① 18|10|09
חישוב
נומרי

הרצה + מתרגל: רענן פטאל

raanan.f@cs.huji.ac.il

רוס 76. באתר התרגיל הודעה על שעות קבלה.

יהיו בקורס 6 תרגילים שמורכבים מ- 3-5 שאלות תיאורטיות ומעשיות (= סקריפט קצר במטלב)

לאתר הקורס אפשר להגיע דרך האתר של רענן. שם מופיע הסילבוס ולינקים לעוד מקומות מעניינים. גם בויקיפדיה מופיעה של...

לקורס שתי מטרות:

- לתת לנו כלים להתמודד עם בעיות נומריות שניתקל בהן
- לעניין אותנו בתחום תורת הקירובים

התרגיל הראשון הוא תרגיל חזרה על חומר בסיסי שצריך לדעת. אבל לא להיבהל - זה לא קורס מתמטי כבד.

סקירה

אינטרפולציה

יש מספר סופי של ערכים של פונקציה. ורוצים להעריך אותה בנקודה אחרת.
זה יכול להיות שימושי אם יש פונקציה שמאוד קשה לחשב אותה ולכן יש לנו רק כמה ערכים בודדים, או במקרה של מדידות נסיוניות.
עוד דוגמא היא במידול גאומטרי - אנחנו רוצים לתת כמה נקודות במרחב ושהמחשב יחבר את זה למשטח חלק.
נדבר על כל מיני סוגי אינטרפולציה שאפשר לעשות וננסה להעריך את השגיאה של האינטרפולציה שלנו.

אינטגרציה

יש הרבה פונקציות שאי אפשר או מאוד קשה לחשב להן אינטגרל (פונק' ללא אנליטיות או אינטגרציה בממדים גבוהים). במקרה הזה נלמד שיטות נומריות כדי לשערך את האינטגרל. שימוש של זה בגרפיקה הוא למשל חישוב תאורה בתמונה...

מערכות משוואות לא לינאריות

איך נפתור משוואה כמו ... = $\sqrt{\sin x}$? עד עכשיו למדנו על משוואות לינאריות. שם הכל היה יפה. אבל בחיים האמיתיים המשוואות לא יפות. למשל טריאנגולציה ב- GPS המשוואות לא לינאריות. אז ה- GPS כל הזמן עושה קירובים נומריים בשביל לחשב את המיקום שלנו...

מערכות משוואות לינאריות

הרבה יותר קלות לפתרון ובתנאים מסוימים מובטח קיום פתרון. נלמד ~~על~~ כל מיני שיטות שמקובל להשתמש בהן רבות בבעיות לינאריות. נפגוש כל מיני משוואות לינאריות מפורסמות. לבסוף נראה איך משוואות לינאריות עוזרות לנו עם משוואות לא לינאריות.

ערכים ווקטורים עצמיים

זה נושא חשוב במדעי המחשב- בעיות אופטימיזציה, למידה, clustering.
יש גם שימושים בפיזיקה.

אופטימיזציה

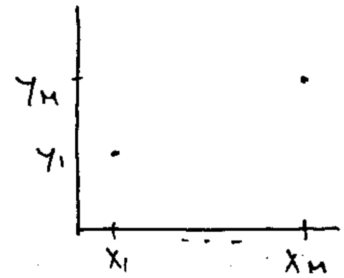
זה כמובן יכול להיות שימושי בכל תחום. אנחנו נניח כאן שהפונקציות אנליטיות ופחות ניגע בכללי גזרה.

משוואות דיפרנציאליות רגילות

אלה משוואות די פשוטות להבנה ולא נתעסק יותר מדי בתאוריה. נניח שמדובר בפונקציה טובה שמקיימת כל מה שצריך ונראה איך אפשר לקרב אותה. חוקי ניוטון הם למעשה משוואות דיפרנציאליות. מודלים כלכליים משתמשים במשוואות דיפרנציאליות...

אינטרפולציה

הגדרת הבעיה: נתונה קבוצת ערכים של פונקציה - קבוצה סופית.
$\{(x_i,y_i)\}_{i=1}^{M}$ כאשר $x_i\in[a,b]$



המטרה היא לשערך את $f(x)$ עבור x כלשהו. אם $x\in[a,b]$ הבעיה נקראת <u>אינטרפולציה</u>. אם $x\notin[a,b]$ הבעיה נקראת <u>אקסטרפולציה</u>.

כאן אנחנו נניח שהפונקציה f שמחפשים היא יפה ונחמדה - חלקה, רציפה וכו'.

לפתרון שני שלבים:

① "התאמה" - fitting - מציאת פונקציה $\hat f$ כך ש- $\hat f(x_i)=y_i$ לכל i. למעשה $\hat f$ בדרך כלל תלויה באילו פרמטרים $C_1,\dots,C_M$ שאותם אנחנו מוצאים בעזרת הנתונים שלנו.

$\boxed{\text{כאמור הבעיה היא למעשה למצוא את } C_1,\dots,C_M \text{ כך שלכל } i \text{ יתקיים } \hat f(x_i,C_1,\dots,C_M)=y_i}$

② הערכה - evaluation - חישוב $\hat f(x)$

<u>דוגמה:</u>
$f(1)=1$
$f(2)=2$

ואנחנו מניחים שמחפשים $\hat f(x,c_1,c_2) = xc_1^2+x^2c_2$ ← אינטרפולנט

יש פה היפוך מסוים של תפקידים. ה-x הוא קבוע במובן מסוים. כי הדבר ש-x נותן לנו אילוץ:

$$f(1)=1 \Rightarrow c_1^2+c_2=1$$
$$f(2)=2 \Rightarrow 2c_1^2+4c_2=2$$

זו מערכת משוואות לא לינארית ובפרט קשה לפתור את זה!!

<u>הגדרה:</u> אינטרפולציה לינארית $\hat f$ היא מהצורה
$$\hat f(x,c_1,\dots,c_M)=\sum_{i=1}^{M} g_i(x)\,c_i$$
כאשר g_i פונקציות כלשהן. סדר האינטרפולציה הוא $M-1$.
האילוצים $\hat f(x_i,c_1,\dots,c_M)=y_i$ נותנים לנו את מע' המשוואות ←

הינארית $\forall j \quad \sum_{i=1}^{M} c_i g_i(x_j) = y_j$

ובצורה מטריציונית:

$$\begin{array}{cc} & i \\ j & \begin{pmatrix} & \vdots & \\ \cdots & g_i(x_j) & \\ & & \end{pmatrix} \end{array} \begin{pmatrix} c_1 \\ \vdots \\ c_M \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_M \end{pmatrix}$$

דוגמה: אינטרפולציה פולינומיאלית היא אינטרפולציה לינארית שבה $g_i(x)=x^{i-1}$
אז מע' המשוואות מקבלת את הצורה

$$\begin{pmatrix} 1 & x_1 & \cdots & x_1^{M-1} \\ \vdots & \vdots & & \vdots \\ 1 & x_M & & x_M^{M-1} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_M \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_M \end{pmatrix}$$

וזו מטריצת ון-דר-מונדה! לא לא מאוד קריטי אבל כדאי לשים ♡ לדברים
כאלה כי למשל למטריצת ון-דר-מונדה יש תכונות נחמדות למיניהן.

באופן שבו זה רשום האינטרפולציה כרוכה בפתרון מערכת משוואות שזה בשיטה
פשטנית (גאוס) ובלי הנחות מיוחדות לוקח $O(M^3)$. אז ה- fitting
היה יקר. אבל עכשיו את ההערכות אפשר לעשות בקלות ב- $O(M)$.
אחרי שיש לנו ביד את הפולינום רק להשתמש בו כמה שרוצים כדי להעריך.

גישה אחרת לפתרון הבעיה היא שימוש בפולינום לגרנז'. שהוא בעצם הפתרון
של המשוואות למעלה: הפולינום שהוא מקיים את אותם אילוצים והוא מאותו
סדר זה חייב להיות אותו הפולינום.

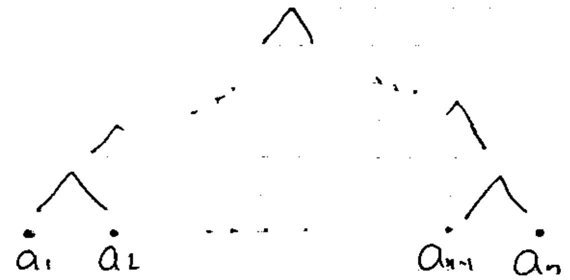
$$p(x) = \sum_{i=1}^{M} y_i \frac{\prod_{j\neq i}(x-x_j)}{\prod_{j\neq i}^{M}(x_i-x_j)}$$

קל לראות ש- $p(x_i)=y_i$ לכל i ; x-ים מופיעים במכפלת $M-1$
לכל היותר ולכן זה הפתרון של מע' המשוואות.
אלא שיש כאן הבדל מבחינת עלות ההערכה. כאן ההערכה היא ב- $O(M^2)$
(M מחוברים ובכל אחד $2M$ מכפלות). שזה פחות טוב מקודם אבל אין
את המחיר של ה- fitting.

3

מהנוסחה של $\boxed{}$ $p(x)$ רק אפשר לחשב את הפולינום עצמו ע"י פתיחת סוגריים. העלות של זה תהיה $O(M\cdot 2^M)$ שזה ממש איום!!

בשתי השיטות האלה יש בעיה (משהו שלא נדבר עליו הרבה) - roundoff errors שזה אומר חישוב של המחשב שנובעות מהייצוג של מספרים ממשיים. אפשר לעשות פה אנליזה מסודרת אבל זה הבעיה היא לא לערבב בחישובים מספרים בסדרי גודל שונים. שיטה שאפשר להשתמש בה היא לסכום בפירמידה:



סטטיסטית כנראה שבסוף שלב נסכם מספרים שהם בערך באותו סדר גודל. הבעיות הללו בעיקר באות לידי ביטוי כשהפולינום מספרים מאוד גבוהים (כי אם אנו $x_i < 1$ ו- $x_j > 1$ אז יש הבדלים גדולים מאוד בין x_i^{M-1} ו- x_j^{M-1}). הפתרון לבעיה הוא שימוש בפולינומים ממעלה נמוכה (spline - בשיעור הבא) או שימוש בשיטת Neville.

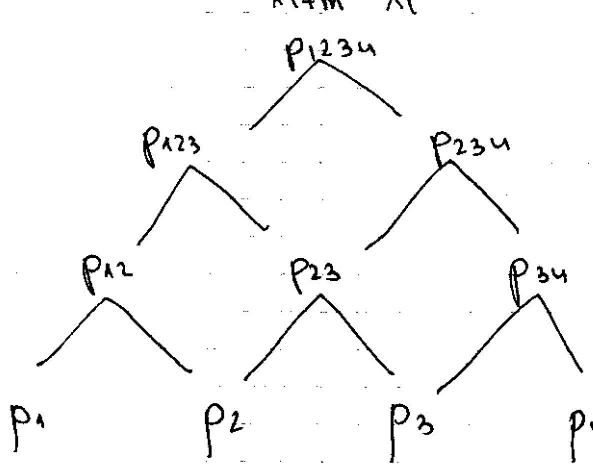
שיטת Neville - נבנה את הפולינום שלנו בשלבים.

נבנה סדרת פולינומים רק ש- $\forall j=i,\dots,i+m \quad p_{i,\dots,i+m}(x_j) = y_j$

הפולינומים הכי פשוטים בסדרה הנו הם $p_i(x) = y_i$ (פולינומים ממעלה 0). מהם נעשה קומבינציה ממעלה גבוהה יותר. יש לזה נוסחה רקורסיבית.

$$p_{i,\dots,i+m}(x) = \frac{(x-x_i)\,p_{i+1,\dots,i+m}(x) + (x_{i+m}-x)\,p_{i,\dots,i+m-1}(x)}{x_{i+m}-x_i}$$

גם שבה יש השפעה בסדר הנקודות פרט קיצוניות



לא משתמשים בשיטה הזאת כדי למצוא את הפולינום ולעקוב אחרי המקדמים, אלא פשוט מציבים את ה-x הרצוי. כמובן, גם פה אין שלב של fitting. אבל היתרון כאן הוא שאלא אם מדובר בנוסחאות מיוחדות, מסכמים כאן בד"כ מספרים מאותו סדר גודל.
עלות ההערכה כאן היא $O(n^2)$. אז לא שיפרנו זמנים אבל שיפרנו דיוק.
בשיעור הבא נעשה הערכה אנליטית של אינטרפולציה פולינומיאלית.

(4)

22.10.09
חישוב נומרי

מי שחדש פה שישלח מייל לרענן - raananf@cs.

דיברנו על אינטרפולציה. יש לנו אוסף סופי של נקודות של
פונקציה ואנחנו רוצים לשערך אותה בנקודה אחרת.
בד"כ משתמשים באינטרפולציה לינארית - אינטרפולציה שבה
האינטרפולנט תלוי לינארית בפרמטרים שלו. רק כשממש לא מוכנים
להתפשר (אם למשל יש מודל פיסקלי מדויק) אז אולי
משתמשים באינטרפולציה לא לינארית.
בפרט, דיברנו על אינטרפולציה פולינומיאלית - זו אינטרפולציה
לינארית שבה הפרמטרים הם מקדמים של פולינום. אז ראינו 3
שיטות לפתרון הבעיה.

I מציאת c_i ע"י פתרון מטריצה. שלב ההתאמה הוא $O(M^3)$
וההערכה היא בזמן $O(M)$.

II שימוש בפולינום לגרנז' - כאן אין שלב של התאמה. עושים מיד
את ההערכה וזה לוקח $O(M^2)$. אמנם אפשר לחלץ מהפולינום
את המקדמים אבל זה לוקח זמן מעריכי - ממש סיפור ביש.

III שימוש בשיטת Neville - עושים הערכה של הפונקציה בצורה
היררכית. הערכת הפולינום היא ב- $O(M^2)$ וזה (עושים) בלי שלב
התאמה. היתרון של השיטה הזו הוא שהיא עמידה יותר לבעיות של
round off errors

היום נדון על השגיאות של האינטרפולציות השונות.

<u>משפט:</u> נניח ש- $\hat{f}(x)=p(x)$ עוברת דרך (x_i,y_i) עבור $i=0,\dots,M$
ונניח שלפונקציה f יש $M+1$ נגזרות בקטע $[a,b]$.
אז קיים $\xi\in[a,b]$ כך ש-

$$f(x)-p(x)=R(x)=\frac{(x-x_0)(x-x_1)\cdots(x-x_M)}{(M+1)!}\cdot f^{(M+1)}(\xi)$$

$$\left(\omega(x)=\prod_{i=0}^{M}(x-x_i)\right) \qquad =\frac{\omega(x)}{(M+1)!}f^{(M+1)}(\xi)$$

הוכחה: נשים ♡ ש- $R(x_i)=0$ לכל $i=0,\dots,M$ (מההנחה ש-p מסכים עם f על (נקודות הקלט).

נגדיר פונקציה $S(x)$ באופן סתום $\omega(x)S(x) = R(x)$

(פונקציה כזאת היא לא יחידה. אנחנו בוחרים אחת שמקיימת את זה).

נקבע את x ונגדיר פונקציה של משתנה חדש z:

$$F(z) = f(z) - p(z) - \omega(z)S(x)$$

לפונקציה $F(z)$ יש $M+2$ שורשים: $z=x$, $z=x_0,\dots,x_M$

כי
$$F(x_i) = f(x_i) - p(x_i) - \omega(x_i)S(x) =$$
$$= f(x_i) - f(x_i) - 0\cdot S(x) = 0$$
$$F(x) = f(x) - p(x) - \omega(x)S(x) =$$
$$= f(x) - p(x) - R(x) = R(x) - R(x) = 0$$

ממשפט רול [אם $f(a)=f(b)$ אז יש $c\in(a,b)$ רק ש- $f'(c)=0$] נובע שיש $M+1$ שורשים ל- $F^{(1)}$.

וכך הלאה ל- $F^{(M+1)}$ יש שורש אחד (לפחות).

מתקיים $F^{(M+1)}(z) = f^{(M+1)}(z) - (M+1)!\,S(x)$

יש שורש ξ - $F^{(M+1)}(\xi)=0$. לכן

$$0 = f^{(M+1)}(\xi) - (M+1)!\,S(x)$$
$$\Rightarrow \quad S(x) = \frac{f^{(M+1)}(\xi)}{(M+1)!} \quad (*)$$
$$\Rightarrow \quad R(x) = S(x)\omega(x) = \frac{\omega(x)}{(M+1)!} f^{(M+1)}(\xi)$$

וכך לכל x מצאנו ξ שמקיים את הדרוש!

הערה: יש פה איזה חיפוף בשלב - הטענה כנראה נכונה עבור נקודות שאינן x_i, כי אז ההוכחה נכונה שם. אבל S יחידה כי היא מוגדרת ע"י ω ו-R באופן יחיד ואין גם קושי בשאלה ביטוי מפורש (*). אבל אם $x=x_i$ אז $R(x)=0$ ו- $\omega(x)=0$! ולכן כל ξ יקיים את מה שצריך בלאו הכי.

יש ל שעבורו $w(x) = \prod(x-x_i) \le M!\,\delta^{M+1}$ ולכן

$$R(x) = \frac{f^{(M+1)}(\xi)}{(M+1)!}\,w(x) \le \delta^{M+1} f^{(M+1)}(\xi)$$

אז ככל שדוגמים יותר נקודות מקטע שלנו δ קטן והאינטרפולציה אמורה להיות טובה יותר. δ הוא למעשה המרחק המקסימלי בין שתי נקודות דגימה עוקבות.

יש פה משחק בין סדר הנגזרת לסדר הפולינום. החסם פה תלוי גם ב-δ (שקטן עם המעלה) אבל גם ב-$f^{(M+1)}$ ויש נטיה לפונקציות שככל שהנגזרת מסדר גבוה יותר החסם עליה גם גבוה יותר. אז מספיק שיש נקודה שבה הנגזרת משתגעת וכל החסם שלנו לא שווה הרבה.

$$|R(x)| \le K \cdot O(\delta^{M+1})$$

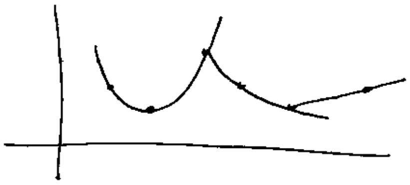
$$K = \max_{\xi \in [a,b]} |f^{(M+1)}(\xi)|$$

מסקנות / סיכום / הערות:

- האמת היא שהקטע $[a,b]$ שלנו מוגדר ע"י $a=x_0$, $b=x_M$ ואז באמת כשמוסיפים עוד נקודות דגימה בתוך $[a,b]$ אז $\delta=\max_{i,j}|x_i-x_j|$ קטן, והחסם שלנו משתפר. (כמו אנחנו מגבילים את עצמנו לאינטרפולציה ולא לאקסטרפולציה).
- הוספת נקודות דגימה משתלמת כמו $O(\delta^{M+1})$ אם מדובר בפונקציות שהנגזרות הגבוהות שלהן לא משתגעות.
- החסם עלול לגדול עם סדר הנגזרת. זה לא אומר שהקירוב בטוח יהיה פחות טוב, אבל אין לנו הבטחה שהוא יהיה טוב. (זה גם פחות יעיל חישובית כמובן)

Splines - אינטרפולציות פולינומיאליות למקוטעין

אם אתה לא מכריח אותו להתאים את כל הנקודות בבת אחת.
אפשר לעשות התאמה לקבוצות של נקודות



<u>אלטרנטיבה I</u>:
חלוקת הקטע לתת מקטעים ה תחומים
ע"י N נקודות - בכל אחד מהם:
ושימוש בהתאמה מעלה $N-1$ בכל קטע.

<u>אלטרנטיבה II</u>:
בחירה שרירותית של N נקודות לאינטרפולציה (מסדר $N-1$).
בחירת הנקודות יכולה להיות שרירותית או פרקטית בד"כ משאירים
את הנקודות הקרובות ביותר לנקודה שאנחנו רוצים לשערך בה.

באלטרנטיבה הראשונה אנחנו בוודאי מקבלים פונקציה רציפה. אבל היא
לא בטוח גזירה בנקודות התפיסה. אז אנחנו יכולים להוסיף
עוד אילוץ על הנגזרת - ושנתפס את ההתאמה בקטע השני
נדרוש שהנגזרת בנקודה הראשונה, שווה לנגזרת של
ההתאמה של הקטע הראשון בנקודה הזאת. זה מוסיף עוד
אילוץ לינארי למערכת ומגדיל את סדר האינטרפולציה ב-1.
אם יודעים את הנגזרות של פונקציית המטרה אז כמובן נדרוש
הסכמה איתן. הפרישה הזו נקראת natural splines

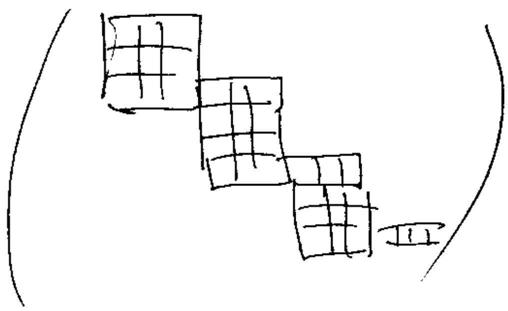
כמה אילוצים עבור התאמה של natural splines:
M נקודות $\Longrightarrow$ 2M אילוצים:

$$\begin{cases} p_1(x_1) = y_1 \\ p_1(x_2) = y_2 \end{cases}$$

$$\begin{cases} p_2(x_2) = y_2 \\ p_2(x_3) = y_3 \end{cases}$$

אילוצי הנגזרות נותנים M אילוצים- $p_1'(x_2) = p_2'(x_2)$: וכו'.

כדי לפתור את זה בצורה סדרתית אנחנו צריכים איזשהו
מידע על הנגזרות בקטע הראשון. ואם יש לנו מידע כזה
אז באמת ניתן לפתור את הקטע הראשון ואז האילוץ
$p_2'(x_2)=p_1'(x_2)$ לא באמת מערב מקדמים של פולינומים
שונים, אלא $p_1'(x_2)$ זה ממש מספר ספציפי.
אבל אם אין לנו מידע כזה אז המשוואות לכל קטע הן
תלויות (סימטריות) זו בזו.
אז המטריצה כולה היא לא מטריצת בלוקים אלא משהו מדורג
כזה:



7

25.10.09

חישוב נומרי

הערות לשיעור הקודם:

- $S(x)$ מהשיעור הקודם לא מוגדר בנקודות $x=x_i$. לא השתמשנו בה בנקודות האלה וזה לא מפריע לנו. היא מוגדרת למקוטעין.
- ראינו ש-

$$f(x)-p(x)=R(x)=\frac{\prod_{i=0}^{M}(x-x_i)}{(M+1)!}f^{(M+1)}(\xi)$$

אנחנו מניחים שהקטע שאנחנו מדברים עליו מוגדר ע"י נקודות הקצה שנתונות לנו. פרט לכך כרגע אנחנו נניח שהמרחקים בין נקודות הדגימה שלנו אחידים ויש δ קב' ש- $x_{i+1}-x_i=\delta$ (אנחנו מניחים שהנקודות ממוינות) ואז $x_M=M\cdot\delta+x_0$.

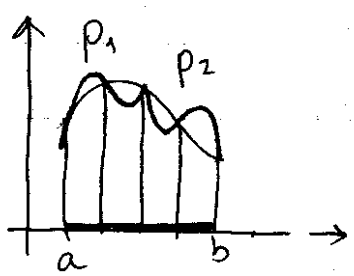
תחת ההנחות האלה מותר לנו להסיק באופן תקין לגמרי ש-

$$|R(x)|\le \delta^{M+1}K$$

כאשר $K=\max_{\xi\in[a,b]}|f^{(M+1)}(\xi)|$. עכשיו הכל תקף!

- לא אוהבים בד"כ לעשות אינטרפולציה מסדרים גבוהים. ואז דיברנו על splines. לגבי יש שתי אלטרנטיבות

I חלוקת הקטע $[a,b]$ לקטעים קטנים ותפירת פולינום בכל קטע.



$p_1(x_0)=y_0 \qquad p_2(x_2)=y_2$

$p_1(x_1)=y_1 \qquad p_2(x_3)=y_3$

$p_1(x_2)=y_2 \qquad p_2(x_4)=y_4$

זה נותן לנו מע' משוואות שהנעלמים שלה הם המקדמים של שני הפולינומים - (c_0^1, c_1^1, c_2^1) ו- (c_0^2, c_1^2, c_2^2) שהמטריצה שלה היא מטריצת בלוקים:

$$\begin{pmatrix}\square & \\ & \square\end{pmatrix}\begin{pmatrix}\vec{c}\end{pmatrix}=\begin{pmatrix}\vec{y}\end{pmatrix}$$

II תפיסה בין תחומי ההגדרה של הפולינומים. טומר, נקודות קרובות ישמשו לתפיסת הפולינום לאותו אזור.

כל פעם שנרצה להעריך בנקודה מסוימת נמצא את הנקודות הקרובות אליה ונעביר פולינום.

לחילופין אפשר לתפור בבת אחת את כל האזורים האפשריים ואז נקבל מטריצת בלוקים. אם כי אם אנחנו צריכים שישמשו להעריך רק נקודה אחת, כנראה לא כדאי לתפור את כולם יחדיו.

על שתי האלטרנטיבות האלה אפשר לתפור גם אם אין תנאים על הנגזרות. אם הנגזרות ידועות מראש אז אפשר פשוט להוסיף אילוצים לכל פולינום, אבל עדיין לכל פולינום יש מס' משוואות בלתי תלויה בפולינומים האחרים ואז אם מסתכלים על הכל ביחד מתקבלת מטריצת בלוקים.

אם אין לנו מידע קודם על הנגזרות נשתמש ב- Natural Splines - משווים בין הנגזרות. מוסיפים למשוואות אילוצים של שוויון הנגזרות.

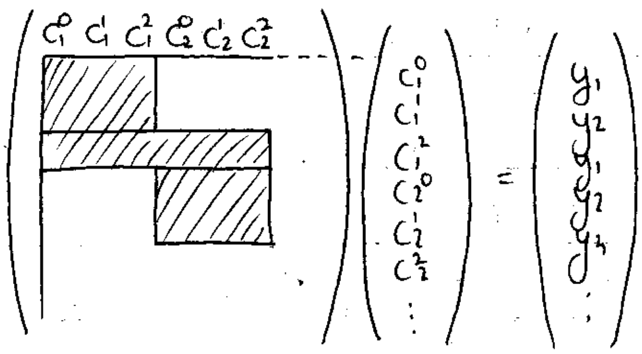
$$p_1(x_0) = y_0 \qquad p_2(x_2) = y_2$$

$$p_1(x_1) = y_1 \longleftrightarrow p_2(x_1) = y_1$$

יוצר רציפות

$$p_1'(x_1) = p_2'(x_1) \longleftarrow$$ מכריח את הנגזרות להסכים

אם הפולינומים ממעלה שניה מתקבלת מטריצה כזאת:



יצא מע' משוואות מסודרת ואותה הרבה יותר קשה לפתור!

