

Niveau 2 - Méthodes Numériques
Cours 4
Simulation de processus stochastiques
Mouvement brownien

Jean-François Berger-Lefebure

10 Octobre 2024

Contents

1 Rappel mouvement brownien	3
2 Simulation d'un mouvement brownien : méthode simplifiée	3
3 Exercice 10 : Simulation d'un mouvement brownien	4
3.1 Notions abordées	4
3.1.1 Stockage en mémoire par colonnes	4
3.2 Question 1 & 3	5
3.2.1 Explication du code	5
3.3 Question 2	6
3.3.1 Explication du code	6
3.3.2 Explication - Maximum d'un mouvement brownien	7
3.4 Question 4	8
3.5 Question 5	8
3.5.1 Estimation avec la première variable de contrôle : W_T	8
3.5.2 Corrélation entre M et W_T	9
3.5.3 Estimation avec la deuxième variable de contrôle : $\int_0^T W_s ds$	10
3.5.4 Notions clés sur les variables de contrôle et leur efficacité	10
4 But de la réduction de variance dans la simulation d'un mouvement brownien	12
4.1 Pourquoi réduire la variance ?	12
4.2 Exemple d'application	12
5 QCM	13
5.1 Questions introduction	13
5.2 Questions essentielles	14

1 Rappel mouvement brownien

- **Objectif de la simulation :** Le mouvement brownien est un processus stochastique continu défini sur $[0, T]$. Cependant, il est impossible de simuler un tel processus de manière continue à cause de l'infinité des points dans un intervalle continu. On discrétise donc l'intervalle $[0, T]$ en une grille de temps $\{t_j\}_{0 \leq j \leq m}$, où $t_0 = 0$ et $t_m = T$. Cela donne $m + 1$ points.
- **Définition d'un mouvement brownien :** Un processus W_t est un mouvement brownien si :
 - $W_0 = 0$ presque sûrement.
 - $t \mapsto W_t$ est continu presque sûrement.
 - $W_t - W_s$ est indépendant de la filtration \mathcal{F}_s pour $t \geq s$. Cela signifie que l'accroissement est indépendant du passé, mais W_t n'est pas indépendant de W_s .
 - $W_t - W_s \sim \mathcal{N}(0, t - s)$, c'est-à-dire que l'accroissement suit une loi normale centrée avec une variance égale à l'intervalle de temps $(t - s)$.
- **Méthode de simulation :** On simule les accroissements indépendants $\Delta W_{t_j} \sim \mathcal{N}(0, t_{j+1} - t_j)$, puis on les ajoute successivement :

$$W_{t_{j+1}} = W_{t_j} + \Delta W_{t_j}.$$

2 Simulation d'un mouvement brownien : méthode simplifiée

- **Accroissements simulés :** Pour simuler un mouvement brownien $(W_t)_{t \in [0, T]}$, on utilise une grille discrète de points $\{t_j\}_{0 \leq j \leq m}$. Les accroissements $Z_j = W_{t_j} - W_{t_{j-1}}$ sont indépendants et suivent une loi normale :

$$Z_j \sim \mathcal{N}(0, t_j - t_{j-1}), \quad 1 \leq j \leq m.$$

- **Construction de la trajectoire :**

- $W_{t_0} = 0$.
- Pour $j = 0, 1, \dots, m - 1$, on calcule successivement :

$$W_{t_{j+1}} = W_{t_j} + Z_{j+1}.$$

- En résumé, chaque point est la somme des accroissements jusqu'à la date actuelle :

$$W_{t_j} = \sum_{k=0}^j Z_k.$$

- **Proposition 4.2 :** La trajectoire simulée $(W_{t_j})_{0 \leq j \leq m}$ correspond exactement en loi au vrai mouvement brownien sur les points t_j . En pratique, si l'on utilise un pas constant $\Delta t = t_j - t_{j-1}$, les simulations sont plus simples et régulières.

Remarque 4.3 : Symétrie du mouvement brownien

- **Symétrie :** Le mouvement brownien est symétrique, ce qui signifie que :

$$W \stackrel{\mathcal{L}}{=} -W.$$

Cette symétrie est liée à celle de la loi normale $\mathcal{N}(0, \sigma^2)$, utilisée pour modéliser les accroissements $W_{t_{j+1}} - W_{t_j}$.

