**TP1 : TRAVAUX PRATIQUES EN MATLAB**

**LOIS DE PROBABILITES DISCRETES, THEORIE DE L’INFORMATION ET CODAGE ENTROPIQUE**

1. **VARIABLES ALEATOIRES DISCRETES**

Nous rappelons dans cette première partie de ce TP quelques notions de base dans l’étude et l’analyse des variables aléatoires notamment celles qui sont les plus utilisées. Parmi les lois de probabilités discrètes les plus utilisées nous avons la loi binomiale et la loi de Poisson

1. **La loi uniforme**

Une **loi de probabilité discrète uniforme** régissant une variable aléatoire X indique que tous les éléments de cette variable aléatoire X, dont le nombre n est fini à savoir X={xa, xa+1, xa+2, …xb-1, ,xb}, possèdent une probabilité de réalisation identique ou équiprobable



Où E(X) = (a + b)/2 et VAR(X) = (n2 – 1)/12

La loi uniforme est la première loi que l’on simule. Diverses méthodes permettent ensuite de simuler à partir de la loi uniforme, un grand nombre de lois, plus ou moins quelconques. Matlab permet de simuler une loi uniforme en utilisant l’instruction : ***rand***. Elle crée un nombre selon la loi uniforme entre 0 et 1. Nous pouvons aussi utiliser l’intruction ***unifrnd(a,b)*** qui génère un nombre aléatoire à partir de la distribution uniforme continue avec les limites inférieures a et supérieures b.

Remarque : Par contre l’instruction ***randn*** crée un nombre aléatoire selon la loi normale

clearall

a=2;

b=7;

nb=1000;

R = unifrnd(a,b,nb,1);

%figure;

%hist(R)

x = a:0.1:b;

p1=unifpdf(x,a,b);

holdon

figure;

stem(x,p1,'filled');

title('Probabilités de chaque évènement');

xlabel('Nombre d evenement');

ylabel('Probabilité');



1. **La loi binomiale**

Une variable Binomiale est donc une variable aléatoire X correspondant à la somme de n variables de Bernoulli. Notée   X : **B**(n,p), où X = nombre de succès au cours de n épreuves de Bernoulli identiques et indépendantes,

Une **épreuve de Bernoulli** de probabilité *p* est une expérience aléatoire comportant deux issues, le succès (avec une probabilité p) ou l'échec (avec une probabilité 1-p=q).

La fonction de probabilité binomiale (équation ci-dessous) fournit la probabilité que x succès se produiront dans n essais d'une expérience binomiale.



Une expérience binomiale a donc quatre propriétés:

1. Elle consiste en une séquence de n essais identiques ;
2. Deux résultats, succès ou échec, sont possibles pour chaque essai ;
3. La probabilité de succès d'un essai, notée p, ne change pas d'un essai à l’autre ;
4. Les essais sont indépendants.

Espérance mathématique d’une loi binomiale

**E(X) = np**

**V(X) = np(1-p)**

Variance d’une loi binomiale

* En Matlab, on utilise la function : ***B=dbinom(x,p,n)***, pour determiner la densité de probabilité binomiale

B étant alors fonction de densité de probabilité binomiale à chacune des valeurs de x en utilisant le nombre d'essais correspondant en n (taille) et la probabilité de succès pour chaque essai en p.

**Exemple :**

Une famille qui a sept (07) enfant, en supposant que la naissance de chaque enfant garçon ou fille a la même probabilité p=0,5. Nous souhaitons donc trouver la densité de probabilité correspondant à toutes les possibilités d’avoir 0, 1, 2, …., 7 garçons

Nous avons donc n=7, p=0.5 et nous devons déterminer B(7, 0,5) = ?