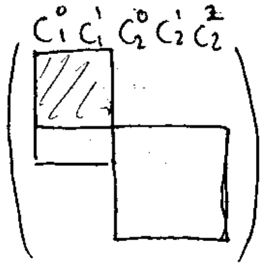
הצעה: נבצע את האינטרפולציה בשלבים. את החלק הכי שמאלי נפתור

כמו שהוא פה לחשוב על הנגזרת. אבל ברגע שעושים את זה
אז יש לנו את ערך הנגזרת בצד ימין שלו. ועכשיו אפשר לתפור
פולינום לקטע הבא והפעם יש לנו ערך ספציפי לנגזרת. וכך נמשיך.
אז יוצא שבקטע פותרים מע' משוואות בבלוקים.

עכשיו יש לנו אילוצים כאלה:

פולינום ממעלה 1
$$\begin{cases} p_1(x_1) = y_1 & p_2(x_2) = y_2 \\ p_1(x_2) = y_2 & p_1'(x_2) = p_2'(x_2) \end{cases}$$

אז מתקבלת מטריצה כזאת:

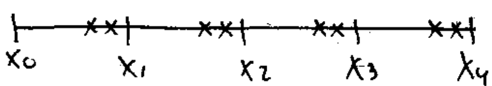


$$\cdot \left(\vec{c} \right) = \begin{pmatrix} y_1 \\ y_2 \\ 0 \\ y_2 \\ y_3 \end{pmatrix}$$

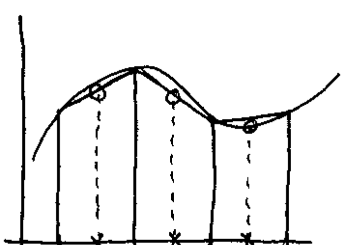
ואת זה יותר קל לפתור כי את השניים הראשונים אפשר לפתור באופן
בלתי תלוי ואז כבר יש לנו אותם בשביל להמשיך. אז למעשה
ה תכנית שתפתור את המטריצה הזאת תהפוך את הבעיה לתהליך סריאלי.

Regular Grids Interpolation

נניח שהנקודות נתונות לנו במרווחים אחידים אבל גם הנקודות שאותן
אנחנו רוצים להעריך תמיד יחתכו את המקטעים באותה צורה.
למשל,



נניח שאנחנו עושים אינטרפולציה בין כל שתי נקודות ע"י פולינום ממעלה 1.



אם למשל הנקודות שרוצים להעריך
נמצאות במרכז הקטעים אז האינטרפולציה היא
ממוצע ערכי y שיש לנו. זו תכונה כללית.
אינטרפולציה לינארית ובפרט פולינומיאלית מייצרת
ערכים חדשים כפונקציה לינארית של הקלט (y_i, x_i).

האינטרפולנט הוא $p(x) = \sum_{i=0}^{M} c_i x^i = \left\langle \begin{pmatrix} 1 \\ x \\ \vdots \\ x^M \end{pmatrix}, \begin{pmatrix} c_0 \\ \vdots \\ c_M \end{pmatrix} \right\rangle = \langle \tilde{x}, \vec{c} \rangle$

כלומר הדגימה ב-p היא מכפלה פנימית ב-$\vec{c}$. אבל $\vec{c} = A^{-1}\vec{y}$

אז $p(x) = \tilde{x}^t A^{-1} \vec{y}$ וזו פונקציה לינארית ב-$\vec{y}$!

אז $p(x)$ שווה לקומבינציה לינארית משקלית של $\vec{y}$.

תחת ההנחות של אחידות נקודות הדגימה החדשות הפעולה היא translation - Invariant. כלומר אם קיבלנו את הווקטור y_i האינטרפולציות החדשות שקיבלנו היו z_i (כרגע אנחנו מניחים שבין כל שני x_i, x_{i+1} יש נקודת הערכה אחת), אז אם נזיז את הקלט ל- y_{i+t}, כלומר אם מוסיפים את הקלט, אז גם הפלט שמתקבל הוא פשוט מוסט ומתקבל z_{i+t}.

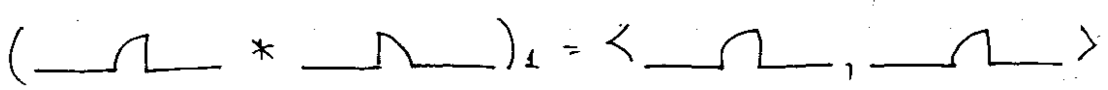
אבל העתקות לינאריות אשר אינווריאנטיות להזזה, ורק הן, מבוטאות ע"י פעולת הקונבולוציה.

עבור שני וקטורים $u, v \in \mathbb{R}^N$ נגדיר את הקונבולוציה $u * v \in \mathbb{R}^N$ ע"י

$$(u * v)_i = \sum_{k=0}^{N-1} u_k v_{(i-k) \bmod N}$$

בהגדרה אנחנו למעשה הופכים את הווקטור v לאינסופי ע"י שרשור שלו וכך האינדקס $i-k$ תמיד חוקי.

למשל,



הקונבולוציה הופכת את הווקטור השני ואז מחשבת את המכפלה הפנימית שלהם עבור כל הסטה.

<u>אבחנה:</u> - אם v קבוע אז הפונקציה $W_v(u) = u * v$ היא לינארית. לכן אפשר לרשום אותה במטריצה

$$\begin{pmatrix} v_1 & v_2 & v_3 & \cdots & \\ & \ddots & \ddots & \ddots & v_3 \\ & & \ddots & & v_2 \\ v_3 & & & \ddots & \\ v_2 & v_3 & & & v_1 \end{pmatrix} \begin{pmatrix} u \end{pmatrix} = \begin{pmatrix} u * v \end{pmatrix}$$

- קונבולוציה היא גם שומרת הזזה.

$$(f * y)_j = \sum_k f_k y_{j-k}$$

$$(f * y_t)_j = \sum_k f_k y_{j-k+t}$$

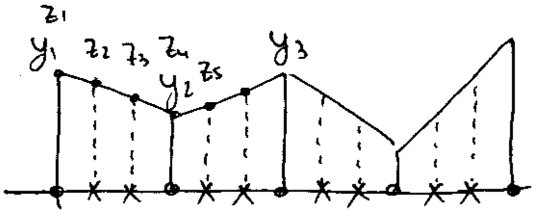
y מוסט ב-t

$$\Rightarrow (f * y_t)_{j-t} = \sum_k f_k y_{j-k} = (f * y)_j$$

וזה בדיוק מה שרצינו.

(9)

איך מראים שכל פונק' לינארית אינווריאנטית להסטה היא קונבולוציה?
היא לינארית ולכן היא למעשה מטריצה. מפעילים את המטריצה
על הבסיס הסטנדרטי וכותבים את האילוצים של אינווריאנטיות
להסטה. אז רואים שמתקבלת מטריצה שהיא המטריצה של קונבולוציה.



דוגמה: עושים up-sampling
אינטרפולציה לינארית מסדר ראשון
שבה מגדילים קצב דגימה פי 3.

אז ברור שמתקיים

$$z_{1+3k} = y_{k+1}$$
$$z_{2+3k} = \frac{2}{3}y_{k+1} + \frac{1}{3}y_{k+2}$$
$$z_{3+3k} = \frac{1}{3}y_{k+1} + \frac{2}{3}y_{k+2}$$

עכשיו אנחנו רוצים לרשום את זה בצורה של קונבולוציה. כלומר,
רוצים למצוא את הפילטר f שיוצר את המצב הזה.
נניח שקצב הדגימה עולה פי 3 אז ה-y שלנו עכשיו הופך
ל- $y \to (y_1\ 0\ 0\ \ y_2\ 0\ 0\ \ y_3\ 0\ 0 \dots)$.
אם הפילטר הוא $f=(a\ b\ c)$ אז הקונבולוציה היא
$$(y_1 a, by_1, cy_1, y_2 a, y_2 b, \dots)$$
ואנחנו רוצים שזה יהיה שווה ל-
$$(y_1, (\tfrac{2}{3}y_1 + \tfrac{1}{3}y_2), (\tfrac{1}{3}y_1 + \tfrac{2}{3}y_2), y_2 \dots)$$
אז הפילטר הוא $(\frac{1}{3}, \frac{2}{3}, 1, \frac{2}{3}, \frac{1}{3})$
זה הגיוני. כל נקודה משפיעה על שתי נקודות מימינה ושתיים
משמאלה. ככל שנקודה קרובה יותר ההשפעה גדולה יותר.

(10) 29.10.09
חישוב נומרי

אינטרפולציה פולינומיאלית של סריגים אחידים

הגדרנו קונבולוציה בין שני וקטורים.

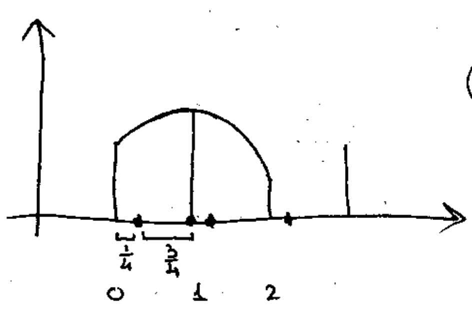
$$(u*v)_i = \sum_{k=0}^{n-1} u_k V_{(i-k)}$$

ונראה איך אפשר לעשות upsampling interpolation.
אנחנו מנסים לתרגם אינטרפולציה פולינומיאלית של סריגים אחידים לפעולת קונבולוציה. יש פה כמה פרמטרים - כמה דגימות אנחנו הולכים להוסיף (באיזה יחס). ואיזה סדר אינטרפולציה.

הנחות:
- מרווחים אחידים בין x_i - הדגימות הנתונות
- מחזוריות של הנקודות החדשות בתוך כל מקטע שמוגדר ע"י נקודות הקלט

מספר הדגימות החדשות משפיע על מספר האפסים שאנחנו הולכים לדחוף בתוך הווקטור y הנתון לנו וסדר האינטרפולציה משפיע על הפילטר f שאיתו אנחנו הולכים לעשות קונבולוציה. הפילטר בעצם אומר כמה כל נקודה מהקלט תורמת לנקודה בפלט. הפילטר לא חייב להיות סימטרי.

דוגמה: נעשה אינטרפולציה ריבועית.
נגדיל את מספר הנקודות פי 2
אם y הופך ל-$(y_0\ 0\ y_1\ 0 \dots)$
נמצא את הפילטר.



$p(0)=y_0$ $\quad p(1)=?$
$p(1)=y_1$ $\quad p(1\frac{1}{4})=?$
$p(2)=y_2$

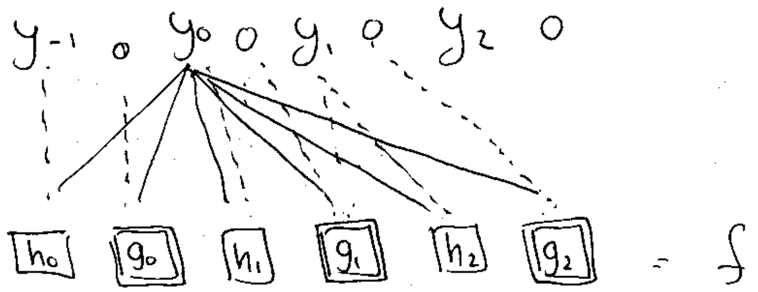
$$p(x) = Ax^2 + Bx + C$$
$$p(0) = C = y_0$$
$$p(1) = A + B + C = y_1$$
$$p(2) = 4A + 2B + C = y_2$$

מפה אפשר למצוא את

$$A(y_0, y_1, y_2)$$
$$B(y_0, y_1, y_2)$$
$$C(y_0, y_1, y_2)$$

$$p(1) = A+B+C = \text{צירוף ליניארי של } y_0, y_1, y_2 \Leftarrow$$
$$= h_0 y_0 + h_1 y_1 + h_2 y_2$$
$$p(1\tfrac{1}{4}) = \frac{25}{16}A + \frac{5}{4}B + C = g_0 y_0 + g_1 y_1 + g_2 y_2$$

המקדמים g_j , h_i הם בלתי תלויים ב-y-ים והם חוזרים על עצמם כל שישייה.



פרקטית זה די מעצבן לעשות את החישובים האלה אז עדיף פשוט לחפש בגוגל - יש אנשים שעשו את זה לפנינו.

Least Squares Approximate Interpolation

יש לנו אוסף נקודות (x_i, y_i) $i=1,\dots,N$ ואנחנו רוצים להתאים אותן באמצעות פולינום שמעלתו קטנה מ-N. לכן סביר שהפולינום לא יתאים באופן מדויק לנקודות אבל אנחנו רוצים פתרון מקורב הכי טוב שאפשר. איך מגדירים הכי טוב? נרצה למזער את השגיאה הריבועית $\sum_{i=1}^{N}(p(x_i) - y_i)^2$

בצורה מטריציונית, אנחנו רוצים למזער לפי $\vec{c}$ את

$$\left[\begin{pmatrix}1 & x_1 & \dots & x_1^M\\ \vdots & & & \\ 1 & x_N & \dots & x_N^M\end{pmatrix}\begin{pmatrix}c_0\\ \vdots\\ c_M\end{pmatrix}-\begin{pmatrix}y_1\\ \vdots\\ y_N\end{pmatrix}\right]^t\left[\underbrace{\begin{pmatrix}1 & x_1 & \dots & x_1^M\\ \vdots & & & \\ 1 & x_N & \dots & x_N^M\end{pmatrix}}_{A}\underbrace{\begin{pmatrix}c_0\\ \vdots\\ c_M\end{pmatrix}}_{\vec{c}}-\underbrace{\begin{pmatrix}y_1\\ \vdots\\ y_N\end{pmatrix}}_{\vec{y}}\right]$$

כלומר מחפשים את $\min_c (A\vec{c}-\vec{y})^t(A\vec{c}-\vec{y})$

$$= \min_c (c^tA^tAc - 2y^tAc + y^ty)$$

את זה אפשר לגזור לפי c - לפי כל קואורדינטה של c -
נקבל מ׳ משוואות ליניאריות - שהרי הפונקציה היא ריבועית.
את זה אפשר לפתור. ואכן נקבל מינימום כי הפונקציה
המקורית היא סכום ריבועים - יכול להיות רק מינימום.

איך גוזרים? נסתכל על הפונקציה הלינארית הכי פשוטה

$$f(c) = \langle v,c\rangle = v^tc$$

$$\Rightarrow \frac{\partial f}{\partial c_i} = v_i = \langle v, e_i\rangle$$

וקטור בסיס סטנדרטי

כעת אם הפונקציה קצת יותר מסובכת $f(c) = Ac$
זה פשוט כמו קודם רק שיש יותר רכיבים לפונקציה. אז

$$\frac{\partial f}{\partial x_j} = Ae_j$$

$$\Rightarrow \nabla f = \left(\frac{\partial f}{\partial x_1}, \dots \frac{\partial f}{\partial x_M}\right) = A$$

אז עכשיו ננסה לגזור את הפונקציה המסובכת שלנו:

$$f(c) = c^tA^tAc - 2y^tAc + y^ty$$

$$\Rightarrow \nabla_c f = 2c^tA^tA - 2y^tA = 0$$

$$\Rightarrow c^tA^tA = y^tA \Rightarrow A^tAc = A^ty$$

ונקבל מע' משוואות שפתירה כל ה-x-ים שונים. ואז ה-c שמתקבל הוא זה שממזער את הבעיה המקורית.
(אגב, הבעיה הזאת נקראת בעיית ה- pseudo inverse)

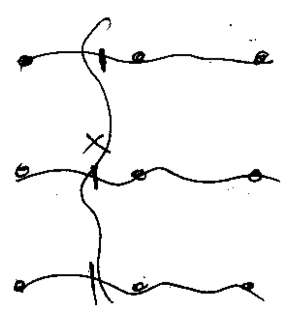
⑦ אינטרפולציה ב- $\mathbb{R}^n$

קלט: $(\vec{x_i}, y_i)$ כאשר $\vec{x_i} \in \mathbb{R}^n$.

אפשרויות:

① בדיוק מה שעשינו ב- $\mathbb{R}$ - לנסות לרשום משוואות עבור פולינום מדרגה מתאימה ולפתור אותן.
פה צריך לשים לב שיש מספר מתאים עבור $M+1$ דרגות חופש. לדוגמא, ל- Ax^2+By^2+Cxy אותה דרגת חופש כמו $Ax+By+Cxy^2$
ולא ברור מה עדיף...

② ממש מה שעשינו בחד-ממד - פשוט כל ציר בנפרד.
לדוגמה בדו-ממד:



קודם נתאים את כל השורות ואז זה יוצר לנו נקודות שאפשר להשתמש בהן כדי לרשום התאמה על העמודה הרצויה.

③ Radial Basis Functions - Scattered Data Interpolation

הרעיון הוא שהנקודות שיותר קרובות לנקודת השערוך שלנו ישפיעו יותר. ההשפעות הן $\phi(r) =$ מונוטונית יורדת ב-r

$$\Phi(\vec{x}) = \sum_{i=1}^{M} w_i \phi(\|x-x_i\|)$$

הפרמטר של האינטרפולציה

ואז צריך לפתור מע' משוואות $\Phi(x_i) = y_i$ לינארית ב- w

(12)

שיטת Shepard –

(תלוי ש- $w_i = y_i$ תחת ההנחה ש- $\phi(0) = 1$

והן דועכות כל כך מהר שכשהן מגיעות לשכנות שלהן

הן כמעט מתאפסות. ואז $y_i \approx \Phi(x_i)$.

וריאנט אחר הוא

$$\Phi(x) = \frac{\sum y_i \phi(\|x - x_i\|)}{\sum \phi(\|x - x_i\|)}$$

וכאן לא צריך את ההנחה ש- $\phi(0) = 1$ וזה מתקיים

כאן הקירוב בטוב הכי.

(13)

11/11/09

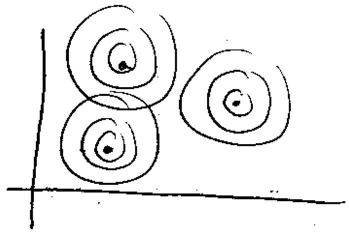
חישוב נומרי

* פורסם תרגיל חדש. את החלקים של ה- Matlab - להגיש את הקוד ואת התוצאות.

Scattered Data Interpolation

ההנחה שהנקודות הגיעו מסריג אחיד היא לא תמיד נכונה וחוץ מזה כשמדובר במימדים גבוהים יש צורך בהרבה מאוד נקודות והסדר של האינטרפולציה יוצא נורא גבוה. יותר מכך לא נהוג להשתמש במונומים כי הם שואפים ל- $\pm\infty$.

אז משתמשים ב- Radial Basis Functions - פונקציות מהצורה $\phi(\|x-x_i\|)$ כאשר x_i נקודת קלט. כלומר משתמשים בפונקציות שמרוכזות סביב נקודות הקלט שלנו ותלויות במרחק מהמרכזים.



האינטרפולציה אז תהיה

$$\Phi(x)=\sum_{i=0}^{M} c_i\phi(\|x-x_i\|)$$

c_i הם הפרמטרים של האינטרפולציה והמשוואות שצריך לפתור כדי לתפור הן $\Phi(x_i)=y_i$.

בד"כ אפשר להניח ש- $\Phi(0)=1$.

משתמשים בד"כ בפונקציות כמו x^{-r}, e^{-x^2} ודומותיהן. שעבורן $\lim_{|x|\to\infty}\phi(x)=0$. אז יוצא שאם נקודה מאוד מרוחקת אז הערך של האינטרפולציה בה הוא 0. זה לא בהכרח נכון אבל בעצם בגלל המרחק שלה אין לנו מספיק מידע כדי לשערך אותה.

אומדים סטטיסטיים לא פרמטריים - חלונות Parzen

יש אוסף תצפיות - רק x_i-ים שאין להם y_i-ים. אז אפשר לדבר על מה הסבירות לקבל תצפית מסוימת. אז נהוג להשתמש ב- $\Phi(x)=\sum_i\phi(\|x-x_i\|)$ וזה איזשהו מדד להסתברות. כדי לקבל הסתברות ממש צריך לנרמל רק שהאינטגרל יהיה 1 אבל זה לא ממש משנה כי בד"כ מעניין להשוות מה יותר סביר ממה ואז הנרמול לא חשוב.

שיטת Shepard -

האינטרפולציה

$$\Phi(x) = \frac{\sum_{i=0}^{M} y_i \, \phi(\|x-x_i\|)}{\sum_{i=0}^{M} \phi(\|x-x_i\|)}$$

* כאן אין צורך בהערכת הפרמטרים c_i.
* כל ערך חדש $\Phi(x)$ הוא קומבינציה קמורה של y_i [כלומר המקדמים של הקומבינציה אי-שליליים וסכומם 1].
* בגלל שקומבינציה קמורה נותנת שלכל x

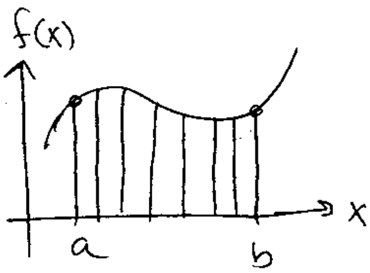
$$\min y_i \le \Phi(x) \le \max y_i$$

אז כאן אין דעיכה ל-0.

אינטגרציה נומרית

נתונה פונקציה $f(x): \mathbb{R} \to \mathbb{R}$ ורוצים להעריך את $\int_a^b f(x)\,dx$

למה להעריך? לא תמיד קיימת פונקציה קדומה אלמנטרית והרבה מקרים אם גם היא קיימת לא קל לחשב אותה...



שלבים:

① חלוקת תחום האינטגרציה למקטעים קטנים באורך h.

② קירוב הפונקציה f בכל קטע ע"י פונקציה פשוטה לאינטגרציה....

③ סיכום כל האינטגרלים בכל קטע

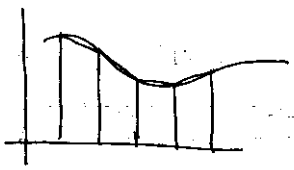
הנחות:

① לכל i $|x_{i+1} - x_i| = h$

זה קצת מזכיר את הגדרת האינטגרל באמצעות סכומי רימן. שם הקירובים היו מסדר 0 וחוסר הדיוק לא הפריע לנו כי בסוף הרי השאפנו את h ל-0. אנחנו לא יכולים להשאיף את h ל-0 אז נפצה על

כך באמצעות קירוב אם יותר של הפונקציה במקטעים שונים.

קירוב מסדר ראשון - שיטת הטרפז



$$\int_{x_i}^{x_{i+1}} p(x)\,dx = h\left(\frac{y_{i+1}-y_i}{2}\right)$$

הערכת שגיאה של אינטגרציה נומרית עם אינטרפולציה פולינומיאלית

יש שתי דרכים:

① שימוש בשגיאה הנקודתית של אינטרפולציה לגרנג'

② שימוש באנליזת אוילר כדי לקבל הערכה מדוייקת יותר.

נתחיל בשיטה הפשוטה יותר. ראינו שפולינום ממעלה M נותן שגיאה נקודתית $R(x)$ שחסומה ע"י

$$|R(x)| \le h^{M+1}\cdot\max_{\xi\in[x_0,x_M]}\left|f^{(M+1)}(\xi)\right|$$

אנו מעוניין לחסום את $\left|\int_{x_i}^{x_{i+M}}\underbrace{(f(x)-p(x))}_{"R(x)"}dx\right|$

$$\left|\int_{x_i}^{x_{i+M}}(f(x)-p(x))\,dx\right| \le \int_{x_i}^{x_{i+M}}|f(x)-p(x)|\,dx =$$

$$= \int_{x_i}^{x_{i+M}}|R(x)|\,dx \le \int_{x_i}^{x_{i+M}} h^{M+1}\underbrace{\max_{\xi\in[a,b]}\left|f^{(M+1)}(\xi)\right|}_{=K}dx =$$

$$= h\cdot M\cdot h^{M+1}K = M\cdot h^{M+2}\cdot K$$

$$\Rightarrow \left|\int_a^b (p(x)-f(x))\,dx\right| \le \sum_{\text{קטעים}}\int_{x_i}^{x_{i+M}}|p(x)-f(x)|\,dx \le$$

$$\le \sum_{\text{קטעים}} M\cdot h^{M+2}\cdot K = \frac{b-a}{\cancel{M}h}\cdot h^{M+2}\cdot K\cdot\cancel{M} = (b-a)h^{M+1}\cdot K$$

אז למשל בשיטת הטרפז השגיאה היא $O(h^2)$.

ניתוח באמצעות פולינומי טיילור ירו) את לנו תסמים יותר הדוקים.

תזכורת: קירוב טיילור לפונקציה f גזירה ברציפות $M+1$ פעמים:

לכל a קיימת סביבה $[a, a+\delta)$ שבה מתקיים לכל x

$$f(x) = f(a) + f'(a)(x-a) + \dots + f^{(M)}(a)\frac{(x-a)^M}{M!} + \frac{f^{(M+1)}(\xi)}{(M+1)!}(x-a)^{M+1}$$

עבור $\xi \in [a, a+\delta)$ כלשהו.

שיטת Simpson

אינטגרציה נומרית בעזרת פולינומים ממעלה שנייה.

מהערכה שבאה מאינטרפולציה נקודתית של פולינומים צריך שהקירוב הזה מדויק כמו $O(h^3)$.

בעזרת אנליזת טיילור נראה שהוא למעשה $O(h^4)$

$$\int_{x_{i-1}}^{x_{i+1}} f(x)dx \approx h\left[\frac{1}{3}y_{i-1} + \frac{4}{3}y_i + \frac{1}{3}y_{i+1}\right]$$

זה הפסוק מה שיוצא משיטת סימפסון

...