- **Intuition :** Si l'on réfléchit une trajectoire W_t par rapport à l'axe horizontal (transformant W_t en $-W_t$), cette trajectoire reste un mouvement brownien en termes de distribution.
- **Application pratique :** Cette propriété peut être utilisée pour la réduction de variance grâce à la méthode des variables antithétiques. En simulant simultanément W_t et $-W_t$, on diminue l'incertitude dans les calculs.

3 Exercice 10 : Simulation d'un mouvement brownien

Exercice 10. Soit $(W_t)_{t \geq 0}$ un mouvement brownien.

- Simulez $n = 10^4$ trajectoires avec un pas de temps de $\Delta t = 0,01$ sur $[0, T]$ (on choisira $T = 1$).
- En déduire un estimateur de $\mathbb{E} [\sup_{0 \leq t \leq T} W_t]$ et donnez sa précision (sans le biais).
- Utilisez $\Delta t = 10^{-4}$ et comparer. On utilisera ce pas pour la suite de l'exercice.
- Proposez un estimateur qui s'appuie sur un processus antithétique et comparez la précision.
- Proposez un estimateur qui s'appuie sur une variable de contrôle et comparez la précision : on utilisera W_T et $\int_0^T W_s ds$.

3.1 Notions abordées

- **Simulation de trajectoires :**

- Simuler $n = 10^4$ trajectoires sur $[0, 1]$ avec un pas de temps $\Delta t = 0.01$. Cela produit $m + 1 = 101$ points par trajectoire.
- Chaque trajectoire est calculée incrémentalement en partant de $W_0 = 0$, avec des accroissements $Z_j \sim \mathcal{N}(0, \Delta t)$:

$$W_{t_{j+1}} = W_{t_j} + Z_j.$$

- Les trajectoires sont stockées dans une matrice de dimension $n \times (m + 1)$, où chaque ligne correspond à une trajectoire et chaque colonne à un instant t_j .

- **Optimisation des calculs :**

- Éviter les boucles imbriquées en utilisant des opérations vectorisées (par exemple, avec NumPy).
- Ajouter les accroissements à toutes les trajectoires simultanément à chaque étape j .

- **Réduction de variance :**

- **Variables antithétiques** : Simuler W_t et $-W_t$ simultanément pour réduire l'incertitude.
- **Variables de contrôle** : Exploiter des grandeurs connues comme W_T ou $\int_0^T W_s ds$ pour améliorer la précision des estimations.

3.1.1 Stockage en mémoire par colonnes

- **Choix de Numpy :**

- La bibliothèque Numpy stocke les données **par colonnes**, ce qui signifie que les éléments d'une même colonne (par exemple, $W[:, j]$) sont contigus en mémoire.
- Une colonne correspond aux valeurs du mouvement brownien pour toutes les trajectoires à un instant donné t_j .

- **Avantages du stockage par colonnes :**

- **Performance optimisée :**

- * Le processeur charge en mémoire des blocs de données adjacentes (cache). Ainsi, lorsqu'une valeur dans une colonne est appelée, les suivantes sont déjà disponibles, ce qui accélère les calculs.
 - * Exemple : Lorsqu'on travaille sur $W[:, j]$ (toutes les trajectoires à l'instant t_j), le calcul est plus rapide, car les données sont contigües en RAM.

- **Inconvénient du stockage par lignes :**

- * Dans le cas d'un stockage par lignes, les valeurs d'une même colonne sont éloignées en mémoire (espacées de n positions). Cela force le processeur à effectuer des appels fréquents en mémoire, ce qui ralentit les calculs.

3.2 Question 1 & 3

```

### Exercice 10 - Simulation du mouvement Brownien
n = 10**4 #nombre de simulations
T = 1.
m = 10**4 #nombre de pas (sans compter la valeur initiale de 0) QUESTION 3
dt = T/m
sdt = np.sqrt(dt)

W = np.zeros([n, m+1]) #Trajectoires du mouvement brownien

for j in np.arange(m):
    W[:,j+1] = W[:,j] + np.random.normal(0., sdt, n)

```

3.2.1 Explication du code

Initialisation des paramètres.