***n=7;***

***p=0.5;***

***g = 0:7;***

***B=binopdf(g,7,0.5);***

***stem(g,B,'filled');***

***title('Nombre de garçons dans une famille de 7 enfants');***

***xlabel('Nombre de garçons');***

***ylabel('Probabilité');***



**B(7,0.5)= {0.0078, 0.0547, 0.1641, 0.2734, 0.2734, 0.1641, 0.0547, 0.0078}**

* Si nous souhaitons aussi determiner la function de repartition d’une distribution binomial on utilise la function***y = binocdf(x,n,p);***

**Même exemple**

***n=7;***

***p=0.5;***

***g = 0:7;***

***figure;***

***B=binopdf(g,7,0.5);***

***stem(g,B,'filled');***

***title('Nombre de garçons dans une famille de 7 enfants');***

***xlabel('Nombre de garçons');***

***ylabel('Probabilité');***

***figure;***

***F=binocdf(g,n,p);***

***stairs(g,F);***

***title('Nombre de garçons dans une famille de 7 enfants');***

***xlabel('Nombre de garçons');***

***ylabel('Probabilité cumulée');***



**F = {0.0078 0.0625 0.2266 0.5000 0.7734 0.9375 0.9922 1.0000}**

* Pour calculer l’espérance mathématique E(X) et la VAR(X) d’un loi binomial sous matlab on peut utilizer la function suivante : ***[E,V]=binostat(n,p)***

**Exemple précédent**

***[E,V] = binostat(7,0.5)***

**E = 3.5000 V = 1.7500**

Nous pouvons verifier ces résultats avec les formules de l’espérance et de la variance d’une variable aléatoire binomial données ci-dessus.

1. **La loi Poisson**

La distribution de probabilité de Poisson est souvent utilisée comme modèle du nombre d'arrivées dans une installation au cours d'une période donnée.

Dans le cas d’une variable de Poisson, les événements se produisent les uns à la suite des autres, de façon aléatoire dans l’espace ou le tempsλ.

Sa loi de probabilité est donnée par :

****

Quant à son espérance mathématique et sa variance ells sont données respectivement par :

**E(X) =** λ **V(X) =** λ

La distribution de Poisson est souvent utilisée pour modéliser le nombre d'événements aléatoires qui se produisent dans un intervalle de temps fixe ou une région spatiale.

Supposons qu’on va passer cinq minutes (soit 300 secondes) à regarder les voitures qui passent dans une ruelle en sachant qu’elles passent généralement à 6 par minute.Ainsi, le paramètre de débit pour notre période de cinq minutes est lambda = 30 (6 voitures/mn × 5mn).On s'attend donc, en moyenne, à voir passer 30 voitures dans cette ruelle toutes les cinq minutes.

Pour simuler cela comme un processus de Poisson, divisons cet intervalle de temps de cinq minutes en très petits intervalles de 1 seconde.

Étant donné que le taux est, en moyenne, de 5 voitures qui passent par minute, cela équivaut à 0,05 par seconde.Dans notre simulation, chaque seconde aura 0,1 chance de voir une voiture qui passe.

***vv = rand(300,1)<0.1; % "vv" voir une voiture qui passe***

***t=1:300;***

***plot(t,vv,'o');***

***title('Probabilité de passage de voitures Durant 5mn');***

***xlabel('Probabilité de passage d’une voiture');***

***ylabel('temps en secondes');***



rand (300,1) crée un vecteur colonne de 300 nombres aléatoires entre 0 et 1, un pour chaque seconde de l'intervalle.

L'opération booléenne <1 renvoie 1 pour chaque valeur inférieure à 1 et 0 pour le reste.

Voyons maintenant combien de voitures vont passer en cinq minutes :

sum(vv)

**vv = 27**

Pour voir à quoi ressemblent généralement les réalisations du processus de Poisson à travers cet exemple il suffit de répéter les quatre dernières commandes suivantes :

**vv = rand(300,1)<0.1;**

**t=1:300;**

**plot(t,vv,'o');**

**sum(vv)**

Sous Matlab on utilise la function***y = poisspdf(x,lambda)***qui calcule la densité de probabilité de Poisson pour chacune des valeurs de x en utilisant les paramètres de taux en lambda.