נניח בה"כ ש- $x_i = 0$. אחרת תמיד אפשר להזיז את מע' הצירים.

$$\int_{x_{i-1}}^{x_{i+1}} f(x)dx = \int_{-h}^{h} f(x)dx = \int_{-h}^{h}\left(f(0) + f'(0)x + \frac{f''(0)x^2}{2} + \frac{f'''(0)x^3}{6} + O(h^4)\right)dx$$

פיתוח טיילור

$$= 2hf(0) + 0 + \frac{h^3}{3}f''(0) + 0 + O(h^5)$$

x זו פונק' אי-זוגית

אינטגרל על קטע בגודל $2h$ של פונקציה שחסומה ע"י h^4.

בשיעור הבא: הקשר בין זה לשיטת סימפסון.

(15)

5/11/09

חישוב נומרי

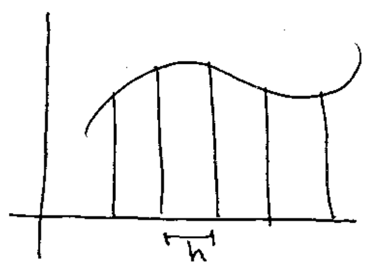
הבהרה לתרגול 1: שאלה 3א':

האינדקס k הוא פשוט הארגומנט של f ; ה-i הוא פשוט משמש שיש הרבה y_i-ים קבועים. בכל מקרה האינדקס הרל של f ושל ל הוא k וה-i פשוט משמש ריבוי אפשרויות. אנחנו אמורים להראות עבור i נתון קבוע שלמו, d הוא קבוע הניפות.

j הוא כמובן האינדקס של ה-output.

אינטרפולציה נומרית בעזרת Newton-Cotes

אינטרפולציה פולינומיאלית



מקרבים את הפונקציה בכל מקטע ע"י פולינום והרי פולינום זו פונקציה שאנחנו יודעים לעשות לה אינטגרל.

מהי שגיאת הקירוב כאן? ראינו בשבועות האחרונים הערכת שגיאה נקודתית של האינטרפולציה:

$$|R(x)| \le \max_{\xi\in[a,b]} |f^{(M+1)}(\xi)|\cdot h^{M+1}$$

ומכאן אפשר לחסום את השגיאה של האינטגרציה:

$$\left|\int_{x_i}^{x_{i+M}} f(x)dx - \int_{x_i}^{x_{i+M}} p(x)dx\right| \le$$

$$\int_{x_i}^{x_{i+M}} |f(x)-p(x)|dx = \int_{x_i}^{x_{i+M}} |R(x)|dx \le$$

$$\le \int_{x_i}^{x_{i+M}} h^{M+1}\cdot\max_{\xi\in[a,b]}|f^{(M+1)}(\xi)|\,dx =$$

$$= M\cdot h\cdot h^{M+1}\cdot K = h^{M+2}MK$$

$$\Rightarrow \left|\int_a^b f(x)dx - \underbrace{\int_a^b p(x)}_{\text{שרשור של פולינומים}}\right| = \left|\sum_i \int_{x_i}^{x_{i+M}} f(x)dx - \int_{x_i}^{x_{i+M}} p(x)dx\right| \le$$

$$\le \underbrace{K h^{M+1} \cdot M \cdot h}_{\text{חסם לכל קטע}} \ \underbrace{\frac{b-a}{Mh}}_{\text{מספר הקטעים}} = K h^{M+1}(b-a)$$

אז גם פה קיבלנו חסם שהוא $O(h^{M+1})$ - כמו באינטרפולציה. אבל זה לא אומר שזה חסם הדוק.

נשתמש באנליזת טיילור כדי לנתח את שיטת simpson $(M=2)$

$$(*) \quad \int_{x_i}^{x_{i+2}} f(x)\,dx \approx h\left[\frac{1}{3}y_i + \frac{4}{3}y_{i+1} + \frac{1}{3}y_{i+2}\right]$$

[קל לראות שזה נכון אם f פונקציה קבועה ואם f לינארית)
האנליזה למעלה אומרת ששיטת סימפסון היא מדויק $O(h^3)$.
אז נעבור לאנליזת טיילור.
בה"כ $x\in[-h,h]$ (הזזה ימינה או שמאלה לא משנה את האינטגרל של הפונקציה).

$$\int_{-h}^{h} f(x)\,dx = \int_{-h}^{h}\left(f(0) + f'(0)x + \frac{1}{2}f''(0)x^2 + \frac{1}{6}f'''(0)x^3 + o(h^4)\right)dx$$

$$= 2h f(0) + \frac{h^3}{3}f''(0) + o(h^5)$$

אבל את זה קשה לנו להשוות ל-(*), כי ב-(*) מדובר על הערכות של f ולא של הנגזרות.
איך מחברים תפוזים ותפוחים? מה מקשר בין ערכים לנגזרות?
טיילור!

$$f(h) = f(0) + f'(0)h + \frac{f''(0)}{2}h^2 + \frac{f'''(0)}{6}h^3 + o(h^4)$$

$$f(-h) = f(0) - f'(0)h + \frac{f''(0)}{2}h^2 - \frac{f'''(0)}{6}h^3 + o(h^4)$$

$$\Rightarrow h\left(\frac{1}{3}y_i + \frac{4}{3}y_{i+1} + \frac{1}{3}y_{i+2}\right) =$$

$$= h\left(\frac{4}{3}f(0) + \frac{1}{3}\left(f(h) + f(-h)\right)\right) =$$

$$= h\cdot\left(\frac{4}{3}f(0) + \frac{1}{3}\left(2f(0) + f''(0)h^2 + o(h^4)\right)\right) =$$

$$= \frac{4h}{3}f(0) + \frac{2h}{3}f(0) + \frac{f''(0)}{3}h^3 + o(h^5)$$

$$= 2h f(0) + \frac{f''(0)}{3}h^3 + o(h^5) = \int_{-h}^{h} f(x)\,dx$$

מה שחישבנו קודם

(16)

אז שיטת סימפסון נותנת קירוב לאינטגרל שהוא טוב כמו הקירוב טיילור מקטע עד כדי $O(h^5)$. וטיב קירוב טיילור הוא $O(h^4)$. אז סה"כ טיב קירוב האינטגרל הוא $O(h^4)$.

$$\int_{-h}^{h} f(x)dx = \underbrace{\int_{-h}^{h}\left(f(0)+f'(0)x+\frac{f''(0)x^3}{6}\right)dx}_{= h\left[\frac{1}{3}f(-h)+\frac{4}{3}f(0)+\frac{1}{3}f(h)\right]} + O(h^5)$$

$$= h\left[\frac{1}{3}f(-h)+\frac{4}{3}f(0)+\frac{1}{3}f(h)\right]+O(h^5)$$

$$\Rightarrow \int_{-h}^{h} f(x)dx = h\left[\frac{1}{3}f(-h)+\frac{4}{3}f(0)+\frac{1}{3}f(h)\right]+O(h^5)$$

אז על כל מקטע יש לנו חסם $O(h^5)$.
עכשיו אם נסכום על כל המקטעים נקבל $\frac{b-a}{Mh}\cdot O(h^5)=O(h^4)$. אז הצלחנו תאורטית חסם יותר הדוק על השיטה של סימפסון.

אז אנליזה נקודתית לא מוצדקת לעובדה שבאיברים אי זוגיים של פיתוח טיילור מתאפסים מתחת לאינטגרל. אנחנו בינתיים עושים פירוק לפי בסיס של הפונקציה ואז אפילו אם יש שגיאה בקירוב של החלק האי זוגי, האינטגרל מתאפס ואז באינטגרל יש פחות שגיאה.

אינטגרציה ב-$\mathbb{R}^n$

כמו באינטרפולציה ב-$\mathbb{R}^n$, ישנן שתי אפשרויות:

1. להתאים פולינום רב-ממדי ולחשב את האינטגרל ממנו.
2. שימוש במשפט פוביני שמאפשר לנו להתייחס לבעיה בשלבים.

$$\int_A f(x,y)dxdy = \int_x \int_y f(x,y)\,dy\,dx$$

במקרה זה אפשר גם להחיל את אנליזת השגיאה שעשינו במקרה החד-ממדי (נראה זאת בתרגול).

מה קורה כאשר N גדול במיוחד?

בחד-ממד ראינו שכדי להשיג דיוק של $O(h^m)$ צריך להשקיע $O(h^{-1})$ עבודה.

במימד d היחס הזה הוא אותו יחס. כדי שתהיה לנו דגימה אחידה במרחקים h של כל ציר צריך $N = O(h^{-d})$ נקודות. והשגיאה היא $E = O(h^M)$. אז $h = N^{-1/d}$

$\Leftarrow$ $E = N^{-\frac{M}{d}}$. אז כש-M גדל החסם קטן וכש-d גדל החסם גדל. בפרקטיקה M בד"כ חסום ע"י כ-8 (כי יש אילוצי חישוב) ואילו d יכול להיות גדול מאוד, אפילו 60.

איך מתגברים על זה?

Monte Carlo Integration

$$I = \int_a^b f(x)dx = \int_a^b \frac{f(x)}{g(x)} g(x)\, dx \approx \frac{1}{N}\sum_{i=1}^{N} \frac{f(x_i)}{g(x_i)}$$

$g > 0$, $\int g(x) = 1$; $x_i \sim g(x)$ בת"ל

זה בעצם חישוב תוחלת של $\frac{f}{g}$ לפי התפלגות g.

כאן השגיאה היא משתנה אקראי, כי לכל הגרלה של דגימות נקבל שגיאה אחרת. לכן נרצה לדבר על תוחלת השגיאה.

$$\mathbb{E}_g\left[\frac{1}{N}\sum_{1}^{N}\frac{f(x_i)}{g(x_i)}\right] = \frac{1}{N}\sum_{1}^{N}\mathbb{E}_g\left[\frac{f(x_i)}{g(x_i)}\right] =$$

$$= \frac{1}{N}\sum_{1}^{N}\int_a^b \frac{f(x)}{\cancel{g(x)}}\cancel{g(x)}\,dx = \frac{1}{N}\sum_{1}^{N}\int_a^b f(x)\,dx =$$

$$= \int_a^b f(x)dx = I$$

אז הקירוב שלנו טוב. והשגיאה

$$\mathbb{E}_g\left(\left[\frac{1}{N}\sum\frac{f(x_i)}{g(x_i)} - \underbrace{I}_{\mathbb{E}_g[\,]}\right]^2\right) = Var_g\left[\frac{1}{N}\sum\frac{f(x_i)}{g(x_i)}\right] =$$

$$= \frac{1}{N^2} V_g\left(\sum\frac{f(x_i)}{g(x_i)}\right) = \frac{1}{N} Var\left(\frac{f(x)}{g(x)}\right)$$

משתנים עם אותה התפלגות בת"ל

אז יוצא לנו ש-

$$E_M = \frac{1}{\sqrt{N}} \cdot STD\left(\frac{f}{g}\right)$$

כש- $\frac{f}{g}$ לא משתנה הרבה הקבוע $STD\left(\frac{f}{g}\right)$
הוא קטן. כלומר צריך ש- $\frac{f}{g} = const$. אז הכי טוב
קורה כש- $g(x) = \frac{f(x)}{\int f(x)dx}$ אבל $\int f dx$ זה מה
שאנחנו מנסים לחשב. הבעיה...
אז אנחנו ננסה למצוא g שכמה שיותר דומה ל- f
g נקראת trial distribution או
Importance Sampling - איפה שחשוב - שם דוגמים.

(18) 8/11/09
חישוב נומרי

הערה לתרגול שאלה 3ב':
מה שבאמת צריך להוכיח זה

$$(f_k * y_i \delta_{k-id})_j = y_i f_{j-kd}$$

בשיעור הקודם דיברנו על אינטגרציה נומרית ב-$\mathbb{R}^d$ כאשר d גבוה.
ראינו ששיטות Newton-Cotes שבהן השגיאה מתנהגת כמו $E \sim N^{-\frac{M}{d}}$ לא טובות כ"כ כשסדר המימד עולה ואז צריך הרבה מאוד נקודות ולקרב בסדרים מאוד גבוהים.
גם מסיבות פרקטיות רוצים להימנע משימוש בפולינומים ממעלה גבוהה (זה קשה חישובית) וכן חסם השגיאה תלוי ב- $f^{(M+1)}$ שיכולה לגדול מאוד כאשר M גדל.
אז דיברנו על שיטת Monte-Carlo:

$$I = \int_a^b f(x)\,dx \approx \frac{1}{N}\sum_{i=1}^{N} \frac{f(\vec{x}_i)}{g(\vec{x}_i)}$$

$$\downarrow$$

$$x_i \overset{iid}{\sim} g, \quad g>0, \quad \int g = 1$$

וראינו $E \sim N^{-\frac{1}{2}}$, כלומר $M = \frac{d}{2}$ שזה מאוד נחמד!
השיטה הזאת נשמעת טוב מאוד מבחינת השגיאות אבל הבעיה העיקרית שלה היא למצוא g טובה ואחר מכן דגימה iid אמינה (שזו בעיה לא טריוויאלית בממדים גבוהים).

נניח ש-g חד-ממדית. איך דוגמים אמינה iid?
ההנחה העיקרית שלו (והיא אכן לגיטימית) היא שיש לנו גישה למשתנה אקראי מתפלג אחיד בקטע $[0,1]$ - $u \sim U[0,1]$
אם אנחנו רוצים איכשהו לייצר דגימות $x(u) \sim g$ כלומר רוצים ש-

$$P(y < x(u) < y+\varepsilon) = \int_y^{y+\varepsilon} g(s)\,ds$$

נניח ש-x הפיכה ומונוטונית עולה. אז גם

$$P(x^{-1}(y) < u < x^{-1}(y+\varepsilon)) = \int_y^{y+\varepsilon} g(s)\,ds$$

נשאיף את ε ל-0 ונתעלם מהאיברים שהם $O(\varepsilon^2)$.

$$P\left(X^{-1}(y) < u < X^{-1}(y) + \varepsilon (X^{-1})'(y)\right) = \varepsilon \cdot g(y)$$

אבל u מתפלג אחיד ולכן

$$\varepsilon g(y) = P\left(X^{-1}(y) < u < X^{-1}(y) + \varepsilon (X^{-1})'(y)\right) = \varepsilon (X^{-1})'(y)$$

$$\Rightarrow \varepsilon g(y) = \varepsilon (X^{-1})'(y)$$

$$\Rightarrow (X^{-1})'(y) = g(y)$$

$$\Rightarrow X^{-1}(y) = \int_{-\infty}^{y} g(s)\,ds$$

אבל $g>0$ ולכן $X^{-1}(y)$ מונוטונית עולה ממש. אז יש לה הופכית וההופכית מונוטונית עולה. אז כן הייתה הנחה לגיטימית מלכתחילה.

כמובן, אנחנו מקווים ש- g היא כזאת שאנחנו יודעים לחשב לה את האינטגרל או שיש לנו טבלאות...

כדי לעבור לרב-ממד אפשר להניח $g(x_1 \dots x_d) = \prod_{i=1}^{d} g_i(x_i)$ ...

ואז לעשות את התהליך בנפרד לכל g_i. מכפלה של פונק' התפלגות היא פונקציית התפלגות אז זה מסדר.

הסיבה שזה לא עובד טוב כל כך בממדים גבוהים היא שאיכות הייצור של $X \sim g$ יורדת...

שיטה שעושה התמודדות עם זה היא MC Monte-Carlo (Markov Chain) שם יוצא שיש תלות סטטיסטית בין הדגימות ולכן צריך הרבה מאוד דגימות. (קורס בסמסטר הבא)

סיכום:

- יודעים לדגום משתנים מקריים מהתפלגות מסוימת g בחד-ממד.
- עבור $f(x), g(x)$ רב-ממדית יש יכולת לדגום $g(x) = \prod g_i(x_i)$ מודלים בעלי רכיבים בלתי תלויים או שישנם מודלים רב-ממדיים שאותם אנחנו יודעים לדגום (למשל התפלגות נורמלית, אחידה וכו').

Rejection Sampling:

$$\int_D f(x)\,dx = \int_{[0,1]^2} \mathbb{1}_D(x) f(x)\,dx$$

$$D \subseteq [0,1]^2$$

כאשר $\mathbb{1}_D(x) = \begin{cases} 1 & x\in D \\ 0 & x\notin D \end{cases}$

אז עושים פה את אותו הדבר רק קודם רק שלפני שמכניסים נקודה לסכום בודקים אם $x\in D$ או לא. זה יכול לעזור כשיש לנו פונקציה שאנחנו יודעים לאינטגרל אבל אין לנו פרמטריזציה טובה של D.

פת׳ משוואות לא לינאריות - n משוואות ב- n נעלמים

יש פונקציה f ורוצים למצוא את השורשים שלה $f(x)=0$.

Bisection Method

מניחים ש- f רציפה ושיש ניחוש התחלתי - נקודות a,b כך ש- $f(a)\cdot f(b)<0$.

הרעיון פשוט עושים חיפוש בינארי - לפי משפט ערך הביניים בהכרח f מקבלת 0 איפשהו בין a ל- b, אז פשוט מחפשים.

$a_0 = a$

$b_0 = b$

$c_n = \frac{a_n+b_n}{2}$

if $f(c_n)>0$

  $b_{n+1} = c_n,\quad a_{n+1} = a_n$

else if $f(c_n)<0$

  $b_{n+1} = b_n,\quad a_{n+1} = c_n$

else

  $f(c_n)=0$ and we're done.

זאת שיטה איטרטיבית והיא משפרת את הקירוב בכל צעד.

מה קצב ההתכנסות?

נניח ש- r פתרון, כלומר $f(r)=0$ אז $a_n<r<b_n$
לכל n. מהשיטה שלנו נובע ש-

$$e_n = |c_n - r| \le |b_n - a_n| = \frac{b-a}{2^n}$$

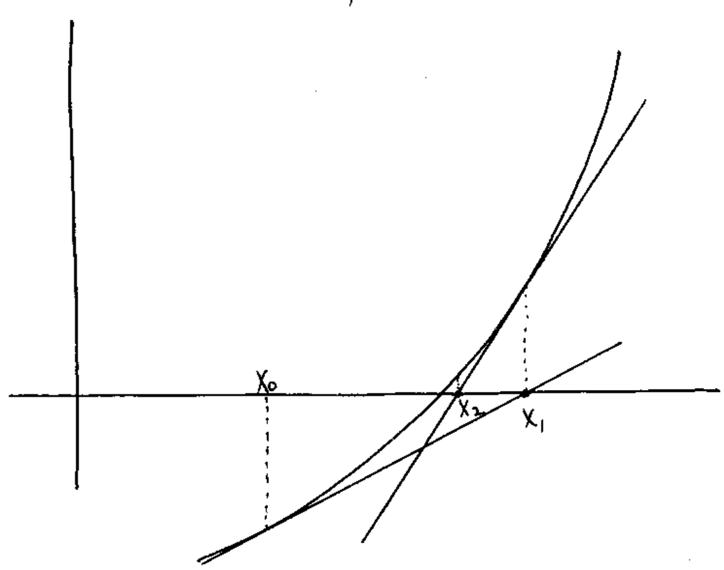
$\Leftarrow$ $\frac{e_{n+1}}{e_n} \le \frac{1}{2}$ וזה נקרא קצב סופר-לינארי (הלינאריות מתייחסת רק שה-n מופיע בחזקה לינארית).

קל להוכיח (ממשפט הסנדביץ') ש- $f(c_n) \to 0$ אז באמת מקבלים את התוצאה שצריך וקצב ההתכנסות הוא סופר-לינארי.

למרבה הצער אי אפשר להרחיב את השיטה הזאת לרב-ממד, כי שם אין דרך פשוטה לתחום את התחום שבו השורש נמצא.

שיטת Newton-Raphson

נניח ש- f גזירה. לצורך יעילות השיטה נניח גם ש- f'' קיימת ורציפה. וגם לנו צריך ניחוש התחלתי שהוא נחמד.
אפשר לא מובטחת התכנסות שזה פחות נחמד.
אבל כשכן יש התכנסות אז זה קורה מאוד מהר.



איטרטיבית, נקרב את הפונקציה ע"י פונקציה לינארית. נבדוק את הערך של f בנקודת ההתאפסות של הקירוב. אם זה מתאפס סיימנו. אחרת ממשיך באותו אופן מאותה הנקודה.

מתחילים מ- x_0 נתון.

בשלב ה- n הקירוב הלינארי הוא $\hat{f}(x) = f(x_n) + f'(x_n)(x-x_n)$

ואז $\hat{f}(x)=0$ כאשר $x_{n+1} = x = x_n - \frac{f(x_n)}{f'(x_n)}$ ומשם ממשיכים.

(20)

* נשים ♡ שאנחנו לא רוצים ש- $f'(x_n)=0$ כי אז אנחנו לא יכולים להמשיך. זה ה- pitfall העיקרי כאן.

הערכת השגיאה: (נניח ש-r הוא הפתרון)

$$e_{n+1} = x_{n+1} - r = x_n - \frac{f(x_n)}{f'(x_n)} - r =$$
$$= \underbrace{x_n - r} - \frac{f(x_n)}{f'(x_n)} = e_n - \frac{f(x_n)}{f'(x_n)} =$$
$$= \frac{f'(x_n)e_n - f(x_n)}{f'(x_n)} = ?$$

נעשה רגע פיתוחי טיילור:

$$0 = f(r) = f(x_n) + f'(x_n)(r-x_n) + \frac{f''(\xi_n)}{2}(r-x_n)^2 =$$
$$= f(x_n) - f'(x_n)e_n + \frac{e_n^2}{2} f''(\xi_n)$$
$$\Rightarrow f'(x_n)e_n - f(x_n) = \frac{e_n^2}{2} f''(\xi_n)$$
$$\Rightarrow e_{n+1} = \frac{e_n^2}{2} f''(\xi_n) / f'(x_n)$$

אז כן התכנסות סופר-ריבועית. כדי שהתהליך הזה יתכנס אנחנו דורשים ש-

$$1 > \frac{e_n}{2} \cdot \frac{f''(\xi_n)}{f'(x_n)}$$

אז אם יש לנו איזה חסמים על f' ועל f'' אז אם הניחוש ההתחלתי שלנו הוא טוב אז זה מתכנס וגם סופר-ריבועית, כלומר יותר מהר מהשיטה הקודמת.

בתד-ממד, אפשר להתווכח על מה עדיף- ניוטון או חיפוש בינארי? אבל היתרון המשמעותי של שיטת ניוטון- רפסון הוא שניתן להרחיב את השיטה להרבה ממדים.

יש לנו פונקציה $f: \mathbb{R}^n \to \mathbb{R}^n$ ואנחנו רוצים למצוא $x \in \mathbb{R}^n$ כך ש- $f(x)=0$. ניתן לרשום $f(x) = (f_1(x), \dots, f_n(x))$ כאשר $f_i: \mathbb{R}^n \to \mathbb{R}$.

הרחבת השיטה היא די ישירה -

$$\hat{f}(x) = f(x_n) + \nabla f(x_n)(x - x_n)$$

מה זה $\nabla f(x_n)$?

$$f = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} \Rightarrow \nabla f = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}, & \dots, & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \\ \frac{\partial f_n}{\partial x_1}, & \dots, & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

נשווה לאפס ונקבל

$$\Rightarrow \quad x_{n+1} = x = x_n - \left(\nabla f(x_n)\right)^{-1} f(x_n)$$

וכאן, כמו שבחד ממד רצינו ש- $f'(x_n) \neq 0$ פה אנחנו צריכים ש- $\nabla f(x_n)$ הפיכה.

אז כל הדברים האלה מגבילים את הפונקציות f שאנחנו יכולים לעבוד איתן.

בשיעור הבא נדבר על ההתכנסות.

12.11.09

חישוב נומרי

Rejection Sampling - נועד לטפל בתחומי אינטגרציה מורכבים D. אז נניח שיש לנו תחום יותר פשוט $D \subseteq U$. אז

$$\int_D f(x)dx = \int_U f(x)\mathbb{1}_D(x)\,dx$$

ועל האינטגרל החדש אנחנו רוצים להפעיל את שיטת מונטה-קרלו.

$$\int_U f(x)\mathbb{1}_D(x)\,dx = \int_{\mathbb{R}^n} \left(f(x)\mathbb{1}_D(x)\right) \frac{\mathbb{1}_U(x)}{|U|}\,|U|\,dx$$

$$\approx \frac{|U|}{N}\sum f(x_i)\mathbb{1}_D(x)$$

$$\Downarrow$$

$$x_i \sim \frac{\mathbb{1}_U(x)}{|U|}$$

את $|U|$ קל לחשב כי בחרנו U כזה פשוט שאנחנו יודעים לעשות עליו אינטגרלים ובפרט לחשב את הנפח שלו. (למשל תיבה חוסמת של D).
נזכור שבשיטת מונטה-קרלו אנחנו רוצים ש-f תהיה כמה שיותר קרובה ל-g. לכן נעדיף ש-U יהיה כמה שיותר הדוק ל-D, כי אנחנו לא רוצים לבזבז דגימות שבהן f מתאפסת - שזה מה שקורה מחוץ ל-D.

אנליזת שגיאה של שיטת ניוטון ב-$\mathbb{R}^N$. הניתוח הרב-ממדי דומה מאוד למה שעשינו בחד-ממד. קיבלנו

$$x_{n+1} = x_n - (\nabla f(x_n))^{-1} f(x_n)$$

אז

$$e_{n+1} = (\nabla f(x_n))^{-1} \begin{pmatrix} e_n^t H_{f_1} e_n \\ \vdots \\ e_n^t H_{f_N} e_n \end{pmatrix} \leq$$

חישוב פשוט שנראה בשיעור הבא.

כאשר

$$H_{f_i} = \begin{pmatrix} \frac{\partial f_i}{\partial x_1^2} & \frac{\partial f_i}{\partial x_1 x_2} & \cdots & \frac{\partial f_i}{\partial x_1 \partial x_N} \\ \vdots & & & \\ \frac{\partial f_i}{\partial x_N \partial x_1} & \cdots & & \frac{\partial f_i}{\partial x_N^2} \end{pmatrix}$$

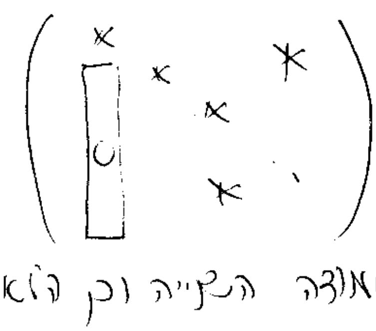
אם f גזירה פעמיים ברציפות אז ניתן להחליף את סדר הגזירה ולכן H_{f_i} סימטרית.

מע' משוואות לינאריות $Ax=b$

שיטות ישירות (direct methods) מבצעות מספר סופי של פעולות חשבוניות שבסופן מתקבל הפתרון המדויק (עד כדי roundoff).
שיטות איטרטיביות מתקרבות לפתרון בשלבים אבל בגבול ובזמן סופית (אולי רק במקרים מאוד ספציפיים) אין פתרון מדויק.