- $n = 10^4$: Nombre de trajectoires simulées.
- $T = 1$: Intervalle de simulation $[0, T]$.
- $m = 10^4$: Nombre de pas de temps pour discréteriser $[0, T]$.
- $\Delta t = \frac{T}{m}$: Taille d'un pas de temps.
- $\sqrt{\Delta t}$: Écart-type des accroissements brownien.

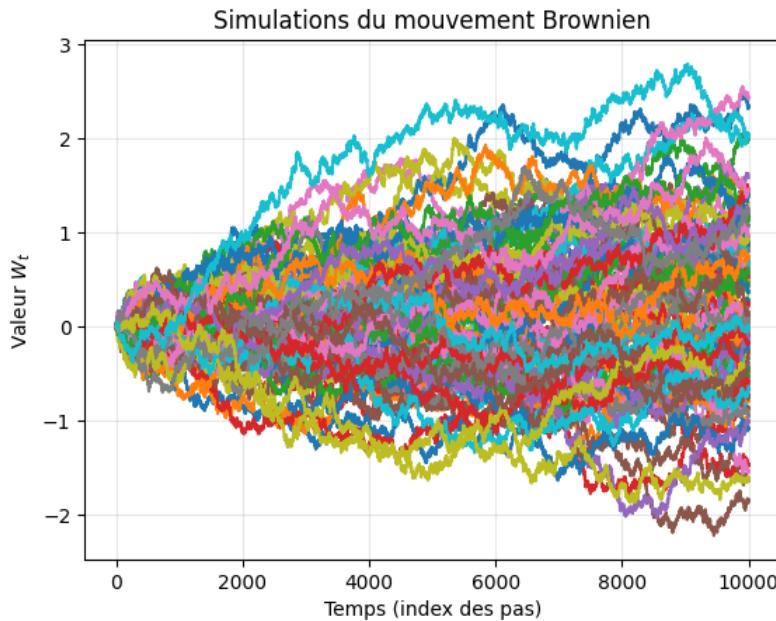
Simulation des trajectoires browniennes.

- Une matrice W de dimensions $n \times (m + 1)$ est initialisée à zéro, où chaque ligne représente une trajectoire discréétisée.
- La boucle `for` met à jour les valeurs de W pour chaque pas de temps :

$$W[:, j + 1] = W[:, j] + Z_j, \quad Z_j \sim \mathcal{N}(0, \sqrt{\Delta t}),$$

où Z_j sont des accroissements indépendants.

Résultat. Chaque ligne de la matrice W contient une trajectoire complète simulée sur $[0, T]$ avec $m + 1$ points.



3.3 Question 2

```
# Méthode classique
# -----
M = npamax(W, axis=1) # Calcul du maximum par trajectoire
em, esd = np.mean(M), np.std(M) # Moyenne et écart-type des max

# Affichage des résultats pour la méthode classique
print("Estimation : %.4f +/- %.4f" % (em, 2 * esd / np.sqrt(n)))

#Vraie valeur : np.sqrt(2/np.pi)
Vrai_M = np.sqrt(2/np.pi)
print("Vraie valeur : %s" % (Vrai_M))

# Estimation : 0.8013 +/- 0.0122
# Vraie valeur : 0.7978845608028654
```

3.3.1 Explication du code

Ligne 1 : Calcul du maximum par ligne. `M = npamax(W, axis=1)` Cette ligne calcule le maximum $M[i]$ pour chaque trajectoire i simulée, en parcourant les valeurs de la matrice W ligne par ligne (`axis=1`). Ainsi, chaque élément de M correspond à :

