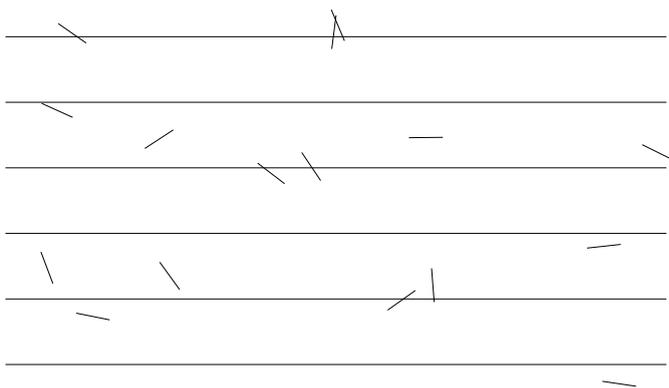


Applications des mathématiques

Simulation



Version pour *Mathematica*

Edition 2017

Marcel Délèze

<https://www.deleze.name/marcel/sec2/applmaths/csud/index.html>

§ 1 Nombres pseudo-aléatoires à distribution spécifiée

§ 1.1 Introduction

Pour résoudre un problème dans lequel intervient le hasard, il existe trois familles de méthodes:

Méthode théorique

Le calcul des probabilités sera étudié plus tard, en 4-ème année, dans le cadre du cours de mathématiques. Nous nous contentons ici d'en faire une simple mention.

Lorsque le problème est assez simple, il peut être résolu d'une manière exacte et complète par le calcul des probabilités. Mais, pour des problèmes plus complexes, il est généralement difficile, voire même impossible, de donner une solution explicite. Il est utile de connaître d'autres approches.

Méthode expérimentale

Il est parfois possible d'organiser une série d'expériences. Par exemple, lancer un dé et noter les résultats obtenus est facile. Mais, pour obtenir une certaine précision, il faut faire un grand nombre d'expériences ce qui est coûteux.

Simulation (ou méthode de Monte Carlo)

Il existe des algorithmes capables de produire des "nombres pseudo-aléatoires" c'est-à-dire des nombres qu'on peut utiliser à la place des nombres donnés par le dé. L'ordinateur étant capable d'effectuer un grand nombre d'expériences en peu de temps, on peut obtenir des résultats plus précis et à meilleur compte qu'avec la méthode d'expérimentation.

Alors que les nombres aléatoires sont engendrés par un phénomène aléatoire (par exemple par tirage à la roulette), les nombres pseudo-aléatoires sont calculés au moyen d'un algorithme; ils offrent néanmoins les mêmes caractéristiques que les nombres aléatoires et peuvent avantageusement les remplacer dans les calculs.

On appelle "méthode de Monte-Carlo" toute méthode de calcul utilisant des nombres pseudo-aléatoires. Cette dénomination se réfère à la roulette du célèbre casino de Monte-Carlo. Parmi les applications des méthodes de Monte-Carlo, citons

- * la programmation de jeux de hasard (dés, cartes, lotto, roulette, ...);
- * la simulation de phénomènes aléatoires (
 - cheminements d'un neutron au sein d'une matière fissile,
 - fluctuations de charge d'un central téléphonique,
 - longueur de la file d'attente à un guichet, ...).

§ 1.2 Nombres pseudo-aléatoires à distribution discrète

Distribution de Bernoulli

$n = 40$;

? BernoulliDistribution

BernoulliDistribution[p] represents a Bernoulli distribution with probability parameter p . >>

Par exemple, une expérience aléatoire consistant à lancer un dé, le nombre de six est

```
six = BernoulliDistribution[ $\frac{1}{6}$ ];
```

[distribution de Bernoulli]

RandomInteger[six, n]

[entier aléatoire]

```
{1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0}
```

Distribution binomiale

? BinomialDistribution

BinomialDistribution[n, p] represents a binomial distribution with n trials and success probability p . >>

Par exemple, une expérience aléatoire consistant à lancer 10 dés, le nombre de six est

```
six10 = BinomialDistribution[10,  $\frac{1}{6}$ ];
```