Par exemple pour le nombre de voiture qui passe dans une ruelle durant 5mn (lambda = 30)

***x = 0:100;***

***y = poisspdf(x,30);***

***stem(x,y,'filled');***

***title('Probabilité de passage d’un nombre de voitures en 5mn');***

***xlabel('Nombre de voitures');***

***ylabel('Probabilité');***



* Si nous souhaitons aussi determiner la function de repartition d’une distribution de Poisson utilise la function***y = poisscdf(x,lambda)***

**x = 0:100;**

**y = poisspdf(x,30);**

**figure;**

**stem(x,y,'filled');**

**title('Probabilité de passage d’un nombre de voitures en 5mn');**

**xlabel('Nombre de voitures');**

**ylabel('Probabilité');**

**Y= poisscdf(x,30);**

**figure;**

**stairs(x,Y);**

**title('Probabilité de passage d’un nombre de voitures en 5mn');**

**xlabel('Nombre de voitures');**

**ylabel('Probabilité cumulée');**

****

Pour calculer sous Matlab l’espérance mathématique et la variance d’une distribution de Poisson on utilise la fonction : ***[E,V] = poisstat(lambda)*** où E est l’espérance et V la variance. On rappelle que dans le cas d’une variable aléatoire de Poisson l’espérance et la variance sont toutes les deux égales à lambda

***[E,V] = poisstat(30)***

**E= 30 V = 30**

**Résumés des commandes sous Matlab**

* **Uniforme :*rand, unifrnd(a,b)***
* **Binomial*: binopdf, binocdf, binostat***
* **Poisson *: poisspdf, poisscdf, poisstat***

1. **THEORIE DE L’INFORMATION**

Selon les concepts de la théorie de l’information le calcul de la l’information associée à chaque symbole d’un alphabet discret (composé d’un nombre fini de symbole) M={m1, m2,........, mK}, de longueur lk fixe ou variable, d’une source est régit par une probabilité d’apparition {P(m1), P(m2),........, P(mk)}est d’une importance cruciale.



Nous avons également besoin de calculer l’entropie de cet alphabet H(X) comme étant sa quantité moyenne d’information :



S’il s’agit maintenant de deux alphabets discrets X et Y, comme par exemple les données source envoyées par émetteur et les données reçues au niveau du récepteur, nous pouvons alors calculer :

* I(X) et I(Y) : Information propre de X et de Y respectivement



* H(X,Y) : Information conjointe ou entropie mutuelle



Où f(x,y) est la densité de probabilité conjointe

* H(X/Y) : Information partielle ou conditionnelle de X sachant Y. Elle représente l’ambigüité ou l’incertitude qui reste sur X pour Y connu.



Où P(Y/X) est la probabilité conditionnelle de Y sachant X

* I(X,Y) : Information mutuelle ou quantité d’information de X réellement apportée par Y
* I(X/Y) et I(Y/X) : Informations conditionnelles
* Exemple 1 : Calcul de l’entropie d’une variable aléatoire uniforme

*Clear all*

*a=2;*

*b=7;*

*nb=1000;*

*R = unifrnd(a,b,nb,1);*

*figure;*

*hist(R)*

*x = a:0.1:b;*

*p1=unifpdf(x,a,b);*

*holdon*

*figure;*

*stem(x,p1,'filled');*

*title('Probabilités de chaque évènement');*

*xlabel('Nombre d evenement');*

*ylabel('Probabilité');*

*e = - sum( p1 .\* log2( max(p1,1e-20) ) );*

*disp(strcat(['Entropy=' num2str(e)]) );*

* Exemple 2 : Calcul de l’entropie d’une image en utilisant directement l’instruction ***entropy***.

*I = imread('circuit.tif');*

*Imshow(I)*

*E = entropy(I)*