$$M[i] = \sup_{0 \leq t \leq T} W_t^{(i)},$$

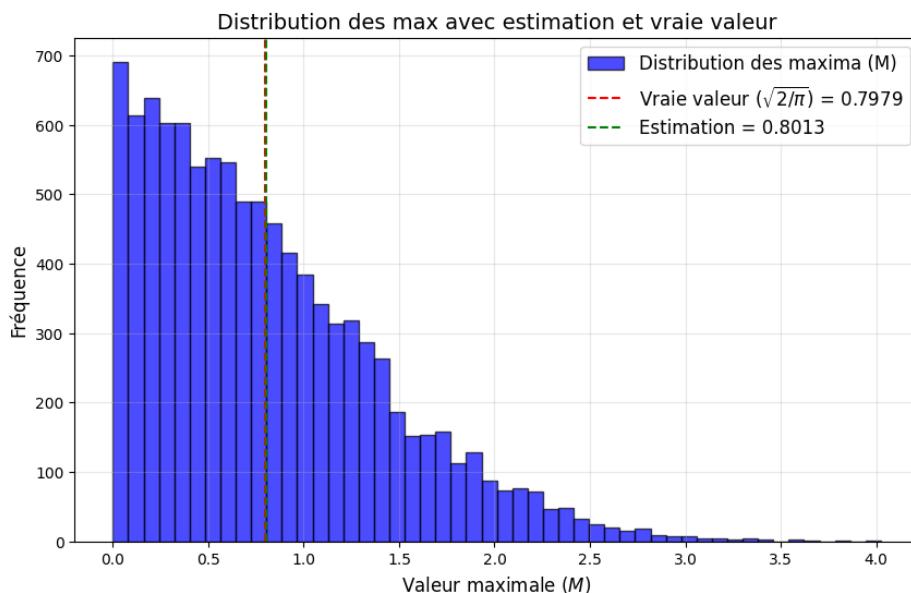
c'est-à-dire le maximum de la trajectoire i du mouvement brownien sur l'intervalle $[0, T]$.

Ligne 2 : Calcul des statistiques: `em, sdm = np.mean(M), np.std(M)`

Ligne 3 : Affichage de l'estimateur Monte Carlo: `print("Estimateur MC : %s ± %s" % (em, 2.*sdm/np.sqrt(n)))`

Ligne 4 (commentaire) : Valeur théorique. `# Vraie valeur : sqrt(2/pi) ≈ 0.797885.`

Cette valeur est connue théoriquement pour un mouvement brownien standard et sert de référence pour évaluer la précision de l'estimation Monte Carlo.



3.3.2 Explication - Maximum d'un mouvement brownien

Cas spécifique : Pour un mouvement brownien standard W_t sur l'intervalle $[0, 1]$, l'espérance du maximum ($\mathbb{E}[\sup_{0 \leq t \leq 1} W_t]$) est donnée par :

$$\mathbb{E}[\sup_{0 \leq t \leq 1} W_t] = \sqrt{\frac{2}{\pi}} \approx 0.797885.$$

Conditions nécessaires : Cette valeur est valable uniquement si les conditions suivantes sont respectées :

- Le processus est un mouvement brownien standard, c'est-à-dire :

$$W_0 = 0, \quad W_t \sim \mathcal{N}(0, t), \quad \text{et les accroissements sont indépendants.}$$

- L'intervalle de temps est $[0, 1]$, avec $T = 1$.

Pour un intervalle général $[0, T]$: Lorsque $T \neq 1$, l'espérance est mise à l'échelle par un facteur \sqrt{T} , car la variance du mouvement brownien est proportionnelle à t . L'espérance devient alors :

$$\mathbb{E}[\sup_{0 \leq t \leq T} W_t] = \sqrt{T} \cdot \sqrt{\frac{2}{\pi}}.$$

Exemples pratiques :

- Si $T = 1$, alors : $\mathbb{E}[\sup_{0 \leq t \leq 1} W_t] = \sqrt{\frac{2}{\pi}} \approx 0.797885$.
- Si $T = 2$, alors : $\mathbb{E}[\sup_{0 \leq t \leq 2} W_t] = \sqrt{2} \cdot \sqrt{\frac{2}{\pi}} \approx 1.128$.
- Si $T = 0.5$, alors : $\mathbb{E}[\sup_{0 \leq t \leq 0.5} W_t] = \sqrt{0.5} \cdot \sqrt{\frac{2}{\pi}} \approx 0.564$.

Conclusion : La constante $\sqrt{\frac{2}{\pi}}$ correspond uniquement au cas particulier d'un mouvement brownien standard sur l'intervalle $[0, 1]$. Pour des intervalles $[0, T]$ avec $T \neq 1$, il faut ajuster cette valeur en fonction de T à l'aide du facteur \sqrt{T} .