[distribution binomiale]

RandomInteger[six10, n]

[entier aléatoire]

```
{6, 2, 0, 4, 1, 2, 2, 2, 0, 3, 3, 3, 0, 2, 0, 3, 3, 1,
 0, 1, 1, 1, 1, 1, 1, 0, 2, 2, 1, 2, 3, 3, 0, 2, 1, 0, 2, 0, 2, 1}
```

Distribution discrète uniforme

? DiscreteUniformDistribution

DiscreteUniformDistribution[$\{i_{min}, i_{max}\}$] represents a discrete uniform distribution over the integers from i_{min} to i_{max} .

DiscreteUniformDistribution[$\{\{i_{min}, i_{max}\}, \{j_{min}, j_{max}\}, \dots\}$] represents a multivariate

discrete uniform distribution over integers within the box $\{\{i_{min}, i_{max}\}, \{j_{min}, j_{max}\}, \dots\}$. >>

Par exemple, une expérience aléatoire consistant à lancer 1 dé, le nombre de points est

```
de = DiscreteUniformDistribution[{1, 6}];
```

[distribution uniforme discrète]

RandomInteger[de, n]

[entier aléatoire]

```
{4, 2, 5, 1, 4, 4, 2, 3, 4, 3, 3, 1, 5, 1, 6, 3, 5, 6,
 2, 3, 5, 6, 1, 5, 2, 5, 5, 4, 5, 1, 3, 2, 5, 6, 1, 6, 5, 3, 1, 1}
```

Distribution géométrique

? GeometricDistribution

GeometricDistribution[p] represents a geometric distribution with probability parameter p . >>

Par exemple, une expérience aléatoire consistant à lancer un dé jusqu'à ce que l'on obtienne un six, le nombre de lancers avant d'obtenir le premier six est

```
jets = GeometricDistribution[ $\frac{1}{6}$ ];
```

[distribution géométrique]

```
RandomInteger[jets, n]
```

[entier aléatoire]

```
{3, 3, 7, 3, 1, 5, 0, 6, 2, 2, 5, 3, 5, 2, 5, 6, 2, 12,
 4, 10, 1, 5, 3, 12, 4, 1, 2, 3, 2, 0, 4, 7, 0, 8, 0, 2, 0, 1, 1, 1}
```

Si on désire que le dernier jet, qui est un six, soit compté,

```
1 + RandomInteger[jets, n]
```

[entier aléatoire]

```
{9, 5, 6, 7, 12, 22, 16, 7, 1, 10, 5, 5, 6, 4, 3, 2, 4, 11,
 7, 7, 1, 4, 7, 6, 5, 2, 12, 1, 11, 1, 6, 11, 5, 2, 1, 2, 2, 5, 3, 3}
```

§ 1.3 Nombres pseudo-aléatoires à distribution continue

```
n = 40;
```

Distribution uniforme

? UniformDistribution

UniformDistribution[$\{min, max\}$] represents a continuous uniform statistical distribution giving values between min and max .

UniformDistribution[] represents a uniform distribution giving values between 0 and 1.

UniformDistribution[$\{\{x_{min}, x_{max}\}, \{y_{min}, y_{max}\}, \dots\}$]

represents a multivariate uniform distribution over the region $\{\{x_{min}, x_{max}\}, \{y_{min}, y_{max}\}, \dots\}$.

UniformDistribution[n] represents a multivariate uniform distribution over the standard n dimensional unit hypercube. >>

Par exemple, pour créer un échantillon de n mesures principales d'angles, sous l'hypothèse que la distribution est uniforme:

```
angle = UniformDistribution[{- $\pi$ ,  $\pi$ ];
```

[distribution uniforme]

```
RandomReal[angle, n]
```

[nombre réel aléatoire]

```
{-0.683042, 2.09415, -2.53417, -1.5028, 3.08988, -2.42516, -1.79832, -1.31385,
 2.11127, -2.49978, -0.482146, 2.95729, -2.60082, -2.09053, -0.290656, -0.348077,
 0.788984, 0.115755, -0.919931, 0.752596, 2.69563, 2.96901, -2.89588, -2.9345,
 -0.501021, 0.394878, 1.82333, -3.1182, -2.96207, 2.43677, -2.1586, -0.669241,
 -1.22721, 1.71811, -2.91862, 3.05009, 1.2726, 2.78849, -1.03727, 1.56252}
```

Distribution normale

? NormalDistribution

NormalDistribution[μ, σ] represents a normal (Gaussian) distribution with mean μ and standard deviation σ .