השיטה הישירה הכי בסיסית היא שיטת האלימינציה של גאוס.
מבצעים אוסף פעולות על המטריצה שמשמרות את מרחב הפתרונות ובסופו של דבר מקבלים מטריצה מדורגת. את הפעולות עושים על המטריצה ועל הווקטור b ובסוף אפשר להפוך את A למטריצת היחידה ואז מ-b נקבל את הפתרון.
הפעולות הן:
- החלפת שורות.
- כפל שורה בסקלר שונה מאפס
- הוספת שורה לשורה אחרת.

אפשר להוכיח שהפעולות האלה משמרות את הפתרונות.
איך נהפוך את המטריצה למטריצה פשוטה יותר?
מההתחלה ננסה לאפס את כל האיברים שמתחת לאיבר הראשון בשורה הראשונה. זה נעשה ע"י הכפלת השורה הראשונה בפקטורים מתאימים והחסרה מהשורות האחרות. אז נקבל



ועכשיו נעשה אותו הדבר לעמודה השנייה וכן הלאה עד שמתקבלת מטריצה משולשת עליונה. ואז מאפסים את כל מה שמעל האלכסון ונשארים עם מטריצה אלכסונית. מפה הכל ברור.

את האלגוריתם הזה אפשר להדגים בדרך אלגברית כ פעולה
אלמנטרית אפשר לבצע ע"י הכפלת A במטריצה מסוימת מצד שמאל.
ואז ישנו תהליך הרכבה שבסופו

$$x = Ix = \underset{\text{פעולה אחרונה}}{(\ )}(\ )\dots\underset{\text{פעולה 1}}{(\ )}Ax = (\ )(\ )\dots(\ )b$$

מה הסיבוכיות של השיטה הזאת? $O(n^3)$ - היינו צריכים
לאפס $O(n^2)$ איברים. לכל איבר כזה היינו צריכים לעשות
$O(n)$ פעולות (כי אלה פעולות על שורות). לכן סה"כ $O(n^3)$.

LU Factorization

נתונה משוואה $Ax=b$. נפרק את A למכפלה

$$A = \underset{\text{משולשית תחתונה}}{L}\ \underset{\text{משולשית עליונה}}{U} = (\triangle)(\nabla)$$

$L_{ij}=0$, $i<j$ $\qquad$ $U_{ij}=0$, $j<i$

למה זה טוב? כי עכשיו יש לנו מערכת משוואות $L(Ux)=b$
ואת זה קל לפתור ע"י הצבה לאחור. קודם נמצא את Ux
ע"י פתרון $Ly=b$ ואז נפתור את $Ux=y$.
את המשוואות האלה קל לפתור ע"י הצבה לאחור.
זה פשוט נעשה באינדוקציה

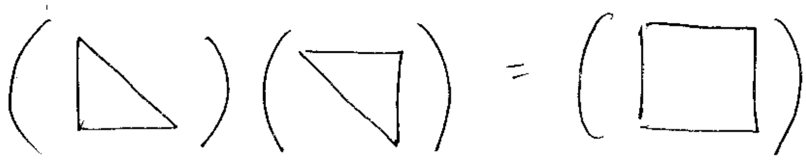
$$(\triangle)\,y = b \implies L_{11}y_1 = b_1 \implies y_1 = \frac{b_1}{L_{11}}$$

$$O(n^2) \qquad y_i = \frac{b_i - \sum_{j<i} L_{ij}y_j}{L_{ii}} \qquad i=1,\dots,n$$

$$(\nabla)\,x = y \implies U_{nn}x_n = y_n \implies x_n = \frac{y_n}{U_{nn}}$$

$$O(n^2) \qquad x_i = \frac{y_i - \sum_{j>i} U_{ij}x_j}{U_{ii}} \qquad i=n,\dots,1$$

אבל איך מחשבים את הפירוק?

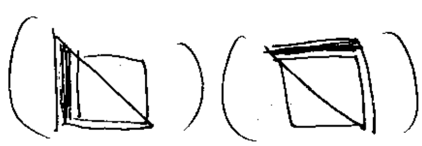


$$L_{11}U_{11} = A_{11}$$

אז אפשר לבחור L_{11}, U_{11} כשהם בלת מתקיים לפורם.

$$L_{21}U_{11} = A_{21}$$

⇐ אפשר למצוא את L_{21}. רק אפשר למצוא את כל העמודה הראשונה ובאופן דומה את כל השורה הראשונה של U, וממשיכים באינדוקציה:



for $k=1,\dots,n$

$$L_{kk}\cdot U_{kk} = A_{kk} - \sum_{i<k} L_{ki}U_{ik}$$

  for $j=k+1,\dots,n$

$$L_{jk} = \frac{A_{jk} - \sum_{i<k} L_{ji}U_{ik}}{U_{kk}}$$

  end

  for $j=k+1,\dots,n$

$$U_{kj} = \frac{A_{kj} - \sum_{i<k} L_{ki}U_{ij}}{L_{kk}}$$

  end

end

השיטה הזאת לוקחת $O(n^3)$. (כלומר לא שיפרנו כלום לעומת גאוס. אבל אם עושים ספירה מדוייקת אז יש הבדל בקבועים - לפירוק יש $\frac{2}{3}$ מהפעולות של גאוס.

בחילוץ גאוס שימוש בעיית pivoting שגורמת לאי-יציבות נומרית. פירוק LU הרבה יותר יציב.

(23) 15/11/09
חישוב
נומרי

פתרון משוואות לינאריות $Ax=b$

שיטות ישירות:
- חילוף גאוס
- פירוק LU
- פירוק Cholesky - היום!

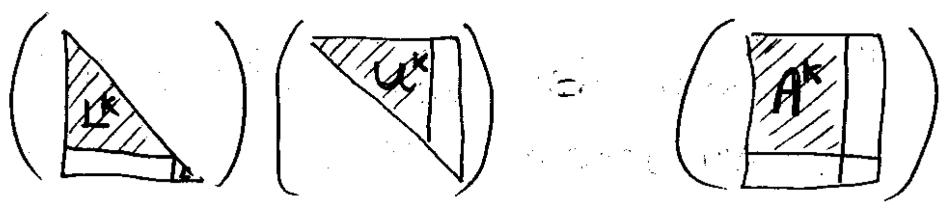
טענה: קיים פירוק LU למטריצות A שכל תת מטריצה עיקרית שלה היא הפיכה (תת מטריצה היא עיקרית אם האלכסון שלה הוא חלק מהאלכסון של A).

הערה: זה לא אמת

הוכחה: באינדוקציה על k רק ש- $A^k_{ij}=A_{ij}$ לכל $k\ge i,j\ge 1$.

עבור $k=1$ $A^k\in\mathbb{R}$ ואז ברור שניתן להציג אותה כמכפלה של שני סקלרים.

נניח נכונות על A^k ונראה עבור A^{k+1}.



מהנחת האינדוקציה יש לנו $L^k\cdot U^k=A^k$

ואנחנו רוצים למצוא $L^{k+1}U^{k+1}=A^{k+1}$

אז אנחנו צריכים להראות שאפשר להוסיף שורה ל- L^k ועמודה ל- U^k כך שהמשוואה תעבוד.

$$\begin{bmatrix} L^k & \\ l^{k+1} & 1\end{bmatrix}\begin{bmatrix} U^k & u^{k+1}\end{bmatrix}=\begin{bmatrix} L^kU^k & L^ku^{k+1} \\ l^{k+1}U^k & l^{k+1}u^{k+1}\end{bmatrix}$$

כלומר אנחנו רוצים $l^{k+1}U^k=A_{k+1,*}$

A^k הפיכה אבל $A^k=L^kU^k$ ולכן גם U^k הפיכה. לכן

מערכת המשוואות הנ"ל פתירה. אז קיים פתרון l^{k+1} שמקיים את הדרוש. באופן זהה נדרוש למצוא u^{k+1} כך ש- $L^k \cdot u^k = A_{*,k+1}$ וזה אפשרי כי L^k הפיכה.

הערה: מההוכחה ניכר שאפשר היה לצמצם את דרישת ההפיכות רק לתת מטריצות עיקריות שמתחילות בפינה.

שיטת LU עבור מטריצות סימטריות נקראת שיטת Cholesky והיא: $A = LL^T$

```
for k=1,...,N
    L_kk^2 = A_kk - sum_{i=1}^{k-1} L_ki^2
    for i=k+1,...,N
        L_ki = (A_ki - sum_{j=1}^{k-1} L_kj L_ji) / L_kk
    end
end
```

$$L_{kk}^2 = A_{kk} - \sum_{i=1}^{k-1} L_{ki}^2$$

$$L_{ki} = \frac{A_{ki} - \sum_{j=1}^{k-1} L_{kj} L_{ji}}{L_{kk}}$$

לאילו מטריצות אנחנו יכולים לפרק $A = LL^T$? אם הצלחנו אז A בוודאי מוגדרת חיובית.
מאידך, אם A מוגדרת חיובית אז גם A^k מוגדרת חיובית לכל k. לכן A^k הפיכה לכן יש ל-A פירוק LU.
אבל אנחנו רוצים שהפירוק שלה יהיה $A = LL^T$.
~~אם A סימטרית ומוגדרת חיובית אז אפשר לפרק $A = Q^T D Q$~~ אלכסונית חיובית

אז יש לנו
$$A = LU$$
$$A^t = U^t L^t \Rightarrow U(L^t)^{-1} = L^{-1} U^t$$

אבל אם L משולשית תחתונה אז גם L^{-1} משולשית תחתונה, והרי U^t משולשית תחתונה. אז אגף ימין משולשית תחתונה. ובאותו אופן אגף שמאל משולשית עליונה. לכן

$U(L^t)^{-1} = L^{-1}U^t = \tilde{D}$ אלכסונית. עכשיו

$$L\tilde{D}L^t = LU = A$$

ועכשיו $A = (L\sqrt{\tilde{D}})(L\sqrt{\tilde{D}})^t$.

$\tilde{D}$ חיובית באלכסון בהכרח כי A מוגדרת חיובית ! - $A = L\tilde{D}L^t$.

לכן אפשר להוציא לה שורש.

<u>סיכום:</u>

① מטריצה עם תת מטריצות עיקריות הפיכות היא מתפרקת ל- LU

② מטריצה סימטרית ומוגדרת חיובית מתפרקת ל- LL^t (צ'ולסקי):

מוגדרת חיובית ⇐ עיקריות מוגדרות חיובית ⇐ עיקריות הפיכות ⇐ יש פירוק LU.

סימטריה ⇓

$$0 \le D \xLeftarrow{A \text{ מוגדרת חיובית}} A = LDL^t \Leftarrow D = UL^{-t} = L^{-1}U \text{ אלכסונית}$$

⇓

$$A = L\sqrt{D}\sqrt{D}^t L^t$$

<u>שיטות איטרטיביות לפתרון $Ax=b$</u>

שיטה איטרטיבית היא שיטה שבה בכל איטרציה משפרים את הפתרון שיש לנו ויש לנו מיני מדדים כדי לקבוע מתי לעצור.

$$Ax = b$$

מטריצת פירוק splitting matrix

$$Q \Rightarrow Qx - Qx + Ax = b$$

$$\Rightarrow Qx = (Q-A)x + b$$

$$x = Q^{-1}(Q-A)x + Q^{-1}b$$

לא עשינו שום מע' משוואות בינתיים רק שהכנסנו איזו מטריצה הפיכה Q (ורצוי כמובן שנדע את ההופכית שלה)

איך זה עוזר לפתור? פשוט נהפוך את זה לאיטרציות:

$$x^{n+1} = Q^{-1}((Q-A)x^n + b)$$

$$x^0 = \text{ניחוש התחלתי}$$

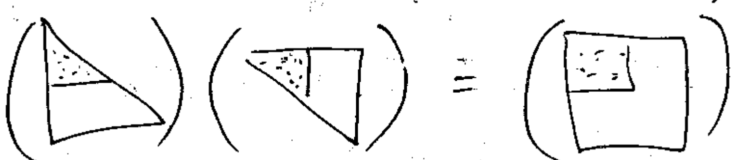
<u>דרישות מ- Q</u> :

1. קלה להיפוך
2. שהתהליך יתכנס

<u>מטריצות Q מפורסמות</u> :

1. $Q = I$ שיטת ריצ'רדסון
2. $Q =$ אלכסונית עם האלכסון של A (Jacobi)
3. $Q =$ משולשית תחתונה עם המשולש התחתון של A (Gauss-Seidel)

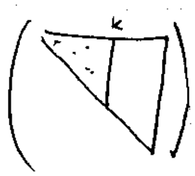
אולי אחת מ משפט הפירוק LU : ראינו שאם כל תתי המטריצות העיקריות הפיכות אז יש פירוק LU. אם המטריצה המקורית A הפיכה אז גם הכיוון ההפוך נכון: נניח שקיים פירוק $A = LU$. כך $L^k U^k$ זה פירוק ה-LU של התת מטריצה העיקרית A^k.



אם נראה ש- L^k, U^k הפיכות נקבל שגם A^k הפיכה.
נניח בשלילה ש- U^k לא הפיכה. כלומר קיים $x \neq 0$ כך ש- $U^k x = 0$.
נתבונן אז ב-

$$\hat{x} = \begin{pmatrix} x \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$\Leftarrow$

$U\hat{x} =$  $\begin{pmatrix} x_1 \\ \vdots \\ x_k \\ 0 \\ \vdots \\ 0 \end{pmatrix} = U^k x = 0$

אבל $\hat{x} \neq 0$ ולכן קיבלנו סתירה לכך ש- U הפיכה. (היא חייבת להיות הפיכה כי A הפיכה).

<u>דוגמה למטריצה שאין לה פירוק LU:</u> $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ יש לה תת מטריצה עיקרית (0) שאינה הפיכה ולכן אפשר להשתכנע שאי אפשר לפרק אותה.

שיטות איטרטיביות לפתרון $Ax = b$

$$Ax = b$$
$$\Rightarrow \quad Qx + Ax = b + Qx$$
$$\Rightarrow \quad Qx = (Q - A)x + b$$
$$\Rightarrow \quad x = Q^{-1}(Q - A)x + Q^{-1}b$$

והאיטרציות הן $x^{n+1} = Q^{-1}(Q-A)x^n + Q^{-1}b$

מה צריך כדי שזה יעבוד?

- צריך ש-Q תהיה הפיכה (ורצוי שזה יהיה אפשרי לחישוב בקלות)
- צריך שהתהליך הזה יתכנס למה שצריך

אם Q דומה ל-A אז ״נקבל״

$$x^{n-1} = (I-I)x^n + A^{-1}b = A^{-1}b$$

אז כנראה נרצה ש-Q תהיה דומה ל-A בכל אופן.

איך בוחרים את Q?

- Richardson - $Q = I$
- Jacobi - $Q = diag(A)$
- Gauss Seidel - $Q = lower\text{-}tr(A)$

כל איטרציה כרוכה בכפל מטריצות. במקרה הכללי זה לוקח $O(N^2)$ אבל אם המטריצה דלילה זה לוקח $O(N)$ (ההגדרה של מטריצה דלילה היא שבכל שורה יש $O(1)$ ערכים שאינם אפס)

אז אם מספר האיטרציות קטן מ-N אנחנו במצב טוב.

<u>הערה</u>: למה מטריצה משולשית תחתונה היא קלה להפיכה? ובכן, אם אנחנו נתבונן בהגדרה $Qx^{n+1} = (Q-A)x^n + b$ ואגף ימין ידוע. אז מה שיש לנו זה מע' משוואות שהמטריצה שלה משולשית תחתונה אז קל מאוד לפתור את זה.

<u>נורמה אופרטורית של מטריצה</u>

$$\|A_k\| = \max_{x \neq 0} \|Ax\|_k / \|x\|_k$$

כאשר

$$\|x\|_k = \sqrt[k]{\sum_{i=1}^{N} x_i^k}$$

(26)

ברור ש- $\|A\| = 0$ אמ"מ $A=0$

<u>למה:</u> $\|A\|_2 = \rho(A)$ (הרדיוס הספקטרלי של A)

<u>הוכחה:</u> נראה למקרה ש- A היא מטריצה חיובית וסימטרית אבל זה נכון תמיד

$$\|A\|_k = \max_{x\neq 0} \|Ax\|_k / \|x\|_k = \max_{\|x\|=1} \|Ax\|_k$$

וברור ש- $\arg\max_{\|x\|=1} \|Ax\| = \arg\max_{\|x\|=1} \|Ax\|_2^2$

$$\max_{x^t x=1} x^t A^t A x = \max_{x^t x=1} x^t A^2 x$$

נשתמש בשיטת כופלי לגרנז' ונקבל $2A^2x = \lambda 2Ix$

$\Rightarrow A^2 x = \lambda x$. אבל A מטריצה חיובית לכן הע"ע של A^2 הם ריבוע הע"ע של A. ואותם ו"ע. לכן המקסימום בהם הו"ע של A-

$Av_i = \lambda v_i$, אז המקסימום מתקבל בע"ע המקסימלי

$$\frac{\|Av_i\|_k}{\|v_i\|} = \frac{\|\lambda_i v_i\|}{\|v_i\|} = |\lambda_i|$$

למה פה אפשר היה להוכיח את זה רק בעזרת ההנחה ש- A סימטרית.

$\|A\| = \max \frac{\|Ax\|}{\|x\|} \Rightarrow$ לכל x, $\|A\|\|x\| \ge \|Ax\|$

$$\Rightarrow \|A^2 x\| = \|A(Ax)\| \le \|A\|\|Ax\| \le$$
$$\le \|A\|\|A\|\|x\| = \|A\|^2\|x\|$$

וכמובן אפשר להכליל ולקבל $\|A^n x\| \le \|A\|^n \|x\|$

אז ניזכר שהיה לנו $x^{n-1} = Q^{-1}(Q-A)x^n + Q^{-1}b$

נגדיר את השגיאה $e^n = x^n - x$

(כאשר x פתרון של המשוואה). מה האבולוציה של השגיאה?

$$e^{n+1} = x^{n+1} - x = Q^{-1}(Q-A)x^n + Q^{-1}b - x$$
$$= x^n - x - Q^{-1}Ax^n + Q^{-1}b =$$
$$= x^n - x - Q^{-1}(Ax^n - b) =$$
$$= x^n - x - Q^{-1}(A(e^n + x) - b) = (I - Q^{-1}A)e^n$$

כלומר $e^{n+1} = (I - Q^{-1}A)e^n$

ופה שוב אנחנו רואים למה נרצה ש- Q תהיה דומה ל- A כי אז $(I - Q^{-1}A)e^n \approx (I - I)e^n = 0$

$$\Rightarrow \|e^{n+1}\| = \|(I - Q^{-1}A)e^n\| \le$$
$$\le \|I - Q^{-1}A\| \, \|e^n\| \left(\le \dots \le \|I - Q^{-1}A\|^{n+1} \|e^0\|\right)$$

לכן נרצה שהנורמה של $I - Q^{-1}A$ תהיה קטנה מ- 1 ואז מובטחת לנו התכנסות. למעשה מספיק שאיזושהי נורמה k תהיה קטנה מ- 1 כי כל הנורמות שקולות אז אם בנורמה מסוימת השגיאה מתכנסת לאפס אז בכל הנורמות השגיאה תתכנס לאפס.

אנליזת התכנסות של Jacobi עבור מטריצות diagonally dominant

כאן $Q = diag(A)$

נרצה למצוא את התנאי על A כך ש- $\|I - Q^{-1}A\|_\infty < 1$

<u>תזכורת</u>
$$\|A\|_\infty = \max_{\substack{\|x\|_\infty = 1 \\ \max|x_i|}} \|Ax\|_\infty = \max_i \sum_j |a_{ij}|$$

$$I - Q^{-1}A = \begin{pmatrix} 0 & \ddots & * \\ * & & 0 \end{pmatrix}$$
$$\downarrow$$
$$(Q^{-1}A)_{ij} = \frac{A_{ij}}{A_{ii}}$$

$$\|I - Q^{-1}A\| = \max_i \sum_{j \ne i} \left| \frac{A_{ij}}{A_{ii}} \right|$$

נאמר שמטריצה היא diagonally dominant

אם $|A_{ii}| \ge \sum_{j \ne i} |A_{ij}|$ (סכום האלכסון גדול מסכום השורות)

ואז $\|I - Q^{-1}A\| < 1$ והאיטרציות מתכנסות.

האלגוריתם:

Jacobi (A, b, x^0)

```
for n = 1, ..., num_iter
    for i = 1, ..., N
        x_i^{n+1} = -Σ_{j≠i} (A_ij / A_ii) · x_j^n + b_i / A_ii
    end
end
```

$$x_i^{n+1} = -\sum_{j \neq i} \frac{A_{ij}}{A_{ii}} \cdot x_j^n + \frac{b_i}{A_{ii}}$$

צריך להחליט על מספר האיטרציות או להחליט על איזשהו מדד שמראה וזה לא בהכרח קל כל כך. למשל אנחנו יודעים את

$$\|A(x-x^n)\| = \|b-Ax^n\|$$

אבל אם מדובר בוקטורים מאוד קטנים אז זה פשוט יהיה קטן. בקיצור, זו שאלה עדינה. אבל בכל זאת נהוג להשתמש בו, כלומר לחכות עד ש- $\|A(x-x^n)\|<\varepsilon$

28 26/11/09
חישוב נומרי

הודעה אדמיניסטרטיבית: לקורס יש תיבה חדשה בבוס 1-.
מעתה מגישים תרגילים לתיבה 14 ולא לתיבה 12.

שיטות איטרטיביות עבור $Ax=b$

$$x^{n+1} = Q^{-1}[(Q-A)x^n + b]$$

בשיטת Jacobi, $Q = diag(A)$.

התכנסות מובטחת עבור מטריצות diagonally dominant - כלומר $|A_{ii}| > \sum_{j\neq i} |A_{ij}|$ לכל i:

$$e^{n+1} = (I - Q^{-1}A)e^n$$

$$\|e^{n+1}\| \le \|I - Q^{-1}A\|^n \|e^0\|$$

השיטות האלה הן
$O(N)$ עבור מטריצות דלילות
$O(N^2)$ עבור מטריצות מלאות
וזה תחת ההנחה שמספר האיטרציות בלתי תלוי ב-N.
אנחנו נראה דוגמאות שבהן זה דווקא כן תלוי ב-N
ואז נראה מה לעשות בעניין.

תזכורת: $\|A\| = \max_{x\neq 0} \frac{\|Ax\|}{\|x\|}$

עבור A סימטרית, $\|A\|_2 = \rho(A)$
עבור מטריצות A נורמליות (כלומר $A^TA = AA^T$)
מתקיים $\|A\|_2 = \sqrt{\lambda_{max}(A^TA)}$
זה תיקון לשיעור הקודם. הזהות $\|A\|_2 = \rho(A)$ לא נכונה לכל המטריצות אלא רק לסימטריות.

בשיטת גאוס-זיידל בוחרים $Q = Lower(A)$. נכון שגם במקרה זה דומיננטיות אלכסונית מבטיחה התכנסות.

נניח ש- x וע"ע של המטריצה $I-Q^{-1}A$ רק ש- $\|x\|_\infty = 1$.

אז $Ix - Q^{-1}Ax = (I-Q^{-1}A)x = \lambda x$

$\Leftarrow$ $(Q-A)x = Qx - Ax = Qx$

אבל $Q = lower(A)$ אז נשאר לנו רק מה שמעל האלכסון. לכל i אז מתקיים

$$-\sum_{j=i+1}^{N} A_{ij}x_j = \lambda \sum_{j=1}^{i} A_{ij}x_j$$

$$\Rightarrow A_{ii}x_i = -\sum_{j=i+1}^{N} A_{ij}x_j - \lambda\sum_{j=1}^{i-1} A_{ij}x_j \quad \forall i$$

נבחר ב- i רק ש- $x_i \ge x_j$ לכל j. אז בהכרח $|x_i| = 1$ שהרי $\|x\|_\infty = 1$. בה"כ $x_i = 1$ (אחרת נכפול ב- $-x$.) אז

$$\lambda A_{ii} = -\sum_{j=i+1}^{N} A_{ij}x_j - \lambda\sum_{j=1}^{i-1} A_{ij}x_j$$

$$\Rightarrow |\lambda||A_{ii}| \le \sum_{j=i+1}^{N} |A_{ij}||x_j| + |\lambda|\sum_{j=1}^{i-1} |A_{ij}||x_j| \le$$

$$\underset{|x_j|\le 1}{\le} \underbrace{\sum_{j=i+1}^{N} |A_{ij}|}_{S_1} + |\lambda| \underbrace{\sum_{j=1}^{i-1} |A_{ij}|}_{S_2} =$$

$$= S_1 + |\lambda| S_2$$

$$\Rightarrow \lambda \le \frac{S_2}{|A_{ii}| - S_1}$$

אבל מתכונת הדומיננטיות $|A_{ii}| > S_1 + S_2$

$\Leftarrow$ $|A_{ii}| - S_1 > S_2$ ולכן

$$\lambda \le \frac{S_2}{|A_{ii}| - S_1} < 1$$

$\Leftarrow$ השיטה מתכנסת שהרי הע"ע של $I-Q^{-1}A$ קטנים ממש מ- 1, ואם נסתכל במטריקת l_2

$$\|e^{n+1}\|_2 = \|(I-Q^{-1}A)e^n\|_2 \le \lambda_{max}(I-Q^{-1}A)\|e^n\|_2$$

ודרשנו ממשוררים ובגלל ש- $\lambda_{max} < 1$ זה מתכנס.

למה $\|Ax\|_2 \le \lambda_{max}(A)\|x\|_2$ לכל A לכסינה? (29)

נניח ש- $Av_i = \lambda_i$ מא"נ. אז

$$\|Ax\|_2^2 = \sum_{i=1}^{N} a_i^2 \lambda_i^2 \le \lambda_{max}^2 \sum_{i=1}^{N} a_i^2 =$$

$$= \lambda_{max}^2 \|x\|_2^2$$

נרשום את הפסאודו-קוד:

Gauss-Seidel

until convergence

for $i = N \ldots 1$

$$x_i^{n+1} = \frac{b_i - \sum_{j=i+1}^{N} A_{ij} x_j^{n} - \sum_{j=1}^{i-1} A_{ij} x_j^{n+1}}{A_{ii}}$$

end

היתרון העיקרי של השיטה הזו הוא שהוא ניתן למימוש in-place בלי הקצאת זיכרון נוסף.

