עידן אלתר 22.10.17

alteridan@gmail.com

**אנליזה נומרית**

תוכן עניינים של הקורס:

1. ייצור מ"מ מהתפלגויות שונות (דגימות)
2. סימולציית מונטה – קרלו
3. תמחור אופציות אמריקאיות

חבילות פייטון שימושיות – Numpy, matplotlib, spicy.stats

רקע – מהי שיטה נומרית?

* נתונה בעיה מציאותית כגון תמחור אופציה מורכבת.
* ניסוח מודל מתמטי לבעיה (אוסף של משוואות דיפרנציאליות סטוכסטיות = מד"ס)
* מחפשים פתרון למודל (נוסחה) אם אין פתרון מפורש, מפעילים שיטה נומרית ומוצאים פתרון מקורב.
* יישום השיטה הנומרית ע"ב מחשב.

**תזכורות** - [משפט הגבול המרכזי](https://he.wikipedia.org/wiki/משפט_הגבול_המרכזי) (CLT) – התפלגות של מס' משתנים מקריים שווי התפלגות בת"ל, מתקרב להתפלגות נורמלית.

**שיטת נק' שבת**

נפתור את המשוואה הבאה :

X^2 -3X +2 = sinX \*1/12

במשוואות בהן הנעלם בתוך פונקציה טריגונומטרית ולא בחוץ, לא ניתן לחלץ את הנעלם.

ננחש נעלם מסויים, נציב באגף ימין (X=1):

Sin 1 \* 1/12 = X^2 -3X +2 => X=0.93

נציב באגף ימין:

Sin 0.93 \* 1/12 => X=0.94

נציב שוב אגף שמאל:

X^2 -3X +2 = sin0.94 \*1/12 => X=0.94

לסיכום: נעשה איטרציות עד שנצליח לקבע ברמת דיוק מסויימת (2 ספרות אחרי הנקודה), את אותו ה-X בדיוק.

**שיטת ניוטון – ראפסון (NR)**

נתונה פונקציה f(x) ונחפש X כך שעבור אותו איקס הערך יהיה אפס f(x) = 0

הרעיון הוא שברזולוציה מספיק גבוהה, הפונקציה דומה לקו ישר.

נחפש נק' החיתוך של הפונקציה עם ציר האיקס. נעביר משיק בנק' שננחש,ונחפש את החיתןך של המשיק עם ציר ה-X. את החיתוך נסמן X1, ונבצע איטרציה נוספת.

אלגוריתם:

1. נעביר משיק לפונקציה ב- X1
2. נחפש את החיתוך של המשיק עם ציר ה-X
3. נסמן חיתוך בX2, ונחזור על התהליך.

נעצור כאפשר האיקס לא משתנה

משפט: אם f גזירה ברציפות, וX0 מספיק קרוב לפתרון, אז השיטה תתכנס לפתרון.

כחול ->צהוב--> כתום .... נתכנס לפתרון.

כדי לפתור זאת, נדרש לתכנת נגזרת. הנ"ל לא מיידי, נדרש התאמה ע"ב אינפי = הגדרת גבולות ונגזרת. עבור u -> 0 (מספר קטן מאוד).

lim (f(x+u) – f(x)) / u = f'(x)

[קישור לפתרון](Python/Numerical%20Methods/NR_BLS.py)

הסבר: פונקציה צעד חדש מקבלת ניחוש לפתרון, ומחזירה ניחוש הבא לפי שיטת ניוטון ראפסון.

נדרש קו ישר העובר בנק' (x1, f(x1)) בעל שיפוע f'(x1).

אח"כ יש למצוא את החיתוך שלו עם ציר X. הפתרון יהא נק' החיתוך עם ציר ה-X של המשיק. X2=X1 – f(x1)/f'(x1)

**שונות מוטמעת** (Implied Volatility)

עבור משוואת B&S Call = fi(d1)\*S – fi(d2)\*k\*e^(-rt)

Put = fi(-d2)\*k\*e^(-rt) - fi(-d1)\*S

כאשר:

d 1 = (ln(S/K) + (r+sigma^2/2)\*t) / sigma\*t^(1/2)

d 2 = (ln(S/K) - (r+sigma^2/2)\*t) / sigma\*t^(1/2)

עפ"י נתוני השוק, נדע מחיר האופציה, מדד בסיס, מחיר מימוש, זמן וריבית = לכן נדע להעריך את סטיית תקן (נדיפות)

שלב 1 = נניח שהשוק מתמחר את האופציות בדיוק לפי הנוסחא, ונסתכל על המשוואה ונלץ את סטיית התקן המתאימות שהופכות הנחה זו לנכונה.