NormalDistribution[] represents a normal distribution with zero mean and unit standard deviation. >>

Par exemple, pour créer un échantillon des tailles de n personnes, de taille moyenne 1.75 m, d'écart-type 0.2 m, sous l'hypothèse que la distribution des tailles est normale:

```
taille = NormalDistribution[1.75, 0.2];
```

[distribution normale]

```
RandomReal[taille, n]
```

[nombre réel aléatoire]

```
{1.93599, 1.74758, 1.83664, 1.50581, 1.90537, 1.69439, 1.82888, 1.88104, 1.73929, 1.61592,
1.83682, 1.88412, 2.14447, 1.86877, 1.74841, 1.96817, 1.57962, 1.67659, 1.66649, 1.53086,
1.91589, 1.65331, 1.75734, 1.59718, 1.98491, 1.79224, 1.66326, 1.56929, 1.49002, 2.05149,
1.4762, 1.93154, 2.16702, 1.88866, 1.30549, 1.7184, 1.9197, 1.79326, 1.7118, 1.51354}
```

§ 1.4 Estimation de l'erreur

Pour de grandes valeurs de n , la moyenne empirique de n tirages indépendants a une distribution approximativement normale. En notant m la moyenne empirique, s l'écart-type empirique et

$\hat{\sigma}_M = \frac{s}{\sqrt{n-1}}$ l'estimation de l'écart-type de la moyenne M , nous aurons approximativement

la probabilité que l'intervalle $[m - \hat{\sigma}_M, m + \hat{\sigma}_M]$ contienne μ est de 68.3 % environ;

la probabilité que l'intervalle $[m - 2 \hat{\sigma}_M, m + 2 \hat{\sigma}_M]$ contienne μ est de 95.5 % environ;

la probabilité que l'intervalle $[m - 3 \hat{\sigma}_M, m + 3 \hat{\sigma}_M]$ contienne μ est de 99.7 % environ.

Justification:

Voir *Probabilités II*, § 2.6:

https://www.deleze.name/marcel/sec2/applmaths/csud/statistique_2/2-stat_II.pdf

§ 1.5 Gestion du temps de calcul

Recherche de l'efficacité

Pour générer des tirages aléatoires, utilisez si possible les distributions de *Mathematica* et la commande **RandomInteger[distr, n]** ou **RandomReal[distr, n]**.

Si la variable est à valeurs $\{0, 1\}$, calculez l'écart-type au moyen de la formule $s = \sqrt{m(1-m)}$.

Si la variable est discrète et ne comporte qu'un nombre fini de modalités, regroupez les données puis calculez la moyenne et l'écart-type sur les données groupées.

Contrôle de la durée

La précision s'améliore avec la taille de l'échantillon. Mais la durée des calculs ne peut pas dépasser la limite du supportable, définie par l'utilisateur. Nous adopterons la règle suivante: *la taille de l'échantillon est choisie de telle manière que la durée de la simulation est de 1 minute environ.*

Pour déterminer le nombre d'expériences, nous effectuons un calcul préalable qui consiste à chronométrer la durée d'un "certain nombre" d'expériences. A partir de là, on calcule combien d'expériences on peut réaliser en 1 minute.