בשיטת Jacobi צריך תמיד להחזיק את x^n ואת x^{n+1}. חוץ מזה בהרבה בעיות שיטת גאוס זיידל מתכנסת טיפ-טיפה יותר מהר מ-Jacobi אבל זה לא משמעותי.

מאידך יש מקרים שבהם Jacobi מתכנס אבל גאוס לא.

הערה: שני הפתרונות מתכנסים לכל וקטור b מכל ניחוש ראשוני x^0!

שיטת Steepest Descent

השיטה מוגדרת עבור מטריצות סימטריות ומוגדרות חיוביות.

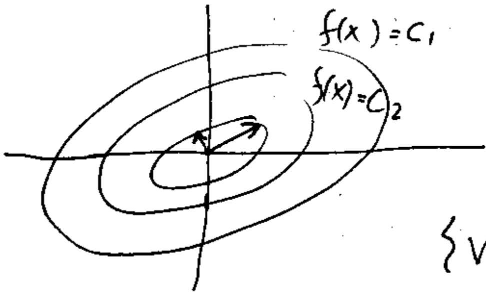
ראינו מ"פ $\langle x, y\rangle = x^t y$

אפשר גם להגדיר פונקציה $\langle x, y\rangle_A = \langle x, Ay\rangle = x^t A y$

ואפשר להראות שאם A סימטרית ומוגדרת חיובית אז

גם זו מכפלה פנימית ולכן אפשר להגדיר נורמה $\|x\|_A^2 = x^t A x$.

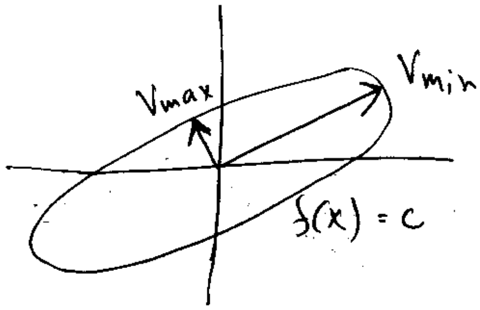
עבור A מוגדרת חיובית הפונקציה $f(x)=x^tAx$ מגדירה אליפסואיד.



אם A מוגדרת חיובית אז אפשר לפרק אותה $A=QDQ^t$

כאשר $Q=\begin{pmatrix} | & & | \\ v_1 & \dots & v_n \\ | & & | \end{pmatrix}$ רוש $\{v_i\}$

מא"נ (ולכן $Q^t=Q^{-1}$) ו- $D=\begin{pmatrix}\lambda_1 & & \\ & \ddots & \\ & & \lambda_n\end{pmatrix}$



נגדיר $f(x)=\frac{x^tAx}{2}-x^tb$. זאת תבנית ריבועית - פרבולה ויש לה מינימום שמתקבל בפתרון של $Ax=b$.

$$\tilde{f}(x)=\|Ax-b\|^2_{A^{-1}}=$$

[נזכור שאם A מוגדרת חיובית אז גם A^{-1}].

$$=(Ax-b)^tA^{-1}(Ax-b)=$$
$$=(x^tA^t-b^t)A^{-1}(Ax-b)=$$
$$=(x^tA^t-b^t)(x-A^{-1}b)=(x^tA-b^t)(x-A^{-1}b)=$$
$$=x^tAx-x^tb-b^tx+b^tA^{-1}b=$$
$$=x^tAx-2x^tb+b^tA^{-1}b$$

אבל $b^tA^{-1}b$ קבוע ולכן למזער את $\tilde{f}$ זה בדיוק כמו למזער את f. והמזער הוא הפתרון של מע' המשוואות.

אז המטרה שלנו היא למזער את $f(x)$ ואת זה נעשה בשיטה איטרטיבית.

<u>סימונים:</u>

$e^n=x^n-r$ השגיאה

$r^n=b-Ax^n$ השארית

$\Rightarrow Ae^n=-r^n$

אנחנו הולכים לחפש את x באופן הבא:

$$x^{n+1} = x^n + \alpha d^n$$

וככה שלב מחפשים את ה- α שממזער את ערך הפונקציה f.

ו- d^n הוא למעשה הכיוון שאנחנו הולכים בו. נבחר את

$d^n = Ae^n = -r^n$.

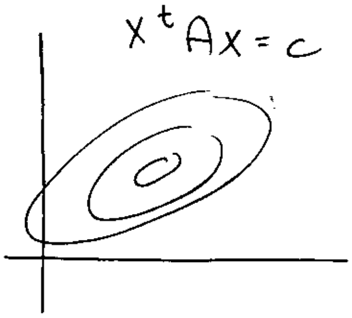
(31) 29/11/09
חישוב נומרי

<u>הערה מהנייות של תרגול 2 שאלה ב</u>

היינו צריכים לחשב את $\int_0^\pi \sin x\,dx$. אנחנו עשינו אנליזת שגיאה
במקרה שמרווחי הדגימה הם אחידים. אז דרשנו אם סתם נתחן
מרווח שגיאה כך שגם h ירצו להיות שווה לא נכנס כי π מספר
שלם של פעמים ואז אנליזת השגיאה שלנו פשוט לא נכונה.
יתרה מזו, השיטות לא נכונות בכלל! כי המשקלות $\frac{1}{3}$, $\frac{4}{3}$, $\frac{1}{3}$
נכונות <u>רק</u> עבור מרווחים קבועים! אז לא רק שאנליזת השגיאה
לא מדוייקת אלא שאפילו לא עשינו אינטרפולציה כמו שצריך!
אז אפשר לתקן את המשקלות עבור 3 הנקודות האחרונות

אז לפחות השיטה תהיה נכונה אבל אנליזת ההתכנסות עדיין
מסתמכת על כך שהמרווחים שווים ולכן נראה לא תהיה התכנסות
של $O(h^2)$.

Steepest Descent



מינימיזציה של $f(x)=\frac{x^tAx}{2}-x^tb$
נותנת לנו את הפתרון של מע'
המשוואות $Ax=b$.

סימונים:
$$e^n = x^n - x$$
$$r^n = b - Ax^n = -Ae^n$$
$$f'(x^n) = Ax^n - b = -r^n$$

אך r^n יש לנו כל שלב. בשיטת ה- Steepest Descent
בוחרים $d^n = Ae^n$ ואז מנסים למזער את $f(x)$ לאורך ציר d^n.
אז ננסה למזער - נגזור ונשווה לאפס

$$\frac{\partial}{\partial \alpha} f(x^n+\alpha r^n) =$$

$$= \frac{\partial}{\partial \alpha}\left[\frac{(x^n+\alpha r^n)^t A (x^n+\alpha r^n)}{2} + (x^n+\alpha r^n)^t b\right.$$

$$= (r^n)^t A (x^n+\alpha r^n) - (r^n)^t b =$$

$$= (r^n)^t\left(A(x^n+\alpha r^n)-b\right) = (r^n)^t\left(\alpha A r^n - r^n\right) = 0$$

$$r^n = \frac{r^{nt} r^n}{r^{nt} A r^n} \Leftarrow$$

Steepest Descent

x^0 - initial guess

until convergence

$$r^n = b - Ax^n$$

$$\alpha = \frac{r^{nt} r^n}{r^{nt} A r^n}$$

$$x^{n+1} = x^n + \alpha r^n$$

end

fine print:

עבור מטריצות דלילות השיטה היא $O(N)$

אבל $O(N^2)$ עבור מטריצות כלליות (מלאות)

וזה בהנחה שמספר האיטרציות בלתי תלוי ב- N.

בשיטה הזאת מתקיימות שתי תכונות:

1) $r^n \perp r^{n+1}$

2) $r^n \perp_A e^{n+1}$

מה קצב ההתכנסות של השיטה? נסתכל קודם על כמה מקרים מיוחדים.

I אם קיים $e^n = v$ כאשר $Av = \lambda v$ אז.

$$x^{n+1} = x^n + \alpha d^n = x^n + \alpha r^n$$

$$\Rightarrow e^{n+1} = x^{n+1} - x = x^n + \alpha r^n - x = e^n + \alpha r^n =$$

$$= e^n - \alpha \lambda v = v - \alpha \lambda v =$$

$r^n = -Ae^n$

$$= V - \frac{r^{nt} r^n}{r^{nt} A r^n} \lambda v = v - \frac{\cancel{\lambda^2} \|v\|^2}{\cancel{\lambda^3} \|v\|^2} \cancel{\lambda} v = v - v = 0$$

מציבים את α

אז המקרה הראשון הוא ממש מצוין לנו!
מה האינטואיציה? בעצם במקרה ויצאנו שציר החיפוש כוון
את הפתרון האמיתי הנכון ולכן תוך צעד אחד זה נפתר.

II $\lambda_i = \lambda$ כל הע"ע של A הם זהים ואז A מגדירה כדור.
אז $A = \lambda I$ סקלרית.
הרי הכיוון שאנחנו בוחרים הוא הגרדיאנט $f'(x^n) = -r^n$
ובמקרה של כדור הגרדיאנט מצביע למרכז הכדור. אז שוב
ציר החיפוש שלנו מכוון את התשובה והחיים יפים.

$$e^{n+1} = e^n + \alpha r^n = e^n - \alpha \lambda e^n = e^n - \frac{r^{nt} r^n}{\underbrace{r^n A r^n}_{= \lambda r^{nt} r^n}} \lambda e^n = 0$$

III המקרה הכללי (כמעט). נניח ש- $\{v_i\}$ בסיס של ו"ע.
כאשר $A v_i = \lambda_i v_i$. אז אפשר לכתוב $e^n = \sum_{i=1}^{N} a_i v_i$

$$\|e^{n+1}\|_A^2 = (e^{n+1})^t A e^{n+1} = (e^n + \alpha r^n)^t A (e^n + \alpha r^n) =$$

$$= \|e^n\|_A^2 + 2\alpha (r^n)^t A e^n + \alpha^2 (r^n)^t A r^n =$$

$$= \|e^n\|_A^2 + 2 \frac{r^{nt} r^n}{r^{nt} A r^n} \cdot r^{nt} \underbrace{A e^n}_{-r^n} + \left(\frac{r^{nt} r^n}{r^{nt} A r^n} \right)^2 r^{nt} A r^n =$$

$$= \|e^n\|_A^2 - 2 \frac{(r^{nt} r^n)^2}{r^{nt} A r^n} + \frac{(r^{nt} r^n)^2}{r^{nt} A r^n} =$$

$$= \|e^n\|_A^2 - \frac{(r^{nt} r^n)^2}{r^{nt} A r^n} = \|e^n\|_A^2 \left(1 - \frac{(r^{nt} r^n)^2}{e^{nt} A e^n \, r^{nt} A r^n} \right)$$

$$= \|e^n\|_A^2 \cdot \omega^2$$

מסמנים ב- ω

נזכור ש-

$$\Leftarrow \quad e^n = \sum \alpha_i v_i$$

$$r^n = Ae^n = \sum \alpha_i \lambda_i v_i$$

$$\Rightarrow \quad \omega = 1 - \frac{\left[\left(\sum \alpha_i \lambda_i v_i\right)^t \left(\sum \alpha_i \lambda_i v_i\right)\right]^2}{\left(\sum \alpha_i v_i\right)^t \left(\sum \alpha_i \lambda_i v_i\right)\left(\sum \alpha_i \lambda_i v_i\right)^t \left(\sum \alpha_i \lambda_i^2 v_i\right)} =$$

$$= 1 - \frac{\left(\sum \alpha_i^2 \lambda_i^2\right)^2}{\left(\sum \alpha_i^2 \lambda_i\right)\left(\sum \alpha_i^2 \lambda_i^3\right)}$$

$\{v_i\}$ בא״נ

יש אנשים שהתעסקו עם הביטוי הזה אבל אנחנו בשלב זה נעבור ל- $N=2$.

אם יש לנו v_1, v_2

ע״ע $\lambda_1 \geq \lambda_2$

נגדיר $K = \frac{\lambda_1}{\lambda_2} \geq 1$ ו- $\mu = \frac{\alpha_2}{\alpha_1}$

$$\omega^2 = 1 - \frac{\left(\alpha_1^2 \lambda_1^2 + \alpha_2^2 \lambda_2^2\right)^2}{\left(\alpha_1^2 \lambda_1 + \alpha_2^2 \lambda_2\right)\left(\alpha_1^2 \lambda_1^3 + \alpha_2^2 \lambda_2^3\right)} \cdot \frac{\frac{1}{\alpha_1^4}}{\frac{1}{\alpha^4}} =$$

$$= 1 - \frac{\left(\lambda_1^2 + \mu^2 \lambda_2^2\right)^2}{\left(\lambda_1 + \mu^2 \lambda_2\right)\left(\lambda_1^3 + \mu^2 \lambda_2^3\right)} \cdot \frac{\frac{1}{\alpha_2^4}}{\frac{1}{\alpha_2^4}} =$$

$$= 1 - \frac{\left(K^2 + \mu^2\right)^2}{\left(K + \mu^2\right)\left(K^3 + \mu^2\right)}$$

K תלוי בתכונות של המטריצה והוא קבוע. אבל μ תלוי בניחוש ההתחלתי שלנו. מתברר שאם $\mu = \pm K$ אז הביטוי הזה מתמקסם ... את השגיאה הגדולה ביותר.

$$1 - \frac{4K^4}{K^5 + 2K^4 + K^3} = \frac{K^5 - 2K^4 + K^3}{K^5 + 2K^4 + K^3} =$$

$$= \frac{K^2 - 2K + 1}{K^2 + 2K + 1} = \frac{(1-K)^2}{(1+K)^2}$$

$\Leftarrow \quad \omega \leq \frac{1-K}{1+K} \lneq 1$. אז השיטה תמיד מתכנסת.

במקרה של N כללי: עושים משהו דומה אבל מגדירים

$$K = \frac{\lambda_{max}}{\lambda_{min}}$$

$$K(A) = \frac{\lambda_{max}}{\lambda_{min}}$$ נקרא ה-condition number של המטריצה. אנחנו רוצים שלה יהיה כמה שיותר גבוה.

(34) 3/12/09
חישובנומרי

אנחנו פותרים את $Ax=b$

אם A סימטרית מוגדרת חיובית, אז בשיטת ה-

Steepest Descent $\|e^{n+1}\|_A \le \|e^n_m\| W_{SD}$

כאשר $W_{SD} = \frac{K(A)-1}{K(A)+1}$; $K(A)$ מספר מצב

$$K(A) = \frac{\lambda_{max}}{\lambda_{min}} \ (\ge 1)$$

Conjugate Gradients

כמו קודם נסמן $e^n = x^n - x$ $\quad X^n = x^n + \alpha d^n$

$r^n = b - Ax^n = -Ae^n$

השיטה מתבססת על הרעיון שיש קשר בין קירוב אופטימלי לכיוונים.

אנחנו רוצים למצוא את $\min_\alpha \|e^n + \alpha d^n\|^2 = \min_\alpha \|e^{n+1}\|$

עבור e^n כלשהו d^n כלשהו (באותה מידה היינו ממזערים את $\|x - \alpha y\|^2$ ו- x,y כלשהם)

$$\frac{\partial}{\partial\alpha}\langle e^n+\alpha d^n, e^n+\alpha d^n\rangle = 2\langle d^n, e^n+\alpha d^n\rangle = 0$$

$$\Leftarrow \quad \alpha = \frac{e^{n^t} d^n}{d^{n^t} d^n}.$$

אם $e^{n+1} = e^n + \alpha d^n$ אז מה שאנחנו רוצים הוא לגרום לזה שהשגיאה שנקבל תהיה מאונכת לכיוון החיפוש הנוכחי.

הצעה: בהינתן סדרת כיוונים $d^1,\dots,d^n$ רק ש- $d^i \perp d^j$ עבור $i \ne j$ אם נחשב את x^n ע"י

```
for i=0,...,n
    x^{i+1} = x^i + α d^i ,    α = e^{i^t} d^i / (d^{i^t} d^i)
end
```

נראה ש- $e^n \perp d^0,\dots,d^{n-1}$. מהאופן שבו בחרנו את α נובע

ש- $e^{n+1} \perp d^n$. ואנחנו יודעים ש- $d^i \perp d^j$
ל- $i \neq j$. נראה שמההנחה ש- $e^n \perp d^0,\dots,d^{n-1}$
נובע ש- $e^{n+1} \perp d^0,\dots,d^n$ ונקבל את מה שצריך באינדוקציה

$$\langle d^k, e^{n+1} \rangle = \langle e^n + \alpha d^n, d^k \rangle =$$
$$= \langle e^n, d^k \rangle + \alpha \langle d^n, d^k \rangle = 0 + \alpha \cdot 0 = 0$$

אז נראה אחלה שיטה. אלא שאנחנו לא יודעים את e^n
ולכן לא יודעים את α ...

בשיטת SD ניסינו למזער את $f(x^n + \alpha d^n)$ כאשר
$f(z) = z^t A z - 2 z^t b + c$
וקיבלנו ש-

$$\|e^{n+1}\|_A^2 = \|e^n + \alpha d^n\|_A^2 =$$
$$= (x^n + \alpha d^n - x)^t A (x^n + \alpha d^n - x) =$$
$$= (x^n + \alpha d^n)^t A (x^n + \alpha d^n) - 2(x^n + \alpha d^n)^t A x + x^t A x =$$
$$= (x^n + \alpha d^n)^t A (x^n + \alpha d^n) - 2(x^n + \alpha d^n)^t b + \underbrace{x^t b}_{c \text{ קבוע}}$$

אז למזער את השגיאה זה בדיוק כמו למזער את
$f(x^n + \alpha d^n)$. נגזור לפי α ונשווה לאפס ונקבל

$$\cancel{2} d^{n\,t} A d^n \alpha + \underbrace{\cancel{2} x^{n\,t} A d^n}_{= 2 d^{n\,t} A x^n} - \cancel{2} d^{n\,t} b = 0$$

$$= d^{n\,t} A d^n \alpha + d^{n\,t}(A x^n - b)$$

$$\Rightarrow \quad \alpha = \frac{d^{n\,t} r^n}{d^{n\,t} A d^n}$$

ב- SD בחרנו $d^n = r^n$ ואז זה מתלכד.

אם מזערו לפי נורמה $\|\cdot\|_2$: $\langle e^{n+1}, d^n \rangle = 0$, $\alpha = \frac{e^{n\,t} d^n}{d^{n\,t} d^n}$

ומזערו לפי נורמה $\|\cdot\|_A$: $\langle e^{n+1}, d^n \rangle_A = 0$, $\alpha = -\frac{d^{n\,t} A e^n}{d^{n\,t} A d^n} = \boxed{\frac{d^{n\,t} r^n}{d^{n\,t} A d^n}}$

וראינו שאם $\{d^i\}$ אורתוגונליים לפי נורמת X אז $e^n \perp_X d^k$
לכל $0 \le k < n$ כאשר $e^{n+1} = e^n + \alpha d^n$ ו- $\langle e^{n+1}, d^n \rangle_X = 0$

שיטת SD מתקבלת עבור $d^n=r^n$ ושימוש בנורמה $\|\cdot\|_A$.
אבל $d^n=r^n$ רק ניצב ל-r^j....

שיטת CG מונה את $\{d^n\}$ שתהיה A-אורתוגונלית
וממזערת לפי נורמת $\|\cdot\|_A$, כאשר בוחרים

$$\alpha = \frac{d^{n^t} r^n}{d^{n^t} A d^n}$$

אז הבעיה נפתרת לאורך d כיוון באופן בלתי תלוי וגם אחרי
N צעדים מתקבלת $e^N=0$.

איך בוחרים $\{d^n\}$ שהם A-אורתוגונליים? צריך
להשתמש בשיטת גרם שמידט: נתון בסיס $\{r^0,\dots,r^{n-1}\}$
ממנו אפשר לייצר בסיס אורתוגונלי (ללא ההכרח לנרמל) ככה

$$d^0 = r^0$$
for $i=1,\dots,N-1$
$$d^i = r^i - \sum_{j=0}^{i-1} \frac{\langle r^i, d^j\rangle_A \cdot d^j}{\|d^j\|_A^2}$$
end

אבל זה קטסטרופלי! למטריצה מלאה A זה לוקח $O(N^3)$
אלא ש...
ראינו ש- $e^{n+1} \perp_A d^1,\dots,d^n$. גרם שמידט מקיים גם ש-

$$Sp\{r^0,\dots,r^i\} = Sp\{d^0,\dots,d^i\}$$

$\Leftarrow$ לכל $j<i$ $\quad d^i \perp_A r^j$

$x^{n+1}=x^n+\alpha d^n \Rightarrow e^{n+1}=e^n+\alpha d^n \Rightarrow r^{n+1}=r^n-\alpha A d^n$

$\Leftarrow \quad Sp\{d^0,\dots,d^i\} = Sp\{d^0, Ad^0,\dots,Ad^{i-1}\}$

$e^{n+1}\perp_A d^1,\dots,d^n$ פירושו לכל $1\le k\le n$ $\quad (e^{n+1})^t A d^k = 0$

$$= -r^{n+1} d^k \quad \Rightarrow \quad r^{n+1} \perp d^1,\dots,d^n$$

⇐ $r^n \perp Ad^k$ לכל $k<n-1$

⇐ $\langle r^n, Ad^k\rangle = 0$ ⇐ $\langle r^n, d^k\rangle_A = r^{n^t} A d^k = 0$

לכל $k<n-1$

⇐ בסכום בגרם שמידט למעשה יש רק איבר אחד.

כאשר $j=i-1$. ז"א יש לנו

for $i=1 \dots N-1$

$$d^i = r^i - \frac{\langle r^i, d^{i-1}\rangle_A}{\|d^{i-1}\|_A^2} d^{i-1}$$

end

וזה הרבה פחות חישובים.

הסברים והמשך שבוע הבא...

(36) 6/12/09
חישוב
נומרי

חזרה על מה שעשינו

אנחנו פותרים משוואה $Ax=b$ והיה לנו

$$e^n = x^n - x \quad , \quad r^n = b - Ax^n = -Ae^n$$

ראינו ש- $\min\|e^n+\alpha d^n\|_X$ מתקבל רק אם

$$\langle e^n+\alpha d^n, d^n\rangle_X = 0 \quad \text{ואז} \quad \alpha = \frac{\langle e^n, d^n\rangle_X}{\langle d^n, d^n\rangle_X}$$

וראינו שאם $d^0,\dots,d^n$ הם X-אורתוגונליים (מהקוד שלנו)

$X=A$ או $X=I$ ואז (כי המ"פ הסטנדרטיות) אז

$$e^{n+1} \perp_X d^0,\dots,d^n$$

אחרת (אם הם לא X-אורתוגונליים) אז $e^{n+1}\perp_X d^n$ וזהו.
(כמו ב- SD)

אם $X=I$ אז $\alpha = \frac{e^{n^t}d^n}{d^{n^t}d^n}$ ואת זה אנחנו לא יודעים לחשב כי אנחנו לא יודעים את e^n.

אם $X=A$ אז $\alpha = \frac{e^{n^t}Ad^n}{d^{n^t}Ad^n} = \frac{-r^{n^t}d^n}{d^{n^t}Ad^n}$ ואת זה אנחנו דווקא יודעים לחשב. ספציפית ב- SD נתתי $d^n = r^n$ ואז $\alpha = \frac{r^{n^t}r^n}{r^{n^t}Ar^n}$

ב- CG גם $X=A$ אבל בוחרים את d^n אחרת. בגלל שאנו רוצים d^n A-אורתוגונליים (ע"י פעולת גרם-שמידט על r^n):

$$d^0 = r^0 = b - Ax^0$$

until converged $(\|r^n\| < \varepsilon)$

$$\alpha = \frac{r^{n^t}d^n}{d^{n^t}Ad^n}$$

$$x^{n+1} = x^n + \alpha d^n$$

$$r^{n+1} = b - Ax^{n+1} = b - A(x^n + \alpha d^n) =$$

$$= b - Ax^n - \alpha A d^n = r^n - \alpha A d^n$$

$$d^{n+1} = r^{n+1} - \sum_{j=0}^{n} \underbrace{\langle d^j, r^{n+1}\rangle_A \cdot \frac{1}{\|d^j\|_A^2}}_{\beta_{n+1,j}} d^j$$

end

עכשיו נראה איך אפשר לייעל את זה גם כדי לעשות את החישוב מהיר יותר ובעיקר כדי שלא נצטרך לזכור את כל ההיסטוריה של ה-d-ים.

אנחנו נראה ש- $\beta_{n+1,j}=0$ לכל $j<n$ (אז בסכום יש למעשה רק מחובר אחד).

מתקיים $e^{n+1}\perp_A d^0,\dots,d^n$ כלומר $r^{n+1}\perp d^0,\dots,d^n$

שכן $\langle e^{n+1},d^j\rangle_A = e^{n+1^t}Ad^j = -r^{n+1^t}d^j =$
$= -\langle r^{n+1},d^j\rangle$

תוצאה, לכל i

$$Sp(r^0,\dots,r^i) = Sp(d^0,\dots,d^i)$$

זו תכונה של תהליך גרם שמידט.

ברור שמתקיים $Sp(v,w)=Sp(v,v+w)$ ולכן

$$Sp(r^0,\dots,r^i,r^{i+1}) = Sp(r^0,\dots,r^i,r^i-\alpha Ad^i) =$$
$$= Sp(r^0,\dots,r^i,-\alpha Ad^i) = Sp(r^0,\dots,r^i,Ad^i)$$

$\Leftarrow$ $Sp(r^0,\dots,r^i) = Sp(r^0,Ad^0,\dots,Ad^{i-1}) =$
$= Sp(d^0,\dots,d^i) = D^i$

אם $r^i\perp d^0,\dots,d^{i-1}$ אז $r^i\perp r^0,Ad^0,\dots,Ad^{i-1}$

$\Leftarrow$ $r^i\perp_A d^0,\dots,d^{i-1}$ $\Leftarrow$ $\beta_{i+1,j}=0\ \ \forall j<i$

וזה בדיוק מה שרצינו.

בספרות, האיבר $\alpha=\dfrac{r^{n^t}d^n}{d^{n^t}Ad^n}$ מופיע בדרך (שקולה) אחרת.

$$r^{n^t}d^n = r^{n^t}(r^n-\beta_{n,n-1}d^{n-1}) = r^{n^t}r^n$$
כי $r^{n^t}\perp d^{n-1}$

מבחינה חישובית זה לא תורם כלום. סתם שנבין מה שכתוב בספרות...

