**Contenus mathématiques :**

• Fréquences

• Simulation d’échantillonnage

• Intervalle de fluctuation

**Contenus en Python :**

• Instructions conditionnelles, Boucles

• Utilisation de listes

• Utilisation du module graphique, paramétrages, fenêtres multiples

**1/ Réaliser une simulation :**

Dans son sac de 100 billes, Bart affirme que 25 sont jaunes et 75 sont rouges.

Lisa souhaite vérifier les dires de Bart.

1. Proposer une méthode à Lisa pour vérifier ces proportions.

Tirer au hasard et avec remise une bille du sac un grand nombre de fois et calculer la fréquence d’apparition de billes jaunes.

1. Protocole de simulation :

Tirer une bille parmi 100 dans lesquelles 25 sont jaunes et 75 sont rouges revient à tirer un numéro au hasard parmi 100. On peut, par exemple, attribuer les numéros 1 à 25 aux billes

jaunes ce qui correspond bien à une proportion de 25%.

Le programme ci-dessous (fichier « **Simulation - Intervalle de fluctuation.py** ») permet de

simuler 100 fois cent tirages d’une bille parmi 100 dans lesquelles 25 seraient jaunes.

.

from lycee import \*

##Création de la liste des abscisses

abscisses=[]

for i in range(100):

abscisses.append(i) #abscisses est la liste [0,1,2,...,99]

##Création de la liste des ordonnées

ordonnees=[]

for i in range(100):

nb\_jaune=0

for j in range(100):

alea=randint(1,100)

if alea<=25:

nb\_jaune=nb\_jaune+1

ordonnees.append(nb\_jaune/100) #liste de la fréquence de boules jaunes

##Partie graphique

repere.clf() #Efface le graphique

repere.axis([0.05,100,0,0.45]) #Bornes de la fenêtre

repere.grid(True) #Affiche la grille

repere.xlabel("Simulations",color="blue") #Affiche la légende en abs

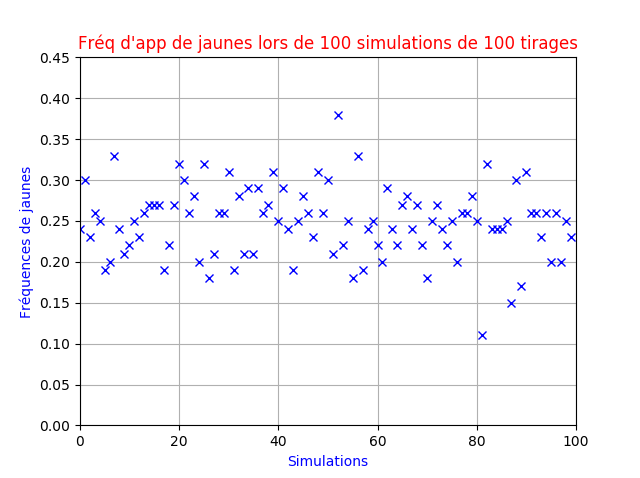
repere.ylabel("Fréquences de jaunes",color="blue")

repere.title("Fréq d'app de jaunes lors de 100 simulations de 100 tirages", color="red") #Affiche le titre

repere.plot(abscisses,ordonnees,"bx") #Trace le nuage de points

repere.show() #Affiche le graphique

Les résultats de 100 simulations sont représentés sur le graphique suivant :



Autour de quelle fréquence théorique semblent s’accumuler les fréquences de billes jaunes ?