Exercice 1 - 1

Dans le but d'effectuer la simulation suivante : "Tirer 20 nombres réels pseudo-aléatoires uniformément répartis entre 0 et 1 et afficher ceux qui sont compris entre 0.4 et 0.6", un étudiant a écrit le programme suivant :

```

echant = Table[RandomReal[{0, 1}], {20}]
           |table |nombre réel aléatoire
{0.911, 0.0633708, 0.557836, 0.979648, 0.90393, 0.479523,
 0.53959, 0.935894, 0.230864, 0.700587, 0.564174, 0.517494, 0.0524311,
 0.522905, 0.504709, 0.837292, 0.343023, 0.249707, 0.867703, 0.528961}

critQ[x_] := 0.4 <= RandomReal[{0, 1}] < 0.6
           |nombre réel aléatoire

Select[echant, critQ]
|sélectionne
{0.979648, 0.230864, 0.528961}

```

Pourquoi le programme ne donne-t-il pas le résultat escompté ?
 Comment le corriger pour qu'il accomplisse la tâche désirée ?

Exercice 1 - 2 (facultatif)

Ecrivez un programme qui distribue aléatoirement les 36 cartes du jeu de jass à quatre joueurs.

Indications

a) Le début du programme qui construit la liste ordonnée des cartes vous est donné:

<https://www.deleze.name/marcel/sec2/applmaths/csud/simulation/annexes/1-2-donnees-exercice.nb>

b) Pour battre les cartes, on peut utiliser l'idée suivante:

RandomSample[Range[k]] donne une permutation aléatoire de la liste
 $\{1, 2, 3, \dots, k\}$:

? RandomSample

RandomSample[{ e_1, e_2, \dots }, n] gives a pseudorandom sample of n of the e_i .
 RandomSample[{ w_1, w_2, \dots } → { e_1, e_2, \dots }, n] gives a pseudorandom sample of n of the e_i chosen using weights w_i .
 RandomSample[{ e_1, e_2, \dots }] gives a pseudorandom permutation of the e_i . >>

Par exemple,

```

RandomSample[Range[10]]
|échantillon aléat· |plage
{4, 6, 8, 10, 2, 5, 9, 7, 3, 1}

```

Après avoir numéroté les cartes de 1 à 36, la permutation **RandomSample[Range[36]]** représente un jeu brassé.

c) Pour le joueur numéro 1, extraire du jeu les cartes numéros 1, 2, 3, 13, 14, 15, 25, 26, 27;
 pour le joueur numéro 2, extraire du jeu les cartes numéros 4, 5, 6, 16, 17, 18, 28, 29, 30;
 etc.

§ 2 Exemples de simulations

§ 2.1 Deux six consécutifs

Combien de fois faut-il lancer le dé, en moyenne, jusqu'à ce que l'on obtienne 2 six consécutifs?

Comme question préalable, lançons d'abord le dé jusqu'à l'apparition du premier six.

? **GeometricDistribution**

GeometricDistribution[p] represents a geometric distribution with probability parameter p. >>

```
n6 = GeometricDistribution[ $\frac{1}{6}$ ];
```

|distribution géométrique

Le nombre de lancers d'un dé pour obtenir un six est donné par

```
1 + RandomInteger[n6, 20]
```

|entier aléatoire

```
{4, 22, 2, 3, 1, 2, 11, 3, 3, 2, 27, 3, 15, 1, 4, 12, 9, 3, 2, 1}
```

Si, dans la liste précédente, on veut déterminer le nombre de lancers pour obtenir deux six consécutifs (pour la première fois), on peut faire comme suit:

- * il faut sommer le nombre de coups : $s = 4 + 22 + 2 + \dots$
- * il faut interrompre la sommation au premier 1 rencontré, à partir du deuxième rang :
 - le premier terme doit toujours être pris, même si c'est 1;
 - le dernier terme est 1 et fait partie de la somme;
- * le résultat est $s = 4 + 22 + 2 + 3 + 1$.