שלב 2 = נפתור עפ"י שיטת ניוטון ראפסון. הפתרון נקרא שונות מוטמעת.

[קישור לפתרון](Python/Numerical%20Methods/blspricingexample.py)

במקרה שלנו, הנעלם הוא סטיית התקן, כפי שתסביר הדוגמא הבאה:

[קישור לפתרון](file:///C:\Users\User\Desktop\Python\Numerical%20Methods\NR_BLS.py)

F(ϭ) = bs\*put(ϭ) – p

מחיר השוק, מחיר השוק לפי הפונקציה

נחפש ϭ כך ש – f(ϭ) = 0 :

ϭ2= ϭ1 – (bs\*put(ϭ1) –p) / bs\*put'(ϭ1)

**ייצור משתנים מקריים**

אנו רוצים לייצר אוסף של מספרים בעלי תכונות הדומות להתפלגות של מ"מ כלשהו. מה נצפה?

1. תכונה בסיסית – היסטוגרמה של שכיחות המ"מ אשר תהא דומה לצפיפות שלו
2. אי תלות בין דגימות – אם יודעים את הערכים עד N, אי אפשר להסיק דבר על ערכו של הערך הבא (N+1) - אף פעם לא נקבל תכונה זו. בדר"כ קיימת תלות כלשהי ונסתפק בתכונה חלשה יותר.
3. חוסר קורולציה בין איברים בסדרה.

**ייצור משתנים פסודו-רנדומליים**

ייצור דגימות של משתנה אחיד U[0,1] הוא הבסיס לכל התפלגות. שכן מדובר על מ"מ:

1 עבור ערכים בין אפס לאחד, אחרת אפס.

אלגוריתמים:

1. **LSG - Linear Sequential Generator –** מייצר סדרה של מ"מ לפי הכלל הבא:

Xn+1 = (aXn +b) mod c

למעשה, שארית החלוקה של קו ישר, במספר קבוע כלשהו מייצרת סדרת מספרים בין 0 ל-1.

אין חזרות, אין סדר נראה לעין, עד האיבר ה-c (שם המשתנה שווה 1). אבל מייצר סדרה מחזורית עד –c, לכן נרצה שיהיה מס' גדול, השקורולציה לא תחזור בסדרה.

הרוב כיום משתמשים ב- Hersenne Twister עם משתנים סמויים שלא נגישים למשתמש.

1. **Random.org –** מבוסס על קרינה אלקטרומגנטית מהחלל, רנדומלי אמיתי. ולכן נניח שיש לנו גישה בלתי מוגבלת למשתנה אחיד בין 0,1.

המטרה היא לייצק דגימות להתפלגות נורמלית, אקספודנציאלית וכו'...

1. **שיטת ההמרה/היפוך/טרנספורמציה-** נרצה להביע את ההתפלגות המצטרפית של X ע"י משתנה אחר – Y. נבחר בחוכמה פונקציה y(t) כך שאם U~U[0,1], אז y(u)~Fx. אז נציב u בy ונקבל: y(u1) = x1, y(u2) = x2 דגימות של X.

נסמן את y(u) להיות F(x)^(-1) כאשר Fx היא ההתפלגות המצטברת של X וFx^(-1) היא ההופכית שלה.

משפט: אם U~U[0,1], אז למ"מ Fx(u)^(-1) יש אותה התפלגות כמו לX. לכן y עובדת עבור שיטת הטרנספורמציה.

דוגמא: x~exp(r)

Fx(t) = 1-e^(-rt)

נרצה למצוא את ההופכית

Fx(t)=S => 1-e^(-rt) = S => 1-S = e^(-rt) => ln(1-s) = -rt =>

t= -ln(1-S)/r = Fx(s)^(-1)

מצאנו ביטוי סגור לכן נדגום משתנה אחיד, ונציב בביטוי זה:

X1 = ln(1-u1)/r

X2 – ln(1-u2)/r

בעיות בשיטה זו:

1. יש מ"מ עבורם הפונקציה המצרפית ההופכית לא קיימת.
2. יש מ"מ שלא ניתן לחלץ את t מהמשוואה: Fx(t) = S ולכן אין דרך פשוטה להביע את Fx(s)^(-1) (עבור נורמלי להפטר מהאנטרגל)
3. ניתן לייצר בעזרת שיטת Box Muller / Harsaglia

[קישור](Python/Numerical%20Methods/HW1.py)

1. שיטת ההדחה (פחות יעיל): נניח שY דומה לX וקל לדגום אותו. בהתאם לערך שהתקבל נחליט האם הוא מתאים לX, אם לא נדחה אותו.