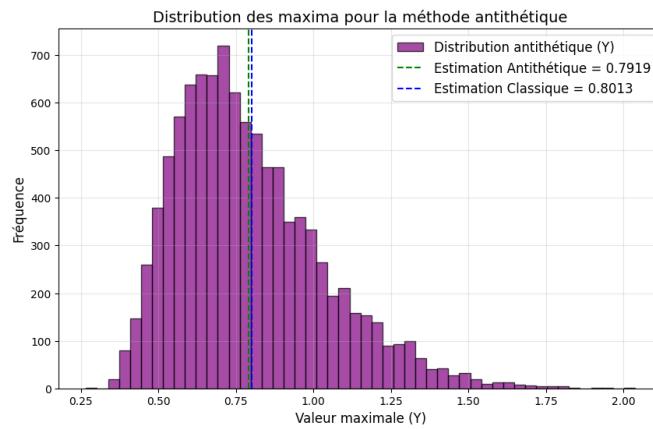
3.4 Question 4

```
# Méthode antithétique
# -----
mW = -W # Matrice des trajectoires opposées
fW = npamax(W, axis=1) # Maximum des trajectoires classiques
fmW = npamax(mW, axis=1) # Maximum des trajectoires opposées
Y = (fW + fmW) / 2 # Moyenne des maxima pour les trajectoires antithétiques

emA, esdA = np.mean(Y), np.std(Y) # Moyenne et écart-type pour la méthode antithétique

# Affichage des résultats pour la méthode antithétique
print("Estimation Antithétique : %.4f +/- %.4f" % (emA, 2 * esdA / np.sqrt(n)))
print("Amélioration Importance : %.2f, soit %.2f fois moins de simulations" % (esd / esdA, (esd / esdA)**2))

Estimation Antithétique : 0.7894 +/- 0.0047
Amélioration Importance : 2.51, soit 6.28 fois moins de simulations
```



3.5 Question 5

3.5.1 Estimation avec la première variable de contrôle : W_T

```
# Étape 1 : Extraction des dernières valeurs (W_T) pour chaque trajectoire
WT = W[:, m] # WT correspond à la dernière colonne de la matrice W, soit les valeurs finales des trajectoires.

# Étape 2 : Calcul de la covariance entre M (maxima) et WT (valeurs finales)
COV = np.cov(M, WT) # Matrice de covariance entre M (maxima des trajectoires) et WT (valeurs finales).

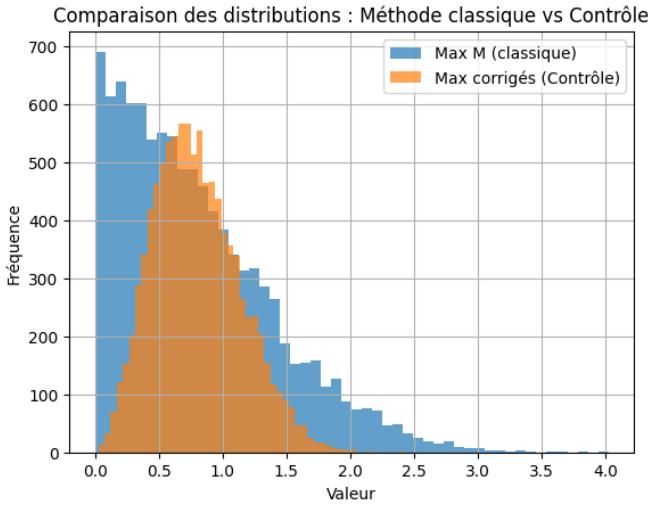
# Étape 3 : Calcul du coefficient optimal pour la réduction de variance
a = COV[0, 1] / COV[1, 1] # Coefficient optimal basé sur la covariance et la variance de WT.

# Étape 4 : Construction de la nouvelle estimation (M corrigé par WT)
fWC = M - a * WT # Nouvelle estimation corrigée avec WT comme variable de contrôle.

# Étape 5 : Calcul de la moyenne et de l'écart-type de la nouvelle estimation
emC, esdC = np.mean(fWC), np.std(fWC) # Moyenne (emC) et écart-type (esdC) pour la méthode de contrôle.

# Résultat avec méthode de contrôle
print("Estimation Contrôle : %.4f +/- %.4f" % (emC, 2 * esdC / np.sqrt(n)))
print("Amélioration Contrôle : %.2f, soit %.2f fois moins de simulations" % (esd / esdC, (esd / esdC)**2))

Estimation Contrôle : 0.7980 +/- 0.0068
Amélioration Contrôle : 1.79, soit 3.21 fois moins de simulations
```