עוד משהו שמופיע מספרית ואפילו יכול לשפר מאוד. מבחינה

אופיע הסקלר $\frac{\langle d^n, r^{n+1}\rangle}{\|d^n\|_A^2}$ שהוא שווה ל -

$$\frac{r^{n+1^t} A d^n}{d^{n^t} A d^n} = \frac{r^{n+1^t}(r^n - r^{n+1})}{\alpha d^{n^t} A d^n} = -\frac{r^{n+1^t} r^{n+1}}{\alpha d^{n^t} A d^n} = -\frac{r^{n+1^t} r^{n+1}}{r^{n^t} r^n}$$

$$r^{n+1} = r^n - \alpha A d^n \Rightarrow A d^n = \frac{r^n - r^{n+1}}{\alpha}$$

$$r^i \perp r^j \quad \forall j < i$$

$$r^{i+1} \perp D^i \quad \text{כי}$$

אז מספרית מעדיפים $\beta = -\frac{r^{n+1^t} r^{n+1}}{r^{n^t} r^n}$ ומשתמשים בו לכתיבת האלגוריתם:

$d^0 = r^0 = b - Ax^0$

until convergence

$$\alpha = \frac{r^{n^t} r^n}{d^{n^t} A d^n}$$

$$x^{n+1} = x^n + \alpha d^n$$

$$r^{n+1} = r^n - \alpha A d^n$$

$$\beta = \frac{r^{n+1^t} r^{n+1}}{r^{n^t} r^n}$$

$$d^{n+1} = r^{n+1} + \beta d^n$$

end

ומה חסכנו? את הכפל במטריצה A בחישוב של β. אם הכפל מלאה זה מאוד משפר זמנים.

התכנסות CG

ב- SD ראינו ש- $\|e^{n+1}\|_A \le \left(\frac{K(A)-1}{K(A)+1}\right)^n \|e\|_A$

ב- CG אנחנו לא נוכיח כי זה סבוך אבל קצב ההתכנסות הוא

$$\|e^{n+1}\|_A \le 2\left(\frac{\sqrt{K(A)}-1}{\sqrt{K(A)}+1}\right)^n \|e^0\|_A \Rightarrow$$

סכום פתרונים לינאריים

	חילוץ גאוס	LU + BS	Cholesky	Jacobi + GaussSeidel	SD	CG
זמן חישוב	$O(N^3)$	$O(N^3) + O(N^2)$	$O(N^3) + O(N^2)$	$O(N)$ - דלילות, $O(N^2)$ - מלאות *	$O(N)$ - דלילות, $O(N^2)$ - מלאות *	$O(N)$ - דלילות, $O(N^2)$ - מלאות *
דרישות	אין	תתי מטריצות עיקריות הפיכות	מוגדרת חיובית וסימטרית	diagonally dominant	מוגדרת חיובית וסימטרית	מוגדרת חיובית וסימטרית
תנאי/קצב התכנסות	—	—	—	$\|I-Q^{-1}A\|_\infty$ **	$\frac{K(A)-1}{K(A)+1}$	$\sqrt[n]{2}\,\frac{\sqrt{K(A)}-1}{\sqrt{K(A)}+1}$

(*) בהנחה שמס' האיטרציות לא תלוי ב-N.

(**) מה זה הקצב אם ב-GS?

ראינו ש- $\lambda < \frac{S_2}{|A_{ii}|-S_1}$

$$S_1 = \sum_{j<i} |A_{ij}|$$

$$S_2 = \sum_{j>i} |A_{ij}|$$

$$\|I-Q^{-1}A\|_\infty = \max_i \frac{\sum |A_{ij}|}{|A_{ii}|} = \frac{S_1+S_2}{|A_{ii}|}$$

$$\Rightarrow |A_{ii}| = \frac{S_1+S_2}{\|I-Q^{-1}A\|_\infty}$$

$$\Rightarrow \frac{S_2}{|A_{ii}|-S_1} = \frac{S_2}{\frac{S_1+S_2}{\|\ \|_\infty}-S_1} = \frac{S_2\|\ \|_\infty}{S_1+S_2-S_1\|\ \|_\infty} = \frac{S_2\|\ \|_\infty}{(1-\|\ \|_\infty)S_1+S_2}$$

$$\le \frac{\|\ \|_\infty S_2}{S_2} = \|\ \|_\infty$$

$(1-\|\ \|_\infty)S_1 \ge 0$

(38) 10/12/09
חישוב נומרי

ראינו שיטות איטרטיביות לפתרון $Ax=b$ שבהן
בוחרים מטריצה Q כלשהי ומגדירים
$$x^{n+1} = Q^{-1}(b-(Q-A)x^n)$$
דיברנו על הבחירה $(Q=I)$, $Q=diag(A)$; $Q=lower(A)$
וראינו שקצב ההתכנסות הוא $\|e^{n+1}\|_\infty \le \|e^n\|_\infty \|I-Q^{-1}A\|_\infty$,
וראינו שצריך שהמטריצה תהיה diagonally dominant
כדי שזה יתכנס.

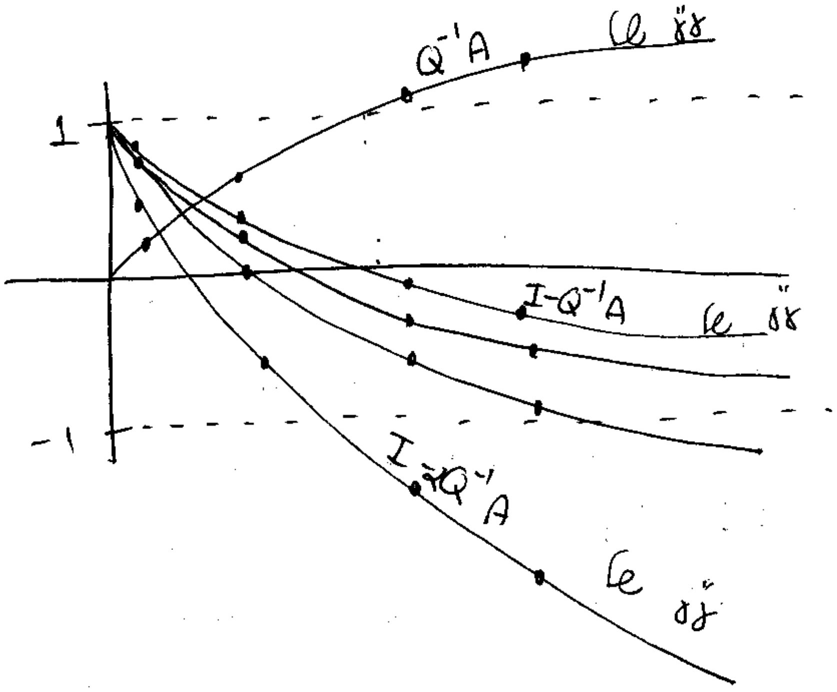
באופן כללי עם נורמה מתקיים
$$\|e^{n+1}\|_\square \le \|e^n\|_\square \|I-\alpha Q^{-1}A\|_\square$$
כדי שזה יתכנס אנחנו צריכים ש- $\|I-\alpha Q^{-1}A\|_\square \lneq 1$
אנחנו נניח ש-A ו-Q מוגדרות חיוביות ונראה מה צריך
להתקיים כדי שזה יתכנס?

Successive Over Relaxation - $Q=\frac{1}{\alpha}I$

ω-Jacobi - $Q=\frac{1}{\alpha}diag(A)$ (פה מניחים בנוסף ש- $0<diag(A)$)

אם אנחנו ב- l_2 אז $\|I-\alpha Q^{-1}A\|_2 \le \rho(I-\alpha Q^{-1}A) = 1-\alpha\lambda_{min}$
כאשר λ_{min} זה מינימלי של $Q^{-1}A$.
אנחנו רוצים למצוא את ה-α האופטימלי:
$$\min_\alpha \underbrace{\max_i|1-\alpha\lambda_i|}_{=\rho(I-\alpha Q^{-1}A)}$$
(λ_i חיוביים כי A,Q מוגדרות חיוביות)

נתבונן בגרף של הע"ע (פשוט ממויינים מהקטן לגדול)



אנחנו רוצים לבחור איזה α רק שהגרף לא יוצא מגבולות
[-1,1]. נרצה לבחור α כזה רק ש-

$$1-\alpha\lambda_{min} = -1+\alpha\lambda_{max}$$

למה? אנחנו רוצים שהגרף שנבחר יהיה כמה שיותר
רחוק מ-1 ומ-1-. המרחק מ-1 הוא $1-\alpha\lambda_{min}$
והמרחק מ-1- הוא $-1+\alpha\lambda_{max}$ אז
אנחנו נשווה אותם. ונקבל

$$\alpha = \frac{2}{\lambda_{max}+\lambda_{min}}$$

בספרות הרבה פעמים בוחרים $\alpha = \frac{2}{\lambda_{max}}$ משתי סיבות:

- צריך לחשב פחות
- הרבה פעמים λ_{min} מאוד קרוב לאפס.

קצב ההתכנסות שמתקבל במקרה הזה (נוי)

$$1-\frac{2\lambda_{min}}{\lambda_{min}+\lambda_{max}} = 1-\frac{2}{\frac{\lambda_{max}}{\lambda_{min}}+1} = 1-\frac{2}{K+1} =$$

$$= \frac{K+1-2}{K+1} = \frac{K-1}{K+1} =$$ כמו ב-SD

Preconditioning

יש בעיות שבהן $\kappa(A)$, $\kappa(Q^{-1}A)$ גבוהים וזו אפילו
תלויים ב- A וזה ממש גרוע!

יש לנו מע' משוואות $Ax=b$ ואם Q הפיכה אז
ו- $Q^{-1}Ax=Q^{-1}b$ יש בדיוק אותם פתרונות. אנחנו נרצה
למצוא Q כזאת שהופכת את מע' המשוואות לפשוטה
מאוד.

אבל אולי נצא בעיה כי עדיין $Q^{-1}A$ לא סימטרית
ואי אפשר לתת אותה ל- SD או ל- CG.

אז נתבונן ב: $E^TAE\hat{x}=E^Tb$. אם נפתור את זה אז
$x=E\hat{x}$ פותר את המערכת שלנו. ושוב, התקווה שלנו
היא של- E^TAE יש κ טוב יותר.

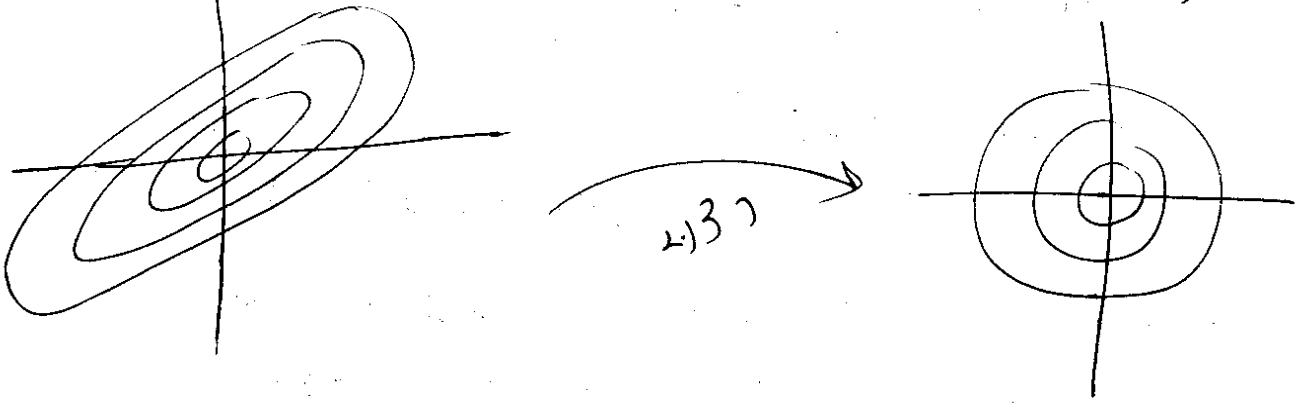
אם $Q^{-1}=EE^T$; $Q^{-1}Av=\lambda v$ אז
$E^{-1}v$ ו"ע של E^tAE עם ע"ע λ.

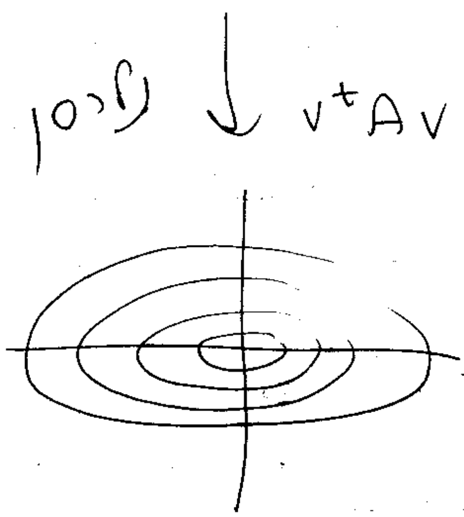
לכן מפני אין העדפה ל- preconditioning ע"י Q^{-1}
או ע"י E. זה למעשה שקול.

אם משתמשים ב- ω-Jacobi או SOR מתאים להשתמש
ב- Q^{-1} (זה הוא מוגדר תמיד) ואם משתמשים
ב- CG או SD אז מתאים להשתמש ב- E.

$$E^tAE\cdot(E^{-1}v)=E^tAv=E^{-1}EE^tAv$$
$$=E^{-1}Q^{-1}Av=E^{-1}\lambda v=\lambda(E^{-1}v)$$
$$\Leftarrow \lambda_{max}(Q^{-1}A)=\lambda_{max}(E^tAE)$$

איך נראית מטריצה עם κ גבוה?





אם A סימטרית ומוגדרת חיובית אז:

אם $D = V^tAV$ אלכסונית עם ע"ע חיוביים אז

$$\sqrt{D}\sqrt{D} = V^tAV$$

$$\Rightarrow I = \sqrt{D}^{-1}V^tA\cdot V\sqrt{D}^{-1}$$

אז אם נבחר $E = V\sqrt{D}^{-1}$ אז היא עושה את העבודה מעולה כי אז $I = E^tAE$

וקיבלנו מע׳ משוואות <u>מאוד</u> פשוטה.

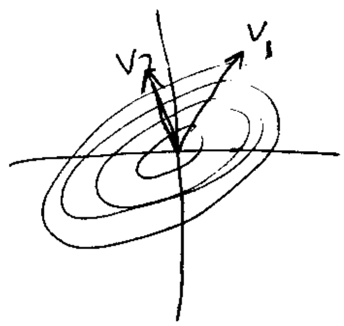
אבל בפועל כמובן לא כזה פשוט למצוא את V המלכסנת...

בפועל מבצעים את התהליך בשני שלבים:

1. Isolation - מציאת הצירים ש-A מכווצת/מותחת
2. Rescale - התיקון של A על הצירים הללו - מתיחת הצירים ש-A מכווצת ולהפך.

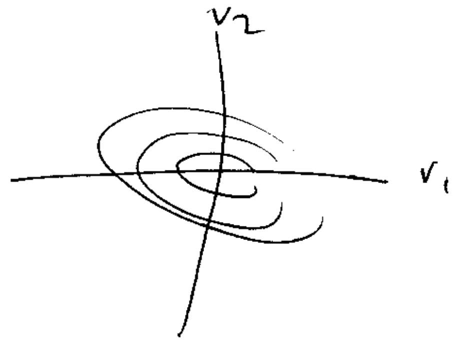
ואת זה נרצה לעשות ובדיוק לא בעזרת לכסון, אלא בשיטה יותר קלה חישובית.

נניח שמצאנו וקטורים v_1, v_2 שהם קרובים לו"ע.

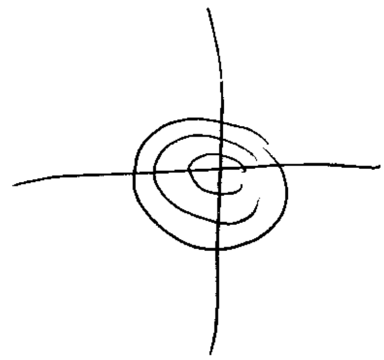


$$V^tAV \longrightarrow$$

$$V = \begin{pmatrix} | & | \\ v_1 & v_2 \\ | & | \end{pmatrix}$$



מתיחה ב- v_2
וכיווץ ב- v_1



① מציאת מע' צירים V
"דומה" לו"ע של A
במובן ש- A מכוונת/
מותחת את הצירים הללו

② Rescale ע"י

$$R = V^t A V$$

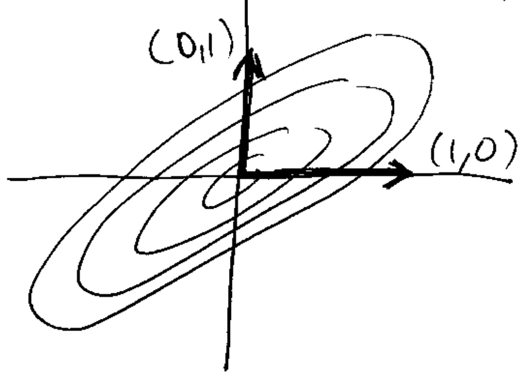
$$E = V \sqrt{diag(R)}^{-1}$$

(אם V במקור ו"ע זה
בדיוק מה שעשינו קודם)

שמצאנו ב- (1)

דוגמה למה שקורה אם ה-V-ים
הם לא טובים:

$$A = \begin{pmatrix} a & b \\ b & a \end{pmatrix}$$



כאן הצירים x ו- y
לא מתכוונים או נמתחים.
אז פה $V=I$ ואז

$$V^t A V = A$$

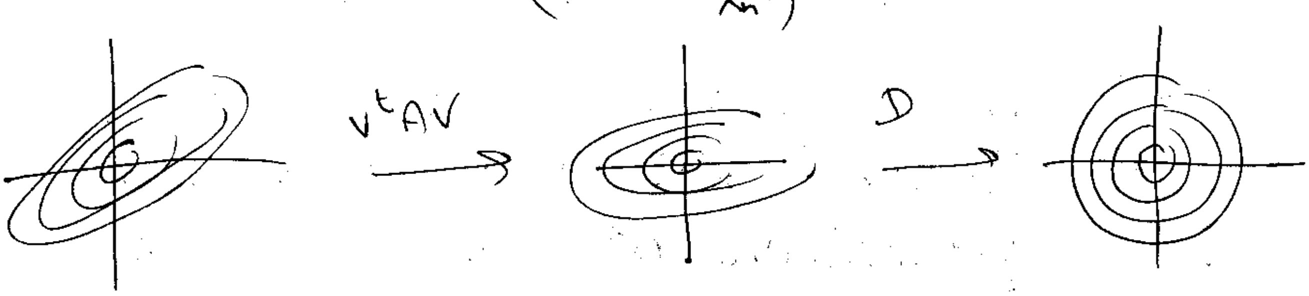
(41) 20.12.09
חשבון נומרי

Preconditioning - המטרה היא להוריד את $K(A)$.
(כי יש בעיות שבהן $K(A) \approx O(N^2)$ ואז אפילו ש-
$$W_{CG} = \sqrt{2}\,\frac{\sqrt{K}-1}{\sqrt{K}+1}$$
ה- CG לא עובד יותר מדי מהר... אז צריך לעשות משהו - א!

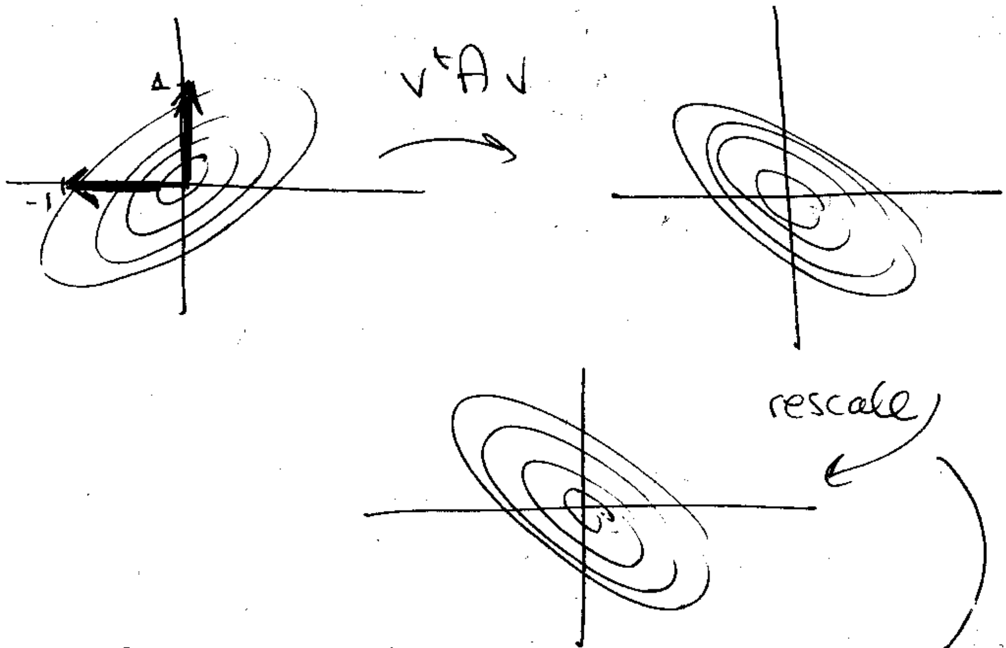
יש לנו מע' משוואות $Ax = b$. אם Q הפיכה זה
בדיוק כמו $Q^{-1}Ax = Q^{-1}b$.
או לחילופין אם נסמן $x = Ey$ אז אפשר לפתור את
$E^tAEy = E^tb$. היתרון של זה הוא שזה אומר
סימטריות ומוגדרות חיוביות ואז אפשר להשתמש ב- CG

אם $Q^{-1} = EE^t$ אז ל- $Q^{-1}A$ ול- E^tAE
יש אותם ע"ע ולכן גם אותו K אז אין יתרון בבחירה
לאחת השיטות על השנייה. אבל לפעמים אין ל- Q^{-1} פירוק
כזה ואז זה לא בטוח יצא אותו דבר.

Preconditioning בנוי קונספטואלית משני שלבים:
① isolation - מציאת מע' קואורדינטות חדשה אידיאלית
V - מט' ו"ע עצמיים. $D = diag(V^tAV)$
② rescaling - $\sqrt{D}^{-t}V^tAV\sqrt{D}^{-1}$ אידיאלית V הייתה
מלכסנת ואז זה היה מביא אותנו למטריצת יחידה אבל
זה כנראה לא יקרה בד"כ.
נניח ש-V מכילה את הו"ע של A כווקטורי השורה:
$$V^tAV = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$$



זה ממש מעולה בתיאוריה. הבעיה היא שפרקטית יכולה להיות לנו בחירה לא טובה של מע' צירים: למשל,



כאן לא קרה כלום כי אין מה לכייל כל מע' הצירים החדשה יש אותו ערך לשני הכיוונים ואז ה-rescaling מקסימום תכפול הכל בסקלר אבל זה לא "מתח" שונה בכיוונים שונים.

ה<u>אנרגיה</u> של וקטור x היא $E(x)=x^tAx=\|x\|_A^2$ (energy norm).

אם v_i וקטור עצמי מנורמל אז $E(v_i)=\lambda_i$

אז אם בוחרים שני וקטורים עם אותה אנרגיה זו בחירה <u>רעה</u> של מע' צירים. אנחנו מחפשים מע' צירים חדשה שיש לה אנרגיות מנורמלות - נמוכות ודומות ואז בשלב המתיחה מאוד קורה משהו משמעותי.

<u>הבחנה</u>: אם בוחרים $V=I$ אז $D=\text{diag}A$ ואז למעשה קיבלנו את שיטת Jacobi. אז השיטה הזאת היא בעצם preconditioning מסוים.

אם אנחנו מכירים איזו תכונה שמשפיעה על האנרגיה
אפשר לנסות להשתמש בה כמדריך לבחירת כיוונים טובים.

(Preconditioned CG) - PCG

☆ transformed PCG - נרצה להפעיל את CG

על מע' המשוואות $E^tAE\hat{x}^n = E^tb$, כאשר $\hat{x}^n = E^{-1}x^n$; אז

$$\hat{r}^n = E^tb - E^tAE\hat{x}^n = E^t(b - AE\hat{x}^n) = $$
$$= E^t r^n$$

$$\Rightarrow \quad \hat{r}^n = E^t r^n$$

נכתוב אז את CG עבור מע' המשוואות $E^tAE\hat{x} = E^tb$:

$$\hat{d}^0 = \hat{r}^0 = E^tb - E^tAE\hat{x}^0$$
$$\alpha = \frac{\hat{r}^{nt}\hat{r}^n}{\hat{d}^{nt}E^tAE\hat{d}^n}$$
$$\hat{x}^{n+1} = \hat{x}^n + \alpha\hat{d}^n$$
$$\beta = \frac{\hat{r}^{n+1\,t}\hat{r}^{n+1}}{\hat{r}^{nt}\hat{r}^n}$$
$$\hat{r}^{n+1} = \hat{r}^n - \beta E^tAE\hat{d}^n$$
$$\hat{d}^{n+1} = \hat{r}^n + \beta\hat{d}^n$$

untransformed PCG ☆

נפתור את הבעיה של המשתנים

$$\hat{d} = E^{-1}d, \quad \hat{x} = E^{-1}x, \quad \hat{r} = E^tr$$

במקום של המשתנים המקוריים.

$$r^0 = E^{-t}\hat{r}^0 = b - AE\hat{x}^0 = b - Ax^0$$
$$d^0 = E\hat{d}^0 = E\hat{r}^0 = EE^tr^0 = Q^{-1}r^0$$
$$\alpha = \frac{(r^{nt}E)E^tr^n}{d^{nt}E^{-t}E^tAEE^{-1}d^n} = \frac{r^{nt}Q^{-1}r^n}{d^{nt}Ad^n}$$

$$x^{n+1} = E\hat{x}^{n+1} = E(\hat{x}^{n} + \alpha \hat{d}^{n}) =$$
$$= E E^{-1} x^{n} + \alpha E E^{-1} d^{n} = x^{n} + \alpha d^{n}$$

$$r^{n+1} = E^{-t}\hat{r}^{n+1} = E^{-t}\hat{r}^{n} - \alpha E^{-t} E^{t} A E \hat{d}^{n} =$$
$$= \cancel{E^{-t} E^{t}} r^{n} - \alpha A d^{n} = r^{n} - \alpha A d^{n}$$

$$\beta = \frac{\hat{r}^{n+1\,t}\, \hat{r}^{n+1}}{(\hat{r}^{n\,t})\, \hat{r}^{n}} = \frac{r^{n+1\,t} E E^{t} r^{n+1}}{r^{n\,t} E E^{t} r^{n}} = \frac{r^{n+1\,t} Q^{-1} r^{n+1}}{r^{n\,t} Q^{-1} r^{n}}$$

$$d^{n+1} = E\hat{d}^{n+1} = E(\hat{r}^{n+1} + \beta \hat{d}^{n}) =$$
$$= E E^{t} r^{n} + \beta E E^{-1} d^{n} = Q^{-1} r^{n} + \beta d^{n}$$

אז אנחנו יכולים לרשום הכל בצורת r, x, d בלי
שנצטרך להשתמש ב- E. אבל אלה המשתנים
המקוריים של הבעיה (עוד יותר להבהיר את נכונות
הקשר בין d ל- $\hat{d}$!) וזה נחמד כי עכשיו
אנחנו לא צריכים לפרק את Q^{-1} ל- EE^{t}. מספיק
רק לדעת שיש פירוק כזה. ועכשיו אפשר להריץ
את CG כמו שאנחנו מכירים אותו.