אלגוריתם:

1. מוציאים מספר c>1 כך ש: fx(s)/fy(s) <c לכל c/

חישוב אנליטי לפני הרצת האלגוריתם. ע"ב ערך c נחליט אם לקבל או לדחות ערך כשמתאים לדגימה מX.

1. מייצרים דגימה Yi מY.
2. מייצרים דגימה חדשה ובת"ל של U[0,1] ונסמן Ui
3. אם מתקיים השיוויון הבא אז Yi מתאים אחרת זורקים את הערכים ומתחילים ב' מחדש.

Ui< fx(Yi) / (c \*fy(Yi))

הערות:

בסופו של דבר נמצא Yi מתאים ונקבל Xi דגימה מ-X. נחזור על שלבים ב-ד' ונייצר סדרה של X. כיצד נמצא את c? הוא בדיוק תוחלת של משתנה שמתפלג גאוסיאנית = 1/p = 1/c. והוא גם ממוצע מס' הדחיות. הוכחה:



1. **אלגוריתם זיגורט (Ziggurat)** – פירמידות של דרום אמריקה.

דומה לשיטת הדחייה,רוצים לדגום מ"מ רציף, מסביב לצפיפות שלו בונים זיגורט (מלבנים משתנים בגדולים לאורך הצפיפות). בקורס נקבל קודקודים.

תכונות נדרשות:

1. יכסה את הפונקציה
2. דטח כל מדרגה קבוע ושווה לשטח כל מדרגה אחרת
3. סה"כ השטח קטן ככל האפשר.

האלגוריתם: הגיון – משתמנים בשיטת הדחייה לייצר דגימות של המ"מ X בעזרת דגימות של מ"מ אחר Y, שהצפיפות שלו היא זיגורט.

1. בוחרים מלבן אקראי
2. מייצרים קאורדינאטת מ"מ X אקראית בתוך המלבן הנבחר (לפי קודקוד מלבן (a,b)
3. דוגמים Zx, אחיד (a,b) בודקים מעל הקורדינאטה בס' ב' האם הפונקציה של הקאורדינטה קטנה משטח המלבן.
4. במקרה הזה הגג קטן מהצפיפות – נחזיר דגימה של מ"מאו
5. גדול (שטח הקו בפנים ביחס למלבן בחוץ) – נדגום מ"מ נוסף Y, כך שהנקודה בתוך צפיפות של X, ונחזיר Zx כדגימה של X אם קטן ממנו, אחרת, נחזור לא'.

בהמשך נבנה זיגורטים כך שבדיוק כאשר הצפיפות עולה מעל הגג של קומה I ומתחילה בקומה I+1 , המשמעות שהבדיקה של סעיף ג' היא פשוטה – האם Zx מתחת לקומה הבאה או לא.

1. **שיטת BOX-MULLER:**

אלגוריתם:

1. יש לדגום 2 משתנים בת"ל בהתפלגות אחידה (0,1) – U1 U2
2. להציב:
3. teta = 2 \* פאי \* U1, המשמעות: תטא אחידה - פיזור אחיד ביחס לזויות
4. רדיוס = שורש מינוס 2 \* ln(U2) המשמעות: גדול לא סביר לקבל וקטן יותר סביר.
5. להציב:
6. Z1 = R cos (teta)
7. Z2 = R sin (teta)
8. **השיפור של מארסאגליה (Marsaglia):** כמו בוק מילר, בלי לחשב את הפונקציות הטריגונומטריות שלעיל. דוגמים ישירות על העיגול.
9. יש לדגום 2 משתנים בת"ל בהתפלגות אחידה (0,1) – U1 U2
10. מציבים: V1 = 2U1-1, V2=2U2-1 ומקבלים U[-1,1] שייךV1,V2
11. אם V1^2 + V^2 >1, חוזרים לא'
12. אחרת, אם עבור V1^2 + V^2 =R, מציבים – Z1/2=V1/2\*sqrt(-2ln® / R)

כאשר Z1/2 הם נורמלי 0,1.

1. **שיטת מונטה קרלו MC :**

חישוב אנטגרלים – נרצה לחשב אינטגרל רגיל אבל לפונקציה אין פונקציה קדומה.