3.5.2 Corrélation entre M et W_T

Définitions :

- M : Vecteur des maxima des trajectoires, où chaque élément est défini par :

$$M[i] = \sup_{0 \leq t \leq T} W_t^{(i)}.$$

- $W[:, m]$: Vecteur des valeurs finales des trajectoires, où chaque élément est donné par :

$$W[i, m] = W_T^{(i)}.$$

Objectif : Calculer le coefficient de corrélation entre M et $W[:, m]$ à l'aide de :

```
np.corrcoef(M, W[:, m]).
```

Interprétation :

- Une corrélation élevée (ρ proche de 1 ou -1) indique que W_T est fortement lié au maximum M . Cela signifie que W_T peut être une **variable de contrôle efficace**.
- Une faible corrélation (ρ proche de 0) signifie que W_T n'explique pas bien M , et son utilisation comme variable de contrôle serait inefficace.

Conclusion : Ce calcul permet de valider l'efficacité de W_T comme variable de contrôle pour réduire la variance dans l'estimation de $\mathbb{E}[M]$.

3.5.3 Estimation avec la deuxième variable de contrôle : $\int_0^T W_s ds$

```
# Contrôle 2 : Utilisation de l'intégrale comme variable de contrôle

# Étape 1 : Calcul de l'intégrale discrète de W_t
IntW = dt * np.sum(W, axis=1)
# IntW correspond à une approximation discrète de l'intégrale \int_0^T W_t dt pour chaque trajectoire.
# np.sum(W, axis=1) calcule la somme des valeurs de W_t sur chaque trajectoire.
# Multiplication par dt pour obtenir une approximation de l'intégrale (méthode des rectangles).

# Étape 2 : Calcul de la covariance entre M et IntW
COV = np.cov(M, IntW)
# COV est la matrice de covariance entre M (maximum des trajectoires) et IntW (intégrale des trajectoires).

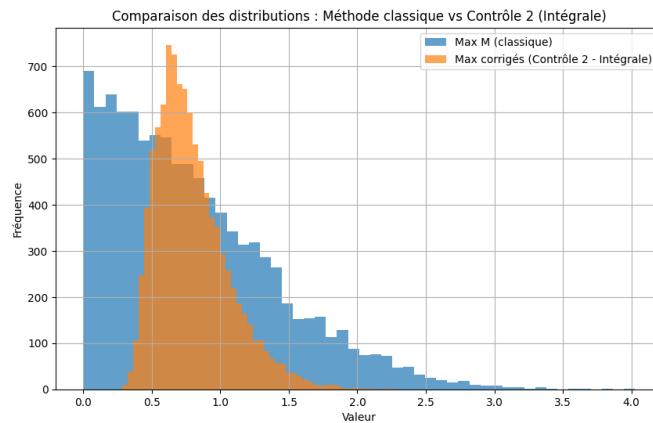
# Étape 3 : Calcul du coefficient optimal pour la réduction de variance
a = COV[0, 1] / COV[1, 1]
# Le coefficient optimal a est défini comme le ratio entre la covariance de M et IntW et la variance de IntW.

# Étape 4 : Construction de l'estimation corrigée
fWC2 = M - a * IntW
# fWC2 est la nouvelle estimation corrigée. IntW est utilisé comme variable de contrôle.

# Étape 5 : Calcul de la moyenne et de l'écart-type de l'estimation corrigée
emC2, esdC2 = np.mean(fWC2), np.std(fWC2)
# emC2 : Moyenne de l'estimation corrigée.
# esdC2 : Écart-type de l'estimation corrigée.

# Estimation avec la deuxième variable de contrôle (IntW)
print("Estimation Contrôle 2 : %.4f +/- %.4f" % (emC2, 2 * esdC2 / np.sqrt(n)))
print("Amélioration Contrôle 2 : %.2f, soit %.2f fois moins de simulations" % (esd / esdC2, (esd / esdC2)**2))
```

