}$ la valeur 1 si la j-ième composante de $\boldsymbol{x}$ appartient à l'intervalle $[1/4, 1/2[$ et la valeur 0 sinon. Ainsi le vecteur $\boldsymbol{y}$ obtenu est une réalisation du processus QPSK filtré NRZ. Représentons alors graphiquement la constellation de la dite QPSK, autrement dit les valeurs du processus $\boldsymbol{y}$.

Dans un second temps, nous allons estimer l'espérance et la variance des variables d'une QPSK et les comparer aux valeurs théoriques. Les estimateurs de l'espérance et de la variance nous sont donnés respectivement par les formules suivantes :

$$\hat{m} = \frac{1}{N} \sum_{j=1}^{N} y_j \qquad\qquad \hat{\sigma}^{\,2} = \frac{1}{N-1} \sum_{j=1}^{N} |y_j - \hat{m}|^2$$

Nous pouvons alors utiliser sous Matlab la fonction "*sum(.)*" pour calculer $\hat{m}$. Quant à $\hat{\sigma}^{\,2}$, ce n'est rien d'autre que la norme au carré du vecteur $\boldsymbol{y}$ divisée par $N-1$. Matlab est un logiciel adapté au calcul matriciel. Ainsi, le produit scalaire de vecteurs $\boldsymbol{y}$ et $\boldsymbol{z}$ à composantes complexes s'écrit simplement sous Matlab[1] : $y * z'$. Notons cela dit qu'il existe sous Matlab des fonctions prédéfinies nommées "*mean(.)*" et "*var(.)*" accomplissant ce travail.

Enfin, pour clore cette partie, nous allons représenter sur un même graphique le module de l'estimateur empirique de l'espérance en fonction du nombre de variables N du processus, ainsi que l'espérance théorique. Nous pouvons également représenter la variance empirique du processus en fonction N et la variance théorique. Que remarque-t-on ?

[1] La transposée conjuguée (transposée hermitienne) du vecteur $\boldsymbol{z}$ s'effectue sous Matlab en ajoutant à $\boldsymbol{z}$ une apostrophe, la transposée non conjuguée (transposée symétrique) en ajoutant un point suivi d'une apostrophe.