רצינו לפתור $Ax = b$
והפכנו את זה ל- $E^{t} A E \hat{x} = E^{t} b \quad (**)$
ראינו שאם נגדיר $x = E\hat{x}$ אז אפשר
אפשר לעשות פעולות כאלה שנותנות לי x כזה
שפותר את $(**)$ ע"י $x = E\hat{x}$. אבל
זה אותו x שפותר את $Ax = b$. מעולה!

(43) 24.12.09
חישוב נומרי

CG	PCG (untransformed)
$r^0 = b - Ax^0$	$r^0 = b - Ax^0$
$d^0 = r^0$	$d^0 = M^{-1} r^0$
$\alpha = \frac{r^{n t} d^n}{d^{n t} A d^n}$	$\alpha = \frac{r^{n t} M^{-1} r^n}{d^{n t} A d^n}$
$x^{n+1} = x^n + \alpha d^n$	$x^{n+1} = x^n + \alpha d^n$
$r^{n+1} = r^n - \alpha A d^n$	$r^{n+1} = r^n - \alpha A d^n$
$\beta = \frac{(r^{n+1})^t r^{n+1}}{r^{n t} r^n}$	$\beta = \frac{(r^{n+1})^t M^{-1} r^{n+1}}{r^{n t} M r^n}$
$d^{n+1} = r^{n+1} + \beta d^n$	$d^{n+1} = r^{n+1} + \beta d^n$

ה- PCG מתקבל ע"י פתרון של $E^t A E \hat{x} = E^t b$, $M^{-1} = E E^t$

והגדרה $E\hat{x} = x$, $\hat{x} = E^{-1} x$

$E^t r = \hat{r}$, $r = E^{-t} \hat{r}$

$E\hat{d} = d$, $\hat{d} = E^{-1} d$

אנו יודעים שזה מחזיר אותנו למשתנים המקוריים וה- x ב- PCG הוא פתרון של $Ax = b$.

לחילופין, ניתן היה לפתח את CG אוקלידית מכפלה אחרת - לא לבחור כיווני חיפוש r^n אלא $M^{-1} r^n$

לסיכום, נסתכל על דוגמה מפורסמת.

Poisson / Diffusion 1D

יש פונקציה $f(x,t)$ ורוצים לפתור את המשוואה

$$\frac{\partial f}{\partial t} = -\Delta_x f \quad , \quad \Delta_x f = b$$

(א) משוואת החום: $\frac{\partial f}{\partial t}(x,t) = -\frac{\partial^2 f}{\partial x^2}(x,t)$ - כאן f היא דו-ממדית

(ב) פואסון: $\frac{\partial^2 f}{\partial x^2}(x) = b(x)$ ← כאן f היא חד-ממדית

$$f(x+\Delta x) = f(x) + \Delta x f'(x) + \frac{\Delta x^2}{2} f''(x) + \frac{\Delta x^3}{6} f'''(x) + O(\Delta x^4)$$
$$f(x-\Delta x) = f(x) - \Delta x f'(x) + \frac{\Delta x^2}{2} f''(x) - \frac{\Delta x^3}{6} f'''(x) + O(\Delta x^4)$$
$$\Rightarrow f''(x) = \frac{f(x+\Delta x) - 2f(x) + f(x-\Delta x)}{\Delta x^2} + O(\Delta x^2)$$

$$f(t+\Delta t) = f(t) + \Delta t f'(t) + O(\Delta t^2)$$
$$f(t) = f(t)$$
$$\Rightarrow f(t+\Delta t) - f(t) = \Delta t f'(t) + O(\Delta t^2)$$
$$\Rightarrow f'(t) = \frac{f(t+\Delta t) - f(t)}{\Delta t} + O(\Delta t)$$

משוואת פואסון דיסקרטית:

$$\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2} = b_i$$ ([illegible])

משוואת החום הדיסקרטית

$$\frac{f_{i,j+1} - f_{ij}}{\Delta t} = \frac{f_{i+1,j+1} - 2f_{i,j+1} + f_{i-1,j+1}}{\Delta x^2}$$

אנחנו רוצים להביא את זה בצורה מטריציונית

אז

פואסון

$$L = \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ 1 & & & 1 & -2 \end{pmatrix} \frac{1}{\Delta x^2}$$

נעביר אגפים במשוואת החום

$$\frac{f_{ij+1}}{\Delta t} + \frac{f_{i+1,j+1} - 2f_{i,j+1} + f_{i-1,j+1}}{\Delta x^2} = \frac{f_{ij}}{\Delta t}$$

$$\Rightarrow f_{i,j+1} + \frac{\Delta t}{\Delta x^2}\left(f_{i+1,j+1} - 2f_{i,j+1} + f_{i-1,j+1}\right) = f_{ij}$$
$$\Rightarrow I + \frac{\Delta t}{\Delta x^2} L)\, \vec{f}_{j+1} = \vec{f}_j$$

(94)

אז יש לנו $L\bar{f}=\bar{b}$.

$$\left(I+\frac{\Delta t}{\Delta x^2}L\right)\bar{f}_{j+1}=\bar{f}_j$$

נשים ♡ ש- v_i ו"ע של L עם ע"ע λ_i אמ"מ
v_i ו"ע של $I+\frac{\Delta t}{\Delta x^2}L$ עם ע"ע $1+\frac{\Delta t}{\Delta x^2}\lambda_i$

נמצא ו"ע - $\quad v_n^k=e^{-ink\frac{2\pi}{N}} \quad (L\in M_N(\mathbb{R}))$

$$(Lv^k)_m = e^{-\frac{i(m-1)k\cdot 2\pi}{N}}-2e^{-\frac{imk2\pi}{N}}+e^{-\frac{i(m+1)k2\pi}{N}}=$$

$$=\underbrace{e^{-\frac{imk2\pi}{N}}}_{v_m^k}\underbrace{\left[e^{\frac{ik2\pi}{N}}-2+e^{-\frac{ik2\pi}{N}}\right]}_{\text{לא תלוי ב-}m\text{ אלא זה הע"ע של }v^k}=$$

$$\Rightarrow \quad v_m^k=e^{-\frac{imk2\pi}{N}}$$

$$\lambda^k = 2\cos(k\cdot 2\pi/N)-2$$

נשים ♡ ש- L היא למעשה מטריצה של פעולת קונבולוציה עם $(0..\ 0\ 1\ -2\ 1\ ..)$ אז יש וקטור l רק ש- $l*f=b$. נעבור לפורייה $\hat{l}\cdot\hat{f}=\hat{b}$
$\Leftarrow$ $\hat{f}=\frac{\hat{b}}{\hat{l}}$. זה נקרא Fast Fourier Solver
והוא לוקח $O(N\log N)$ כי הוא משתמש ב- FFT
כדי לפתור באופן מדויק. אבל אנחנו מתעניינים בשיטות איטרטיביות...

L היא לא הפיכה (היא מאפסת את הוקטור $\vec{1}$).
אבל בתרגיל ראינו שזה לא משנה. CG דווקא מוצא את הפתרון הכי טוב מאלה שכן אפשר להגיע אליהם.
אז במקום להסתכל על $K(L)=\frac{4}{0}$ לא נסתכל
על 0 אלא על הע"ע שאחריו.

$$\frac{4}{2\cos(\frac{2\pi}{N})-2}\approx\frac{4}{2(1-\frac{4\pi^2}{N^2})-2}=2\frac{N^2}{\pi^2}=O(N^2)$$

אז $K=O(N^2)$. וזה לא משהו...

אז

$$\omega_{SD} = \frac{K-1}{K+1} = \frac{N^2-1}{N^2+1} = \frac{N^2}{N^2+1} - \frac{1}{N^2+1} \le \frac{N^2}{N^2+1}$$

$$\|e^n\| \le \|e^0\| \, \omega_{SD}^n \le \|e^0\| \left(\frac{N^2}{N^2+1}\right)^n =$$

מתי?

אם $n \sim N^2$ אז

$$\left(\frac{N^2}{N^2+1}\right)^n \sim \frac{1}{e}$$

זה משהו....

→ אנחנו צריכים לעשות preconditioning

וזה בשיעור הבא...

(45) 27/12/09

חישוב נומרי

אנחנו נמצאים בעיצומה של דוגמה של משוואת פואסון ומשוואת החום.

$$\frac{\partial f(x,t)}{\partial t} = -\frac{\partial^2 f(x,t)}{\partial x^2} \quad \Big| \quad \frac{\partial^2 f(x)}{\partial x^2} = b(x)$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\left(I + \frac{\Delta t}{\Delta x^2} L\right)\bar{f}_{j+1} = \bar{f}_j \quad \Big| \quad L\bar{f} = \bar{b}$$

נתרכז בעיקר ב- $L\bar{f} = \bar{b}$.

ראינו שיש וקטורים עצמיים וקטור $V_j^k = e^{\frac{-ijk\cdot 2\pi}{N}}$

עם ערכים עצמיים $\lambda_k = 2(\cos\frac{\pi k}{N} - 1)$ רכיב

ואז $K(L) \approx O(N^2)$

אנחנו יודעים ש- $(\omega_{SD})^n = \left(\frac{K-1}{K+1}\right)^n = O(\|e^n\|_A)$

אם לוקחים $n = \frac{aK}{2}$, כלומר מספר האיטרציות פרופורציוני ל- K אז נקבל שהשגיאה היא

$$\left(\frac{K+1}{K+1} - \frac{2}{K+1}\right)^n = \left(1 - \frac{2}{K+1}\right)^n \approx \left(1 - \frac{2}{K}\right)^{\frac{aK}{2}} \xrightarrow[K\to\infty]{} e^{-a}$$

אבל הנקודה היא שזה חסם הדוק. כדי לקבל שגיאה קבועה בהכרח צריך שמספר האיטרציות יהיה פרופורציוני ל- K ואם $K \sim O(N^2)$ אז זה די גרוע...

לכן אנחנו צריכים לעשות preconditioning כדי רצ׳ני ל- L.

נשים ♡ שע"ע נמוכים מתאימים לו"ע שבניהם תדרים נמוכים ולהפך.

כמובן, נשים ♡ ש- $L = -D^t D$ כאשר

(D מייצגת נגזרת ראשונה ; L- היא נגזרת שנייה)

$$D^t = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \\ -1 & & & & 1 \end{pmatrix}$$

האנרגיה ש- L משרה היא

$$E(x) = x^t L x = x^t D^t D x = \|Dx\|^2$$

אז יש קשר בין האנרגיה לנגזרת - שינוי גדול ← אנרגיה גדולה. שינוי קטן ← אנרגיה קטנה.

<u>בסיס היררכי</u> (Multi Grid Methods)

$$V_i^0 = [0 \dots 0\ \underset{\overset{\nearrow}{2i}}{1}\ 0 \dots 0] \quad \frac{N}{2} \text{ וקטורים}$$

$$V_i^1 = [0 . 0\ \tfrac{1}{2}\ \underset{\overset{\nearrow}{4i}}{1}\ \tfrac{1}{2}\ 0 . 0] \quad \frac{N}{4} \text{ וקטורים}$$

$$V_i^2 = [0 \dots 0\ \tfrac{1}{4}\ \tfrac{1}{2}\ \tfrac{3}{4}\ \underset{\overset{\nearrow}{8i}}{1}\ \tfrac{3}{4}\ \tfrac{1}{2}\ \tfrac{1}{4}\ 0 \dots 0] \quad \frac{N}{8}$$

לבסיס הזה אנחנו רוצים לעבור. הוא יהיה טוב לנו אם רמות האנרגיה יהיו שונות. האנרגיה היא

$$E(v) = \frac{v^t L v}{v^t v} = \frac{v^t D^t D v}{v^t v}$$

בכל רמה בווקטורים שלנו יש אותה אנרגיה כי הם פשוט הזזה אחד של השני. אז נסמן ב- v^j נציג כלשהו מרמה j. אז

$$v^t v = \sum_{i=1}^{N} (v_i^j)^2 \approx 2^{j+1} \int_0^1 x^2 dx = \frac{2^{j+1}}{3}$$

$$v^t L v = v^t D^t D v = \|Dv\|^2 = \underbrace{2^{-2j}}_{\text{גודל ההפרש}} \cdot \underbrace{2^{j+1}}_{\text{מספר ההפרשים}} = 2^{1-j}$$

$$\Rightarrow E(v^j) = \frac{3 \cdot 2^{1-j}}{2^{j+1}} = 3 \cdot 2^{-2j}$$

האנרגיות הכי נמוכות מתקבלות ב- $j = \log N$ ואז

$$E(v) = 3 N^{-2}$$

שלה מסדר כמו הו"ע של הווקטור עם הערך ע"ע הגדול ביותר. אז זה נחמד. (אם כי 2 לא הורחה שלמה).
בתרגול נראה משהו קצת פשוט יותר שעובד פחות טוב.

בכל מקרה כדי שנוכל להשתמש בבסיס הזה אנחנו צריכים לדעת להכפיל מהירות מווקטורים האלה.
בכל רמה יש N איברים ששונים מ-0 ויש $\log N$ רמות אז במטריצה V יש $N\log N$ איברים ששונים מ-0.
אבל יש אלג' Fast Cascade שיודע לפתור את זה ב-$O(N)$, כלומר יודע להעביר וקטורים למע' הצירים החדשה ב-$O(N)$.

פירוק לע"ע וו"ע

נניח של-A יש n ע"ע שונים (ובפרט A לכסינה).

השיטה הישירה היא לפתור את $\det(A-\lambda I)=0$
אבל זה ממש קשה, לא רק שמדובר במימדים גבוהים, אלא שגם צריך לפתור פולינום ממעלה גבוהה וזה פשוט קשה ממש ממש.

הנה שיטה נחמדה:

<u>Power Method:</u>

u^0 - initial guess

$u^{n+1} = Au^n$

$u^{n+1} = u^{n+1}/\|u^{n+1}\|$

נניח ש- $v_1,\dots,v_N$: ו"ע ; $u^0=\sum v^k a_k$

$$\Rightarrow u^1 = \frac{\sum a_k\lambda_k v^k}{c_1}, \dots, u^n=\frac{\sum a_k v^k\lambda_k^n}{c^n}$$

קבוע נרמול $\rightarrow c^n$

אם $\lambda_1 > \dots > \lambda_N$ אז

$$\lambda_1^n \gg \dots \gg \lambda_N^n$$

אז הטענה היא שאחרי כמה צעדים של ההכפלות

מי שהכי דומיננטי הוא λ_1^n והאחרים זניחים ביחס אליו.

ואז $u^n \xrightarrow[n\to\infty]{} v^1$

אבל אנחנו לא יכולים לעשות אינסוף איטרציות ...

מה הדיוק וכמה איטרציות צריך?

מי שהכי מפריע לדיוק הוא הע"ע השני בגודלו.

אנחנו רוצים ש- $\lambda_2^n \ll \lambda_1$ רק שגם λ_2 יהיה זניח...

מתי $\frac{\lambda_2^n}{\lambda_1^n} < \varepsilon$?

זה יראה לנו את מספר האיטרציות שצריך לעשות...

$$\log \frac{\lambda_2^n}{\lambda_1^n} \le \log \varepsilon$$

$$\Rightarrow n \ge \frac{\log \varepsilon}{\log \lambda_2 - \log \lambda_1}$$

נשים לב ש- $\log \varepsilon < 0$, אבל $\log \lambda_1 > \log \lambda_2$

כך זה מסתדר... וככל ש-ε קטן n גדל...

אגב- מה שמעניין אותנו פה זה למעשה הערכים המוחלטים של הערכים העצמיים. אז הכל נכון מוגדר.

הערך $\frac{\lambda_2}{\lambda_1}$ נקרא הפער הספקטרלי. אילו הייתה לנו דרך להעריך אותו היינו יכולים לדעת כמה איטרציות צריך. אבל אנחנו לא יודעים אותו...

(47) 31/12/09
חישוב נומרי 2

פירוק ספקטרלי

אנחנו מניחים ש-A יש n ע"ע שונים (בפרט היא לכסינה) ונניח גם שהיא נורמלית.

ראינו את שיטת החזקות:

Power Method

u^0 - initial guess

$u^{n+1} = A u^n$

$u^{n+1} = u^{n+1} / \|u^{n+1}\|$ ← הנרמול הזה חשוב משיקולים נומריים.

נניח ש- $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$ והו"ע המתאימים $v_1, \dots, v_n$.

ונניח ש- $u^0 = \sum a_i v_i$. אז

$u^{k+1} = \sum a_i \lambda_i^k v_i / c_n$ ← נג מל

מכיוון ש- $|\lambda_n|^k << \dots << |\lambda_1|^k$ בסוף זה יתכנס לו"ע שמתאים ל- λ_1.

יש סיכוי ש- $a_1 = 0$ ואז לא נקבל את v_1. אז בד"כ מגרילים את u^0 באופן אקראי ואז יוצא שההטלה על כל הקואורדינטות היא מאוד דומה. בפועל, כל הסיפור הזה מאוד רגיש נומרית והשיטה מתכנסת אפילו אם a_1 קטן מאוד מאוד.

ראינו ש- $N \cdot \left(\frac{\lambda_2}{\lambda_1}\right)^n$ הוא חסם של השגיאה בצעד ה- n לכן אם רוצים שגיאה קטנה מ- ε צריך מספר איטרציות

$$n = \frac{\log \varepsilon}{\log|\lambda_2| - \log|\lambda_1|}$$

מקרה פתולוגי: נניח ש- $\lambda_1 = 1$; $\lambda_2 = 1 - \frac{1}{N^2}$. אז

החסם שלנו הוא

$$\left(1-\frac{1}{N^2}\right)^n \cdot N = \Big(\underbrace{\left(1-\frac{1}{N^2}\right)^{N^2}}_{e^{-1}}\Big)^{\frac{n}{N^2}} \cdot N \approx e^{-\frac{n}{N^2}} \cdot N$$

ולכן מספר האיטרציות צריך להיות בסדר גודל של N^2 כדי שיצא משהו נורמלי. ואם N גדול אז זה חבל.

את השיטה הזאת אפשר לשדרג קצת כדי שנוכל לחשב לא רק את הו"ע של הע"ע הכי גדול אלא כמה ו"ע.

Power-Method for m Eigenvectors

```
k = 1, ..., m
    for n = 1, ...
        u_k^{n+1} = A u_k^n
        u_k^{n+1} = u_k^{n+1} - sum_{j=1}^{k-1} u_j <u_j, u_k^{n+1}>
        u_k^{n+1} = u_k^{n+1} / ||u_k^{n+1}||
    end
    u_k = u_k^n
end
```

האלטרנטיבה הפשוטה יותר היא

```
k = 1, ..., m
    u_k^0 = ניצב לכל u_j הקודמים
    for n = 1, 2, ...
        u_k^{n+1} = A u_k^n
        u_k^{n+1} = u_k^{n+1} / ||u_k^{n+1}||
    end
    u_k = u_k^n
end
```

(48)

זה קל למימוש אבל זה מאוד מאוד לא יציב נומרית.

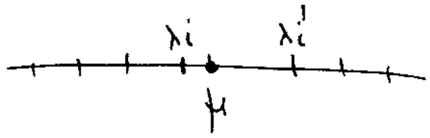
אז אנחנו יודעים לחשב ע"ע גדולים. איך נחשב קטנים?
נתבונן ב- $\lambda_{max} I - A$ והע"ע הגדולים
כאן מתאימים לע"ע הקטנים של הבעיה המקורית A.
נניח שלא יודעים את λ_{max} בדיוק? האם כדאי
לנו לרפות בחסם גדול יותר? לא! כי אז אנחנו
מקטינים את הפער הספקטרלי. אז חשוב שנדע
את λ_{max} בדיוק טוב..

הסבר יותר מדויק: הפעלת PM על $\lambda_{max} I - A$
נותנת את הע"ע הכי קטנים (לא בערך מוחלט) של A.
אז אם למשל $\lambda_{max} \approx -\lambda_{min}$ זה לא נותן לנו הרבה
כי בלאו הכי היינו מקבלים גם את λ_{min} מ- PM על
A כי הערך המוחלט שלו גדול.
אבל אם A מוגדרת חיובית אז כל הע"ע שלה חיוביים
וה- PM על A נותנת ע"ע גדולים בלבד (השאר
קרובים לאפס). ואז $\lambda_{max} I - A$ באמת נותן את
הקטנים וזה ממש מוסיף לנו מידע חדש.

Inverse Power Method (u^0, μ)

solve $(A-\mu I)u^{n+1} = u^n$ // i.e. $u^{n+1} = (A-\mu I)^{-1} u^n$
$u^{n+1} = u^{n+1} / \|u^{n+1}\|$

זה ה- PM כאשר הוא מחשב ע"ע של $(A-\mu I)^{-1}$
שהע"ע שלה הם $\frac{1}{\lambda_i-\mu}$ וזה ממש מגניב כי עכשיו
זה אומר שאנחנו מוצאים λ_i רק ש- $\frac{1}{\lambda_i-\mu}$ גדול
שזה אומר ש- $\lambda_i \sim \mu$. אז מצאנו דרך למצוא ע"ע
שקרובים ל- μ כלשהו שבחרנו. ומה הקאץ'?



$$\frac{\lambda_i - \mu}{\lambda_i' - \mu} = \frac{\text{שני הכי גדול}}{\text{הכי גדול}}$$

והיחס הזה יכול להיות קטן כרצוננו - ככל שנבחר את μ להיות קרוב ל-λ_i.

יש מיני כללי אצבע לבחירת μ בצורה טובה.

נתראה שבוע הבא

(49) 3/11/10
חישובים נומריים

הערות לשאלה 4:

דוגמה לבסיס Haar ראשון $x \in \mathbb{R}^8$

$$\begin{array}{c|cccccccc}
 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
l_0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\
\hline
l_1 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\
 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\
\hline
l_2 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
\hline
l_3 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}$$

אבל ברמה השלישית אין טעם להסתכל כי זה בלאו הכי בגרעין.

דיברנו על PM ושידרגנו אותה ל- m-PM שמאפשר לחשב m ע"ע מקסימליים.
דיברנו על IPM שהיתרון העיקרי שלה הוא שאפשר לחפש ע"ע שקרובים לערך σ שמעניין אותנו ואז גם הו"עים ממש טובים. המחיר הוא שבכל צעד צריך לפתור מע' משוואות שלעיתים היא עצמה סובלת מ-K גרוע.

גם מקרה את σ כה אמרנו במקרה ש- A נורמלית.
הן גם מחשבות רק חלק מהע"ע ועובדות מהר רק עם המטריצות דלילות.

שיטת Jacobi Iteration מניחה ש- A סימטרית (שזה קצת יותר מחמיר) אבל היא מוצאת את כל הע"ע והרבה יותר יציבה..

אם A סימטרית אז יש לה מטריצה מלכסנת V,
$D = VAV^t$ כאשר V מטריצה סיבוב; השורות
של V הן הו"ע של A.
הרעיון הוא שבאופן איטרטיבי נמצא את V ע"י
הרכבה ממטריצות סיבוב פשוטות יותר.

```
V = I
while  Aij ≠ 0 ,  i≠j
      find R^ij  s.t  (R A R^t)ij = 0
      V = R V
      A = R A R^t
end
output   A  //  = D  - ע"ע
         V  //    ו"ע
```

$$
\begin{array}{cc}
 & \begin{array}{ccccccc} & & i & & j & & \end{array} \\
R^{ij} = & \begin{pmatrix}
1 & & & & & & \\
 & \ddots & & & & & \\
 & & \cos\theta & & \sin\theta & & \\
 & & & \ddots & & & \\
 & & -\sin\theta & & \cos\theta & & \\
 & & & & & \ddots & \\
 & & & & & & 1
\end{pmatrix}
\begin{array}{c} \\ \\ i \\ \\ j \\ \\ \\ \end{array}
\end{array}
$$

זו מטריצת סיבוב שיש לה דרגת חופש אחת.
אנחנו צריכים למצוא θ כזה שהמשוואה שלנו
תיפתר.

$$(R^{ij} A R^{ij^T})_{ij} = ?$$

כדי למצוא את האיבר ij מספיק להתמקד לחישוב

$$\left[\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}\begin{pmatrix} A_{ii} & A_{ij} \\ A_{ji} & A_{jj} \end{pmatrix}\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}\right]_{ij} =$$

נסמן $c = \cos\theta$, $s = \sin\theta$

$$=\left[\begin{pmatrix} cA_{ii}+sA_{ji} & cA_{ij}+sA_{jj} \\ -sA_{ii}+cA_{ji} & -sA_{ij}+cA_{jj}\end{pmatrix}\begin{pmatrix} c & -s \\ s & c\end{pmatrix}\right]_{ij} =$$

$$= -s^2A_{ji} - csA_{ij} + c^2A_{ij} + scA_{jj} =$$

$$= A_{ij}(c^2-s^2)+cs(A_{jj}-A_{ii}) \underset{\uparrow}{=} 0$$

זה מה שרצינו לפתור

עכשיו נעשה פישוטים טריגונומטריים. נשים ♡ ש-

$$\frac{c^2-s^2}{2cs} = \cot 2\theta$$

$$\Rightarrow \alpha = \cot 2\theta = \frac{A_{ii}-A_{jj}}{2A_{ij}}$$

זו משוואה שאנחנו יודעים לפתור!! Woohoo!

אלא שהפעול $\cot^{-1}$ גם שם זה יקר אז נעשה עוד מניפולציות.

נגדיר $t=\frac{s}{c}$. אז מתקיים $t^2+2t\alpha-1=0$

(קל לוודא. זו סתם זהות).