Estimation Contrôle 2 : 0.7890 +/- 0.0053
 Amélioration Contrôle 2 : 2.25, soit 5.06 fois moins de simulations



3.5.4 Notions clés sur les variables de contrôle et leur efficacité

1. Justification théorique des choix des variables de contrôle

- Une **variable de contrôle** est une variable aléatoire C pour laquelle :
 - $\mathbb{E}[C]$, son espérance, est connue exactement.
 - Elle est corrélée avec la variable d'intérêt (M , ici le maximum).
- L'estimation corrigée est construite comme suit :

$$M_{\text{corrigé}} = M - a \cdot (C - \mathbb{E}[C]),$$

où a est un coefficient optimisé pour minimiser la variance.

- Deux variables proposées dans le cours :

- W_T : La dernière valeur de la trajectoire brownienne.
- $\int_0^T W_s ds$: L'intégrale de W_s sur $[0, T]$.

2. Lien entre corrélation et réduction de variance

- La réduction de variance est donnée par :

$$\text{Réduction de variance} = 1 - \rho^2,$$

où ρ est le coefficient de corrélation entre M et C .

- **Corrélation élevée (ρ proche de 1 ou -1)** : Réduction significative de la variance.
- **Corrélation faible ($\rho \approx 0$)** : La variable de contrôle est inefficace.

3. Exploration des puissances de W_T

- Transformations possibles : W_T^2, W_T^3, \dots
- Effets sur la corrélation :
 - W_T^2 tend à réduire la corrélation car il perd le signe de W_T .
 - W_T^3 peut améliorer la corrélation en conservant une asymétrie.

4. Intuition sur la corrélation

- Si la trajectoire termine à une valeur élevée (W_T), il est probable que le maximum (M) ait aussi une valeur élevée.
- Cela repose sur le fait que M est influencé par le comportement global de la trajectoire.

5. Validation empirique avec `np.corrcoef`

- Pour valider une variable de contrôle, calculez la corrélation entre M et C :

```
np.corrcoef(M, C).
```

- Cela permet :
 - D'identifier les variables candidates les plus efficaces.
 - De quantifier l'amélioration attendue en termes de réduction de variance.

4 But de la réduction de variance dans la simulation d'un mouvement brownien

4.1 Pourquoi réduire la variance ?

La réduction de la variance est une technique essentielle dans les simulations numériques, notamment dans le cas des simulations de mouvement brownien, pour les raisons suivantes :

- **Précision accrue des estimateurs** : Lorsque la variance des simulations est réduite, les estimateurs des quantités d'intérêt (comme l'espérance ou le maximum d'un processus) convergent plus rapidement vers leur vraie valeur. Cela signifie que pour un nombre donné de simulations, la précision des résultats est améliorée.
- **Réduction du coût de calcul** : Une variance plus faible permet d'obtenir des résultats précis avec un nombre réduit de simulations. Cela diminue significativement le coût de calcul, particulièrement utile pour des simulations intensives.
- **Applications financières** : Dans le cadre de la finance, où les mouvements browniens modélisent souvent l'évolution des prix, réduire la variance des simulations permet d'évaluer plus efficacement les prix d'options ou les risques associés à certains produits financiers.

4.2 Exemple d'application

Prenons l'exemple d'une simulation visant à estimer la valeur maximale M atteinte par un mouvement brownien sur un intervalle de temps $[0, T]$. Les méthodes de réduction de variance incluent :

- **Variable antithétique** : On utilise des trajectoires opposées pour réduire la variance en moyenne.
- **Variable de contrôle** : On ajuste les simulations avec des variables corrélées, comme la dernière valeur du mouvement (W_T) ou son intégrale sur $[0, T]$.

En conclusion, la réduction de la variance est cruciale pour obtenir des estimations fiables à moindre coût, rendant les simulations plus efficaces et accessibles pour des calculs complexes.

5 QCM

5.1 Questions introduction

1. Pourquoi ne peut-on pas simuler un mouvement brownien de manière continue ?

- A) Parce que W_t n'est pas défini sur un intervalle continu.
- B) Parce qu'il est impossible de représenter une infinité de points avec un ordinateur.
- C) Parce que les accroissements ne sont pas indépendants.