Le générateur de nombres pseudo-aléatoires qui simule "le nombre de fois qu'il faut lancer le dé pour obtenir deux six consécutifs" est

```
deuxSix[] := Module[{s, t},
  |module
  s = 1 + RandomInteger[n6];
  |entier aléatoire
  t = 1 + RandomInteger[n6];
  |entier aléatoire
  s += t;
  While[t > 1, t = 1 + RandomInteger[n6]; s += t];
  |tant que |entier aléatoire
  Return[s]
  |reviens
```

Mesurons d'abord la durée d'une "petite" simulation:

```
duree = Timing[
  |chronomètre
  n = 5000;
  x = Table[deuxSix[], {n}]
  |table
] [[1]]
```

1.02961

Ce résultat est un nombre de secondes. Accordons maintenant une minute à l'exécution de la

simulation principale:

$$n = n * \text{Round}\left[\frac{60}{\text{duree}}\right]$$

290 000

$$x = \text{Table}[\text{deuxSix}[], \{n\}];$$

$$m = \text{N}[\text{Mean}[x]]$$

41.974

$$s = \text{StandardDeviation}[x];$$

$$sm = \text{N}\left[\frac{s}{\sqrt{n}}\right]$$

0.0750823

Interprétation : avec une probabilité de 95 %, le nombre cherché μ est situé dans l'intervalle

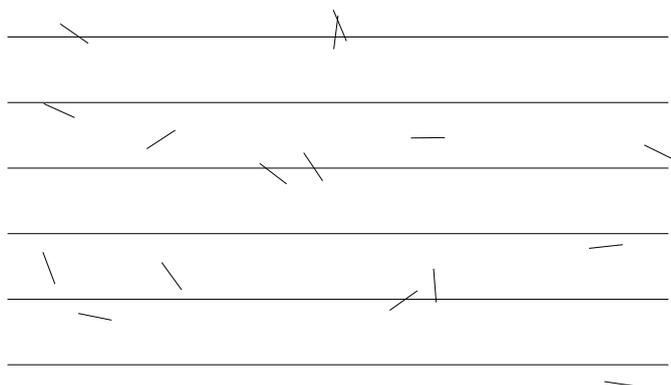
$$\{m - 2 sm, m + 2 sm\}$$

$$\{41.8239, 42.1242\}$$

Ce problème peut être résolu par le calcul des probabilités. Le résultat exact est $\mu = 42$.

§ 2.2 Aiguille de Buffon

On lance une aiguille sur un plancher. La longueur de l'aiguille est 1 et la largeur d'une planche est 2. Quelle est la probabilité que l'aiguille touche 2 planches ?



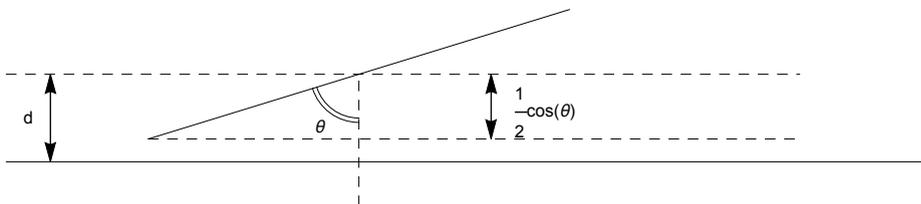
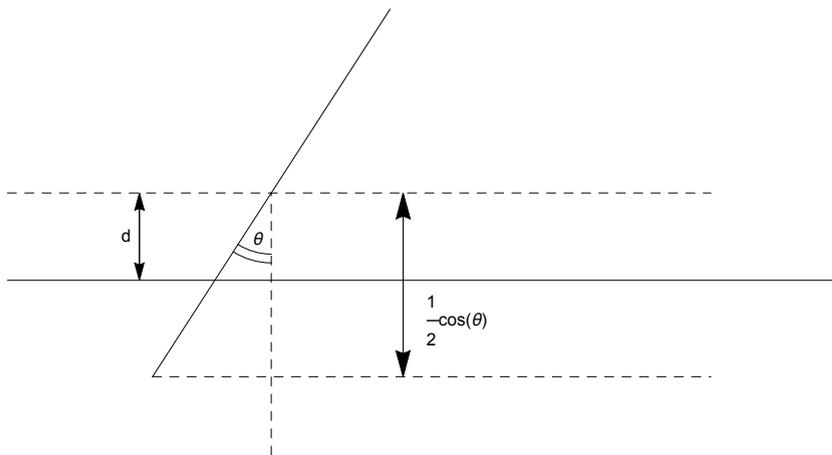
Critère de réussite

Dans un premier temps, organisons une seule expérience.