**2/ Découvrir un intervalle de fluctuation :**

1. A partir du graphique précédent, calculer la proportion de points situés dans la bande

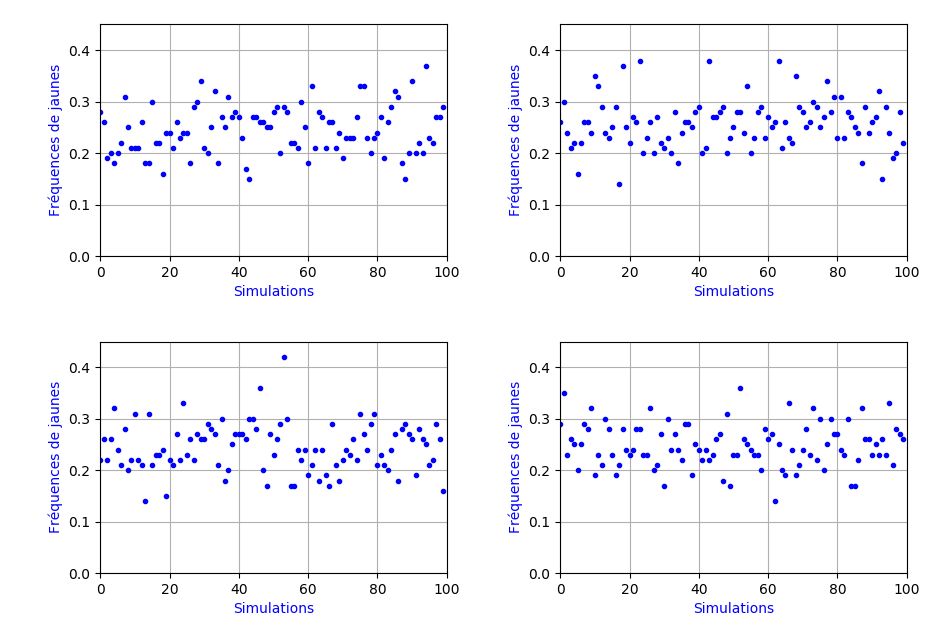
horizontale délimitée par les ordonnées 0,15 et 0,35 (incluses).

2 fréquences sont hors de la bande 0,15-0,35 soit

98% des fréquences appartiennent à l’intervalle 0,15-0,35

1. Dans les simulations précédentes, on a au moins 95% des fréquences de billes jaunes qui appartiennent à l’intervalle [0,15 ; 0,35].

Vérifier si cela est toujours le cas dans les simulations suivantes :



97%

99%

95%

98%

1. • Simplifier les bornes de l’intervalle . Que retrouve-t-on ?

• Que représentent 0,25 et 100 dans l’intervalle précédent ?

[0,15 ; 0,35]

On retrouve les bornes de la bande

0,25 est la fréquence de billes jaunes, 100 le nombre de billes

On dit que l’intervalle [0,15 ; 0,35] est **un** **intervalle de fluctuation** au seuil de 95% des fréquences de billes jaunes.

**3/ Amélioration du programme :**

Compléter le programme pour qu’il affiche lors des simulations la proportion de fréquences de billes jaunes comprises entre 0,15 et 0,35.

##Calcul des fréquences dans [0,15;0,35]

n=0

for i in range(len(ordonnees)): #i allant de 1 à la long de la liste ordonnees

if ordonnees[i]<0.15 or ordonnees[i]>0.35:

n=n+1

print(100-n,"% de fréquences dans [0,15;0,35]")

Programme avec l’affichage multifenêtre :

from lycee import \*

def simulation():

##Création de la liste des abscisses

abscisses=[]

for i in range(100):

abscisses.append(i) #abscisses est la liste [0,1,2,...,99]

##Création de la liste des ordonnées

ordonnees=[]

for i in range(100):

nb\_jaune=0

for j in range(100):

alea=randint(1,100)

if alea<=25:

nb\_jaune=nb\_jaune+1

ordonnees.append(nb\_jaune/100) #liste de la fréquence de boules jaunes

##Partie graphique

repere.axis([0.05,100,0,0.45]) #Bornes de la fenêtre

repere.grid(True) #Affiche la grille

repere.xlabel("Simulations",color="blue") #Affiche la légende en abscisses

repere.ylabel("Fréquences de jaunes",color="blue")

repere.plot(abscisses,ordonnees,"b.") #Trace le nuage de points

repere.subplot(221) #Positionne le 1er tracé dans la partie 1 du cadre 2x2

simulation()

repere.subplot(222)

simulation()

repere.subplot(223)

simulation()

repere.subplot(224)

simulation()

repere.show() #Affiche le graphique