Réponse : B

2. Quel est le rôle du pas de temps Δt dans la simulation ?

- A) Définir l'intervalle entre deux points consécutifs sur chaque trajectoire.
- B) Contrôler la variance des accroissements Z_j .
- C) Déterminer le nombre de points à simuler.

Réponse : A et C

3. Comment simule-t-on les accroissements Z_j d'un mouvement brownien ?

- A) En générant des variables suivant une loi normale $\mathcal{N}(0, \Delta t)$.
- B) En générant des variables exponentielles indépendantes.
- C) En calculant directement $W_{t_j} - W_{t_{j-1}}$.

Réponse : A

4. Pourquoi stocker les trajectoires dans une matrice et non dans un tableau unidimensionnel ?

- A) Pour faciliter le calcul parallèle des trajectoires.
- B) Pour optimiser les calculs grâce à des opérations vectorisées.
- C) Parce que les trajectoires ne peuvent pas être indépendantes.

Réponse : A et B

5. Quelle est la principale différence entre une méthode avec et sans boucle forte pour la simulation ?

- A) La méthode avec boucle forte est plus rapide.
- B) La méthode sans boucle forte permet d'utiliser des bibliothèques comme NumPy pour vectoriser les calculs.
- C) La méthode avec boucle forte est nécessaire pour les processus dépendant du passé.

Réponse : B et C

6. À quoi sert la méthode des variables antithétiques ?

- A) À réduire le temps de calcul.
- B) À diminuer l'incertitude sur les estimations.
- C) À vérifier la symétrie du mouvement brownien.

Réponse : B

7. Pourquoi W_T ou $\int_0^T W_s ds$ peuvent-ils être utilisés comme variables de contrôle ?

- A) Parce que leur valeur moyenne est connue ou peut être calculée précisément.
- B) Parce qu'ils réduisent la variance des estimations.
- C) Parce qu'ils sont indépendants des trajectoires simulées.

Réponse : A et B

5.2 Questions essentielles

1. Pourquoi est-il nécessaire de discréteriser le temps pour simuler un mouvement brownien ?

- A) Parce qu'il est impossible de représenter une infinité de points.
- B) Parce que les simulations numériques ne gèrent que les nombres rationnels.
- C) Parce que le mouvement brownien est défini sur des grilles discrètes.

Réponse : A

2. Que modélise la variable Δt dans une simulation ?

- A) La variance des accroissements du mouvement brownien.
- B) La distance temporelle entre deux points.
- C) L'échelle des accroissements simulés.

Réponse : B

3. Quelle propriété des accroissements d'un mouvement brownien est utilisée pour les simulations ?

- A) Leur indépendance.
- B) Leur distribution gaussienne.
- C) Leur décroissance à long terme.

Réponse : A et B

4. Pourquoi utilise-t-on des matrices pour stocker les trajectoires simulées ?

- A) Pour manipuler plusieurs trajectoires simultanément.
- B) Pour faciliter les opérations vectorisées avec NumPy.
- C) Pour calculer directement les accroissements à chaque étape.

Réponse : A et B

5. Qu'est-ce qu'une variable de contrôle dans le contexte des simulations ?

- A) Une variable auxiliaire pour réduire la variance des estimations.
- B) Une variable qui suit une loi uniforme.
- C) Une variable dont l'espérance est connue.

Réponse : A et C

6. Quelle est l'utilité de la méthode antithétique dans les simulations de Monte Carlo ?

- A) Réduire l'écart-type des estimations.
- B) Augmenter la précision des résultats.
- C) Générer des trajectoires supplémentaires pour valider les simulations.

Réponse : A et B

7. Quelle propriété est essentielle pour une bonne variable de contrôle ?

- A) Une forte corrélation avec la variable d'intérêt.
- B) Une variance égale à celle de la variable d'intérêt.
- C) Une indépendance totale vis-à-vis des simulations.

Réponse : A

8. Pourquoi simule-t-on $-W_t$ dans la méthode antithétique ?

- A) Pour exploiter la symétrie du mouvement brownien.
- B) Pour diminuer l'erreur en moyenne.
- C) Pour accélérer les calculs.

Réponse : A et B