Notons

d = distance entre le centre de gravité de l'aiguille et la droite la plus proche;

θ = angle non orienté entre l'aiguille et la perpendiculaire aux droites.



On voit sur les deux graphiques précédents que

l'aiguille coupe un interstice entre deux planches si et seulement si $\frac{1}{2} \cos(\theta) > d$

```
dist = UniformDistribution[{0, 1}];
      |distribution uniforme
```

```
angle = UniformDistribution[{0,  $\frac{\pi}{2}$ }]
      |distribution uniforme
```

Mesurons d'abord la durée d'une "petite" simulation:

```
duree = Timing[
  |chronomètre
  n = 10000;
  d = RandomReal[dist, n];
  |nombre réel aléatoire
  theta = RandomReal[angle, n];
  |nombre réel aléatoire
  succes = Count[Table[0.5 Cos[theta[[i]]] > d[[i]], {i, 1, n}], True]
  |compte |table |cosinus |vrai
] [[1]]
0.0312002
```

Accordons maintenant une minute à l'exécution de la simulation principale:

```

n = n * Round[ $\frac{60}{\text{duree}}$ ]
19 230 000

d = RandomReal[dist, n];
 $\theta$  = RandomReal[angle, n];
succes = Count[Table[0.5 Cos[ $\theta[[i]]$ ] > d[[i]], {i, 1, n}], True]
m = N[ $\frac{\text{succes}}{n}$ ]
s =  $\sqrt{m(1-m)}$ ;
sm = N[ $\frac{s}{\sqrt{n}}$ ]

```

0.318275

0.000106222

Interprétation : avec une probabilité de 95 %, le nombre cherché est situé dans l'intervalle

$\{m - 2 sm, m + 2 sm\}$
 $\{0.318063, 0.318488\}$

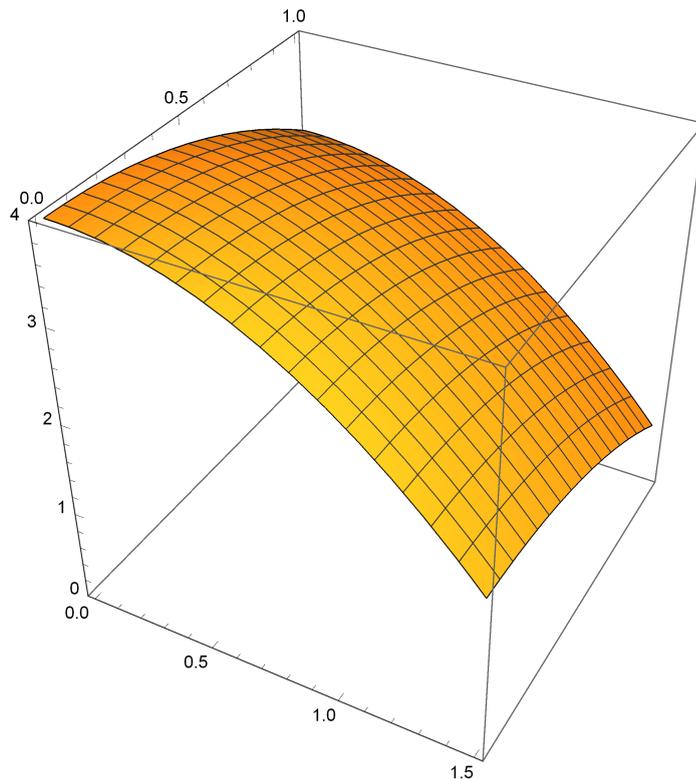
Ce problème peut être résolu par le calcul des probabilités. Le résultat exact est $\mu = \frac{1}{\pi} \approx 0.31831$.

§ 2.3 Calcul d'un volume par la méthode de Monte-Carlo

Considérons le corps $C = \{(x, y, z) \mid 0 < x < \frac{3}{2}, 0 < y < 1 \text{ et } 0 < z < 4 - x^2 - y^2\}$.

Quel est son volume ?

`Plot3D[4 - x2 - y2, {x, 0, 1.5}, {y, 0, 1}, PlotRange -> {0, 4}, BoxRatios -> {1, 1, 1}]`
 [tracé de surfaces] [zone de tracé] [rapports de boîte]



Une expérience élémentaire consiste à placer un point (x, y, z) au hasard dans le parallélépipède avec

$$\begin{aligned} 0 &\leq x < 1 \\ 0 &\leq y < 1.5 \\ 0 &\leq z < 4 \end{aligned}$$

L'expérience est réussie si le point est situé au-dessous de la surface $z = f(x, y)$ c'est-à-dire si

$$z < 4 - x^2 - y^2$$

La fréquence de réussite m est égale au rapport entre le volume "V" cherché et le volume "boite" du parallélépipède:

$$m = \frac{V}{\text{boite}} \iff V = m * \text{boite}$$

```
absc = UniformDistribution[{0, 1.5}];
```

[distribution uniforme]

```
ordon = UniformDistribution[{0, 1}];
```

[distribution uniforme]

```
cote = UniformDistribution[{0, 4}];
```

[distribution uniforme]

Mesurons d'abord la durée d'une "petite" simulation:

```

duree = Timing[
  [chronomètre
    n = 10000;
    x = RandomReal[absc, n];
      [nombre réel aléatoire
    y = RandomReal[ordon, n];
      [nombre réel aléatoire
    z = RandomReal[cote, n];
      [nombre réel aléatoire
    succes = Count[Table[z[[i]] < 4 - x[[i]]2 - y[[i]]2, {i, 1, n}], True]
      [compte [table] [vrai]
  ] [[1]]
0.0624004

```

Accordons maintenant une minute à l'exécution de la simulation principale:

```

n = n * Round[ $\frac{60}{duree}$ ]
  [arrondis
9 620 000

x = RandomReal[absc, n];
  [nombre réel aléatoire
y = RandomReal[ordon, n];
  [nombre réel aléatoire
z = RandomReal[cote, n];
  [nombre réel aléatoire
succes = Count[Table[z[[i]] < 4 - x[[i]]2 - y[[i]]2, {i, 1, n}], True]
  [compte [table] [vrai]
7 016 622

m = N[ $\frac{succes}{n}$ ]
  [valeur numérique
0.729379

s =  $\sqrt{m(1-m)}$ ;
sm = N[ $\frac{s}{\sqrt{n}}$ ]
  [valeur numérique
0.000143242

boite = 1 * 1.5 * 4;
vol = m * boite
4.37627

incert = sm * boite
0.000859451

```

Interprétation : avec une probabilité de 95 %, le nombre cherché est situé dans l'intervalle

{vol - 2 incert, vol + 2 incert}

{4.37455, 4.37799}

Ce problème peut être résolu par calcul intégral. Le résultat exact est 4.375.

Calcul des probabilités exactes

Le calcul des probabilités exactes des exemples des paragraphes 2.1, 2.2 et 2.3 sort des objectifs de ce cours. Les intéressés (en particulier le professeur) peuvent trouver ces calculs dans les annexes:

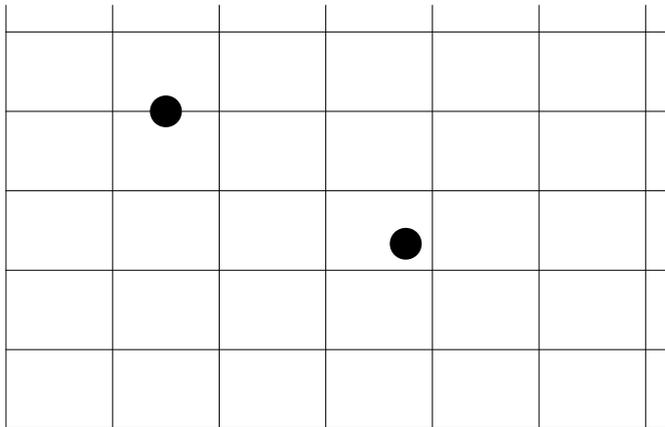
<https://www.deleze.name/marcel/sec2/applmaths/csud/simulation/annexes/2s-prob-exactes.pdf>

Exercices du § 2

Exercice 2 - 1

Perpendiculairement à un vaste grillage, on lance une balle au hasard.

La balle a un rayon de 0.6 et les mailles du filet sont rectangulaires de 4 sur 3.



Quelle est la probabilité que la balle traverse le grillage sans le toucher?

- Calculez la probabilité exacte μ de cet événement.
Indication: utilisez une méthode géométrique).
- Estimez la fréquence de succès m au moyen d'une simulation.

Exercice 2 - 2

A l'aide d'une méthode de Monte-Carlo, estimez le volume d'une boule de rayon 1, centrée à l'origine:

$$B = \{(x, y, z) \mid x^2 + y^2 + z^2 \leq 1\}.$$

Commentaire:

$x^2 + y^2 + z^2$ représente le carré de la distance entre le point $P(x, y, z)$ et l'origine $O(0, 0, 0)$:

$$OP^2 = x^2 + y^2 + z^2.$$

B est donc l'ensemble des points P tels que $OP^2 \leq 1$.

Exercice 2 - 3 (Permutations sans point fixe)

k hommes, avec leurs k épouses respectives, participent à une soirée dansante. L'animateur de la

soirée forme des couples au hasard.

Estimez la probabilité qu'aucun homme ne danse avec son épouse.

Indications:

- 1° Numérotons les couples et représentons la situation par une liste $v = \{3, 47, \dots, 14\}$ qui signifie, dans le cas particulier,
 - l'homme numéro 1 danse avec la femme numéro 3,
 - l'homme numéro 2 danse avec la femme numéro 47,
 - ...
 - l'homme numéro k danse avec la femme numéro 14.
- 2° Initialement, la liste est $v_0 = \{1, 2, \dots, k\}$. Effectuez une permutation au hasard selon la méthode décrite dans l'exercice 1-2 pour battre les cartes.
- 3° Une épreuve est ratée s'il existe un numéro j tel que $v[[j]] = j$ c'est-à-dire s'il existe une coïncidence entre les listes v et v_0 . En *Mathematica*, le test peut s'écrire


```
If[MemberQ[v - v0, 0], 0, 1]
```
- 4° Il faut organiser une série de n épreuves.
Pour de grandes valeurs de k , l'espérance mathématique μ tend vers $\frac{1}{e} \approx 0.368$

Exercice 2 - 4

Combien de fois faut-il lancer un dé pour qu'apparaisse l'événement "tirer un six" ?

Exercice 2 - 5

Combien de fois faut-il lancer trois dés pour que la somme des 3 dés dépasse 15 ?

Indications: structurez votre programme en sous-programmes, chaque sous-programme correspondant à une fonction:

- * une expérience consiste à lancer 3 dés et à tester si leur somme dépasse 15;
- * la variable aléatoire est le nombre d'expériences nécessaires pour obtenir une expérience réussie;
- * une simulation consiste à évaluer un grand nombre de fois la variable aléatoire.

Exercice 2 - 6

On lance le dé jusqu'à ce que l'on obtienne deux issues consécutives égales puis on considère la somme des points obtenus.

Par exemple : $1 + 5 + 3 + 5 + 4 + 4 = 22$.

Estimez la moyenne et l'écart-type des valeurs ainsi obtenues.

Faites un diagramme en bâtons (ou un histogramme) des résultats obtenus.

Lien vers les corrigés des exercices

<https://www.deleze.name/marcel/sec2/applmaths/csud/corriges/simulation/simul-cor.pdf>