אז יש לנו משוואה ריבועית שהשורש הקטן שלה הוא

$$t = \frac{\text{sign}(\alpha)}{|\alpha|+\sqrt{\alpha^2+1}} \underset{\alpha}{\approx} \frac{1}{2|\alpha|}$$

α גדול גדול

לחשב שורש זה יותר מהיר מלחשב cot. למעשה, המעבדים של אינטל משתמשים בשיטת ניוטון כדי לעשות זאת.

נזכור גם ש- $c=(t^2+1)^{-1/2}$ ו- $s=c\cdot t$

ומרגע שיש לנו את s ואת c אנחנו יודעים לכתוב את המטריצה R.

מה הסיבוכיות?

יש N^2 איטרציות. פיתרון המשוואה לוקח $O(1)$

אבל צריך גם לעדכן $A = RAR^t$

$V = VR$

בגלל המבנה של R המכפלה VR ממש פשוטה ולוקחת $O(N)$ וכך גם העדכון של A.

אז סה"כ השיטה היא $O(N^3)$.

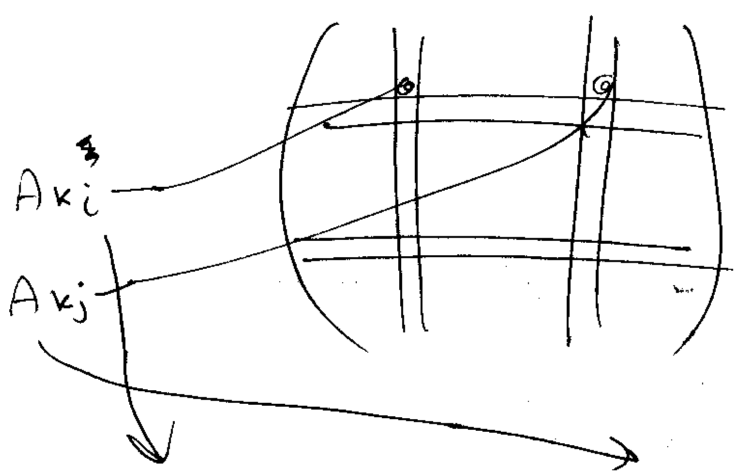
אבל האם השיטה נכונה בכלל? אנחנו רוצים שככל שעוברות איטרציות

$$\|A\|_F - \sum a_{ii}^2 \longrightarrow 0$$

נורמת פרובניוס $\sum_{i,j} a_{ij}^2$

נסתכל על איטרציה אחת. נתבונן רק במכפלה RA. אותו דבר גם יעבוד למכפלה מימין.

מה שעושים במכפלה RA הוא ערבוב של שתי שורות.



$$cA_{ki} + sA_{kj}\ ,\ cA_{kj} - sA_{ki}$$

בדיקה פשוטה מראה שסכום הריבועים שלהם נשאר זהה. אבל הרי איבר אחד התאפס ואת האיבר השני אנחנו לא סופרים. אז סה"כ יש ירידה מונוטונית ממש וזה בדיוק מה שרצינו.

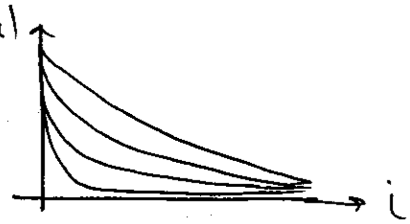
(51) 7/11/10

חישוב נומרי 2

היום נלמד מה באמת עושים כדי למצוא ע"ע. לא נלמד אותם לעומק ולא נוכיח דברים אבל הסיבות העיקריות שמשתמשים בהן ולא במה שלמדנו קודם הן יציבות נומרית ומהירות התכנסות. באתר הקורס יש לינק לספר שבו בפרק 6 יש הרחבה של התכנסות וחישוב הקצב. זה ארוך נורא...

שיטת ארנולדי

אם הע"ע הכי גבוה של A הוא 1 אז כשמפעילים אותה בהכפלה עדיין נשאר ע"ע 1 וכל השאר קטנים מעריכית.



שיטת ארנולדי מנסה איכשהו להשתמש בכל ערכי הביניים שחושבו - $Av_0, A^2v_0, \dots$

Arnoldi

v_1 - initial guess

for $j=1,\dots,m$

$$w_j = Av_j - \sum_{i=1}^{j} \overbrace{\langle Av_j, v_i\rangle}^{h_{ij}} v_i$$

$h_{j+1,j} = \|w_j\|$. if $h_{j+1,j} < \varepsilon$ then stop. — זה לא מפריע לסימון h_{ij} כי שם $i \le j$

$v_{j+1} = w_j / h_{j+1,j}$

end

אחרי שעושים את זה יש לנו

$$h_{j+1,j} \cdot v_{j+1} = Av_j - \sum_{i=1}^{j} h_{ij} v_i$$

$$\Rightarrow \quad Av_j = \sum_{i=1}^{j+1} h_{ij} v_i \qquad \forall\ 1 \le j < m+1$$

נגדיר

$$V^m = \begin{pmatrix} | & | & & | \\ v_1 & v_2 & \dots & v_m \\ | & | & & | \end{pmatrix}$$

$$H^m_{ij} = h_{ij}$$

$$\Rightarrow \underset{n\times n}{A}\underset{n\times m}{V^m} = V^m H^m + \Big(\overset{m-1}{0} \Big| v_{m+1} \Big) \cdot h_{m+1,m}$$

$$\Rightarrow V^{mt} A V^m = \cancel{V^{mt} V^m} H^m + V^{mt}\Big(0 \Big| v_{m+1}\Big) h_{m+1,m} =$$

$$= H^m + 0 = H^m$$

כי V^m אורתוגונלית

כי v_{m+1} מאונך V^{m-1}

מטריצת Hessenberg

בעזרת H^m נמצא ו"ע.

אם $H^m y_k = \lambda_k y_k$ אז $u_k = V^m y_k$

קירוב לוקטורים העצמיים של A.

המדד לשגיאה הוא הנורמה $\|(A-\lambda_k I)u_k\|_2$

נחשב:

$$A u_k = A V^m y_k =$$

$$= (V^m H^m + \boxed{|}\, h_{m+1,m})\, y_k =$$

$$= \lambda_k V^m y_k + y_k^{(m)} \cdot v_m \cdot h_{m+1,m} =$$

$$= \lambda_k u_k + y_k^{(m)} \cdot h_{m+1,m} \cdot v_m$$

$$\Rightarrow \|(A-\lambda_k I)u_k\|_2 = |y_k^{(m)} h_{m+1,m}|\, \overbrace{\|v_m\|}^{=1} =$$

$$= |y_k^{(m)} h_{m+1,m}|$$

לכן האיכות תלויה ב- $h_{m+1,m}$ וזה אכן היה המדד שלנו לעצירה. $y_k^{(m)}$ לא יכול להפריע לנו ממש כי הוא רכיב של ו"ע של מטריצה קטנה ואנחנו יכולים להניח שהוא מנורמל.

נפי שנה באמת יהיה טוב, ε דריך לקטון פרופורציונלית לגודל של הע"ע הכי קטן שאנחנו רוצים למצוא.

כי גם ε לא מספיק קטן $s\varepsilon$ אנחנו לא יכולים
לדעת אם הע"ע של הע"ע הקטנים לא התערבבו איכשהו.
המדד הזה של ע"ע נמוכים הוא נמוך...
מה הסיבוכיות של האלגוריתם?
כמו הרידה הוא $O(n\ m^2)$ עבור דלילות!
$O(n^2m^2)$ עבור מלאות
וסיבוכיות הזיכרון היא $O(nm)$
(זה מאוד מתאים למקרים שבהם $n \gg m$).

שיטת Lanczos

כמימוש שיטת ארנולדי במקרה ש-A סימטרית. אזי
$$|j-i| \le 1 \Rightarrow h_{ij} \neq 0$$
ותוך מזה מתברר שהיא סימטרית. נסמן $h_{ii} = \alpha_i$
$$h_{i+1,i} = h_{i,i+1} = \beta_i$$
זה מוריד את השיטה ל- $O(nm)$ או $O(n^2m)$
נכיח: $\langle Av_j, v_i\rangle \underset{A \text{ סימטרית}}{=} \langle v_j, Av_i\rangle$

$$Sp\{v^1, Av^1, Av^2, \dots, Av^{j-1}\} = Sp\{v^1, v^2, \dots, v^j\}$$
$$v^j \perp v^{j-1}, \dots, v^1 \Rightarrow v^j \perp Av^{j-2}, \dots, Av^1, v^1.$$
$$\Rightarrow \langle v_j, Av_i\rangle = 0 \quad \text{כאשר} \quad i < j-1$$

(53) 10/1/10
חישוב נומרי

QL- decomposition

רוצים למצוא פירוק $A=QL$ כאשר Q מטריצה סיבוב
אומר $Q^tQ=I$
ו- L משולשית תחתונה
(יש גם פירוק QR כאשר R משולשית תחתונה).

דרך 1: ע"י גראם-שמידט
נבצע גראם-שמידט על וקטורי העמודה של A- $a_{*1},\dots,a_{*n}$

$$q_{*1}=\frac{a_{*1}}{\|a_{*1}\|}$$

$$q_{*2}=\frac{a_{*2}-\langle a_{*2},q_{*1}\rangle q_{*1}}{\|\uparrow\|}$$

$\vdots$

ברור ש- Q יוצאת מטריצה טובה. (ראה ש- $L=Q^tA$
יוצאת מטריצה משולשית תחתונה. נשים ♡ ש-
$(Q^tA)_{ij}=\langle q_{*i},a_{*j}\rangle$ ומהבניה שלנו נובע ש-

$$(Q^tA)_{ij}=\langle q_{*i},a_{*j}\rangle \underset{\text{כאשר } i<j}{=} 0$$

כדי לקבל פירוק למשולשית עליונה פשוט עושים גראם שמידט מסדר הפוך. ע"י פעולה על השורות במקום העמודות ניתן להפוך את סדר המטריצות.
הסיבוכיות של זה היא הסיבוכיות של גראם-שמידט - $O(n^3)$.
! הערה: אנחנו לא מסתכלים פה על אף תכונה של A. זה עובד לכל מטריצה.

דרך 2: Householder Reflection
בונים מטריצות $Q^1,\dots,Q^n$ באופן הבא:

$$V=\frac{a_{*1}-\|a_{*1}\|e_1}{\|\uparrow\|} \quad (\text{מנורמל})$$

$$Q^1=I-2VV^t$$

בשלב הבא מסתכלים על התת מטריצה

$$Q^t A = \begin{pmatrix} \times & \cdots & * \\ 0 & \boxed{\quad} & \\ \vdots & & \\ 0 & & \end{pmatrix}$$

ונבצע עליה את אותה פעולה לקבלת $\tilde{Q}_2$ ואז נגדיר

$$Q^2 = \begin{pmatrix} 1 & 0 \cdots 0 \\ 0 & \\ \vdots & \tilde{Q}^2 \\ 0 & \end{pmatrix}$$

וכך נמשיך... לבסוף את R נקבל ע"י $Q_n \cdots Q_1 A = R$

ו- Q הוא $Q = Q_1^t \cdots Q_n^t$

למה זה עובד. ראשית,

$$(I - 2vv^t)A = \begin{pmatrix} * & \\ 0 & \\ \vdots & * \\ 0 & \end{pmatrix}$$

$$\left(I - 2 \begin{pmatrix} v \end{pmatrix} \begin{pmatrix} v^t \end{pmatrix} \right) \begin{pmatrix} A_{*1} \end{pmatrix} =$$

$$= a_{*1} - 2vv^t a_{*1} =$$

$$= a_{*1} - 2 \underbrace{\frac{(a_{*1} - \|a_{*1}\| e_1) \langle a_{*1} - \|a_{*1}\| e_1, a_{*1} \rangle}{\| a_{*1} - \|a_{*1}\| e_1 \|^2}}_{= \frac{(a_{*1} - \|a_{*1}\| e_1) \cancel{(\|a_{*1}\|^2 - \|a_{*1}\| a_{11})}}{\|a_{*1}\|^2 - 2\|a_{*1}\| a_{11} + \|a_{*1}\|^2} = \frac{}{2 \cancel{(\|a_{*1}\|^2 - \|a_{*1}\| a_{11})}}}$$

$$= a_{*1} - a_{*1} + \|a_{*1}\| e_1 = \begin{pmatrix} * \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

באופן דומה השאר מסתדר.

מה הסבוכיות החישובית? הפעולה העיקרית כאן היא הכפל $Q_n \cdots Q_1 A$

(הכפל QA) והוא ב- $O(n^2)$ כי למעשה vv^t

היא פשוט שורה שחוזרת על עצמה בהפעלות שונות.
⇐ קל לנפול בה.

דרך 3: Givens

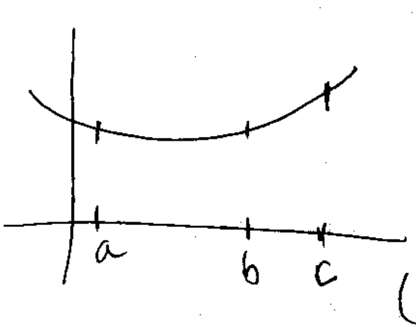
ננסה לפעול בדרך 2 אבל שנשתמש במטריצות Q שראינו בשיטת Jacobi. המטריצות האלה תלויות רק בפרמטר אחד והמטרה היא לאפס שלב-שלב עוד ועוד איברים מתחת לאלכסון. אפשר להשתכנע שאפשר לעשות את זה רק שהמעבר משלב לשלב לא יפגע אותנו: עושים משהו דומה לחילוץ גאוס: קודם מאפסים רק את העמודה הראשונה, אז את השנייה וכן הלאה.
סה"כ השיטה הזו היא גם $O(n^3)$. יש $O(n^2)$ איברים לאפס וכל אחד מהם כרוך ב- $2n$ פעולות.

למה כל זה טוב? זה עוזר לחשב דטרמיננטות של מטריצות
מפולע כי אם $A=QR$ אז
$$|A|=|Q|\cdot|R| = 1\cdot|R| = |R|$$
! - R משולשית אז $|R|$ הוא פשוט מכפלת האלכסון.
אז עכשיו החישוב ממש פשוט.

אופטימיזציה ($(\min f(x))$

Bracketing 1D

אין שימוש בנגזרות!
אבל מניחים שהפונקציה רציפה.

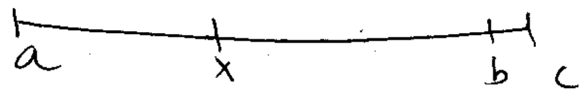


מתחילים עם שלושה $a<b<c$
$f(b) < f(a), f(c)$

בוחרים $x\in[a,b]$.
אם $f(x)<f(b)$ אז $(a,b,c)\to(a,x,b)$
אחרת $(a,b,c)\to(x,b,c)$

בדוגמה הזו אפשר לבחור $x\in[b,c]$ ולהמשיך בצורה.

איפה כדאי לבחור את x? קל להשתכנע שבאופן טבעי עדיף לבחור את x בקטע הרחב יותר:



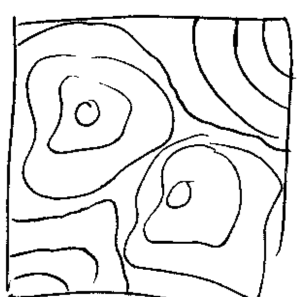
ושוב איפה כדאי לבחור את x? יש חישובים שאפשר לעשות שכדאי לבחור את x רק שהוא מתקדם מ-b לתוך הקטע הגדול ביחס $\phi=\frac{\sqrt{5}-1}{2}$.

דרך אחרת לבחור את x: יש לנו 3 נקודות אז נתפור פרבולה ביניהן. את הפרבולה אפשר לגזור ולמצוא מינימום באופן אנליטי. המינימום הזה יהיה ה-X החדש שלנו.

בפועל בד"כ משלבים בין האסטרטגיות. - Brent Method

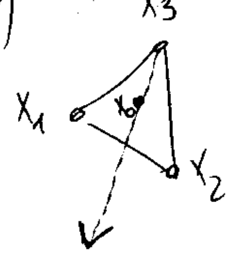
Down hill Simplex (Nedler-Mead)

השיטה הזו מוכללת להרבה ממדים וגם היא לא מניחה שום דבר על הנגזרות.



יש $n+1$ נקודות ובה"כ

$$f(x_1)\le f(x_2)\le\dots\le f(x_{n+1})$$



① Reflection - $x_0=\frac{\sum x_i}{n+1}$ מרכז מסה

$$x^*_{n+1}=x_{n+1}+\alpha(x_0-x_{n+1}),\quad \alpha\ge 1$$

(ואז שוב מסדרים את הנקודות)

② expansion:

if $f(x^*_{n+1})<f(x_1)$ // a miracle!

$$x^*=x_{n+1}+\alpha'(x^0-x_{n+1}),\quad x_{n+1}=x^*_n$$

כלומר אם הלך לנו טוב נבדוק אם אפשר ללכת עוד באותו כיוון)

* יש 6 מיני שלבים שקובעים אילו פעולות לעשות מבין השתיים. ב- Wikipedia יש הדגמה יפה.

אם ① לא הצליח עושים

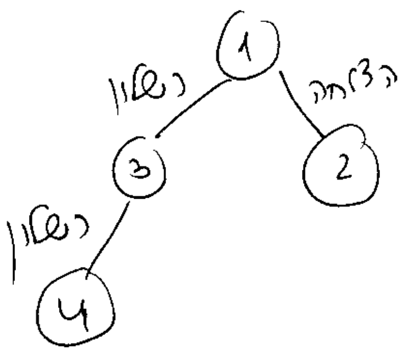
③ Contraction

$$x_{n+1} = x_{n+1} + \rho(x^0 - x_{n+1})$$

$$\rho < 1$$

ואם גם זה לא הצליח:

④ Reduction



כל הנקודות פרט לנקודה הכי טובה מתקדמים אליה:

$$x_i = x_i + \sigma(x_i - x_1)$$

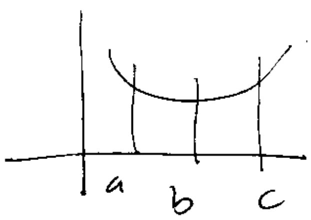
(56) 14|11|10
חישוב
נומרי

הערה לתרגול: בשאלה 4 אנחנו עובדים עם המטריצה

$$\begin{pmatrix} -1 & 1 & & & 0 \\ 1 & -2 & 1 & & \\ & & \ddots & & \\ & & 1 & -2 & 1 \\ 0 & & & 1 & -1 \end{pmatrix}$$

אנחנו משתמשים בשני pre-conditioners ורואים מה קורה.

Bracketing



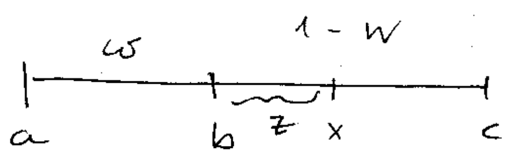
$a<b<c$

$f(b)<f(a), f(c)$

נסמן $w=\frac{b-a}{c-a}$ אז.

$1-w=\frac{c-b}{c-a}$

אנחנו רוצים לבחור נקודה חדשה. הדבר הטבעי הוא לבחור אותה בתוך החלק הגדול יותר. זה ממזער את ה- worst-case.



בה"כ $w<\frac{1}{2}$

שתי האפשרויות הן שהרוחב הבא $1-w$ או $w+z$
אנחנו מניחים שהולך להיות איזה דימיון עצמי בחלוקות אז ההשפעה של z במקרה שעוברים ל- $w+z$ כבר ברורה לנו.
אבל אנחנו לא יודעים מה ההשפעה של z אם נצטרך ללכת ל- $1-w$, אז בדור הבא יהיה לנו $w'=\frac{z}{1-w}$.
כדי להקטין את המקרה הגרוע נדרוש ש- $1-w=w+z$,
כלומר $z=1-2w$ ונדרוש self-similarity, כלומר

$$w=\frac{z}{1-w} \Rightarrow w=\frac{1-2w}{1-w} \Rightarrow w-w^2=1-2w$$

ואת זה אפשר לפתור ולקבל $w_{1,2}=\frac{3\pm\sqrt{5}}{2}$ - יחס הזהב המפורסם.

בשיעור הקודם הזכרנו גם שיש שיטה שמשלבת בין
בחירה לפי יחס הזהב לבין קירוב פולינומיאלי. איך
מתקבל

$$x = b - \frac{1}{2}\frac{(b-a)(f(b)-f(c)) - (b-c)^2(f(b)-f(a))}{(b-a)(f(b)-f(c)) - (b-c)(f(b-f(c))}$$

Line Minimization Methods

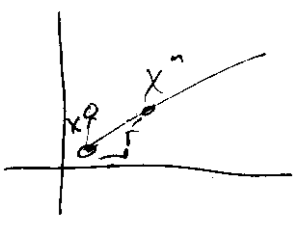
רוצים למזער את $f(x)$ כאשר $x\in\mathbb{R}^n$ כאשר
מניחים שקיים ממזער חד ממדי $\min_{\alpha} f(\bar{x}+\alpha\bar{d})$
(למשל bracketing או שיטה אחרת)

ממזערים חד-ממדיים:

- שיטת החצייה (bracketing)
- Steepest Descent - $x^{n+1} = x^n - f'(x^n)\cdot\varepsilon$
  פשוט הולכים בכיוון הגרדיאנט. ככל ש-ε יותר קטן
  זה מדויק יותר אבל כבד ביותר צעדים.
- שימוש בניוטון לפתרון $f' = 0$

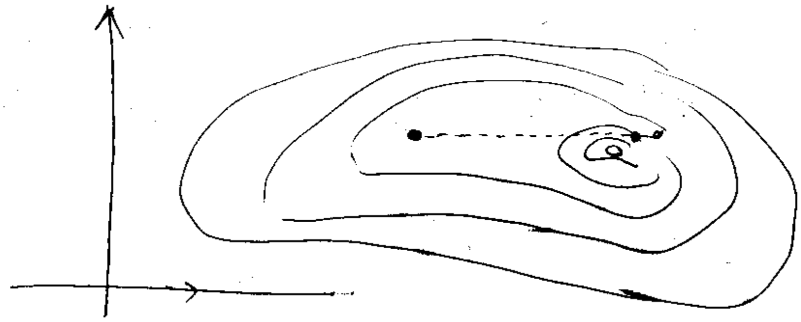
אז השאלה שנשארת צריך לדון מה היא האסטרטגיה
לבחירת כיווני החיפוש:

- שיטת Powell: נבחר בהתחלה $d_i=e_i$ וקטורי
  היחידה עבור $i=1,\dots,n$. אחרי סיבוב כזה ננסה
  להשביח את כיווני החיפוש: נבחר באקראי איזה d_i
  ובמקומו נבחר $d_i = x^n - x^0$
  ועכשיו שוב חוזרים על n צעדים עם
  ה כיוונים. שוב מעדכנים d_i
  אקראי וכך הלאה.



בעיה גדולה בשיטה:

נניח שיש לנו כזאת פונקציה:



מתחילים בנקודה המסומנת. סביר שאחרי הסבב הראשון נתקדם ימינה ואז נבחר d_i אקראי לעדכן אותו ל- e_1. ככה נמשיך ולאט לאט נאבד G יכולת לשלוט על הכיוונים ופשוט נתקדם רק ימינה!

Modified Powell Method -

כמו שיטת Powell אבל שלא נזרוק כיוון d_i אקראי אלא רק d_i ש- $i = \arg\max_i f(x_{i-1}) - f(x_i)$

כלומר זה שתרם הכי הרבה למזעור.

מזעור ב-$\mathbb{R}^n$ עם נגזרות

$$\underbrace{f(x^n+h)}_{x^{n+1}} = f(x^n) + \nabla f(x^n)\cdot h + \frac{1}{2} h^t H_f(x^n)\cdot h + O(\|h\|^3)$$

$$(H_f(x))_{ij} = \frac{\partial^2 f(x^n)}{\partial x_i \partial x_j}$$

למזער את זה, זה כמו למזער את

$$F(h) = h^t b + \frac{1}{2}\cdot h^t A h$$

כאשר $b = \nabla f(x^n)$, $A = H_f(x^n)$

ואת זה אנחנו יודעים לפתור ע"י CG או SD

כי זאת בעיה לינארית.

הערה: אם היינו מסתפקים בנגזרת הראשונה אז יוצא שמאמדים את הפונקציה ע"י פונקציה לינארית ואז ראוי לקרות $h = \pm\infty$ שזה ודאי לא טוב.

אז זה ממש חשוב שהתחשבנו גם בהסיאן.

אם בכל זאת רוצים רק סדר ראשון, אז יש "SD תמימ'" : $x^{n+1} = x^n - \nabla f \cdot \varepsilon$

שיטות אקראיות למזעור

Random Walk - $x^* = x^n + r$

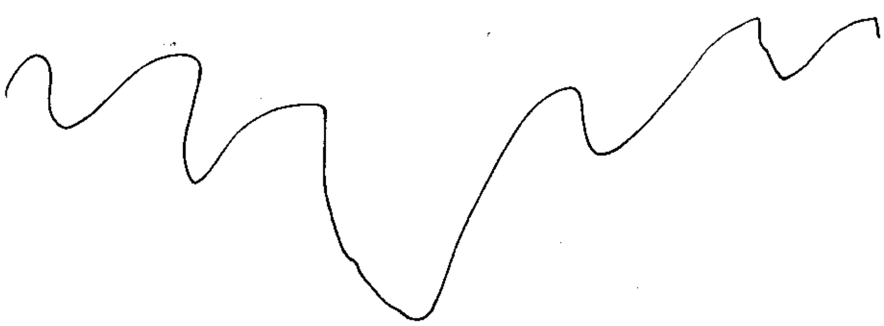
$r \sim N(0, \varepsilon^2)$

בסיכוי $p = e^{-\frac{f(x^{n+1}) + f(x^n)}{T}}$ מגדירים $x^{n+1} = x^*$

אחרת מגדירים $x^{n+1} = x^n$

T נקרא טמפרטורה. אם T נמוך נקבל אמ"מ $f(x^n) > f(x^*)$. אם T גבוה אז תמיד נקבל

Simulated Annealing - מתחילים ב-T גבוה ולאט לאט מקטינים אותו. זה בא להתמודד עם הבעיה



התקווה היא שזה יביא אותנו יותר קרוב למינימום גלובלי.

על זה יש מלא וריאציות. למשל אלגוריתמים גנטיים שעוקבים אחרי מספר רב של נקודות בטמפרטורות שונות ומרבים והורגים אותן לפי הצורך.

58

21/1/2010

חשבון נומרי

משוואות דיפרנציאליות

מהי משוואה דיפרנציאלית?

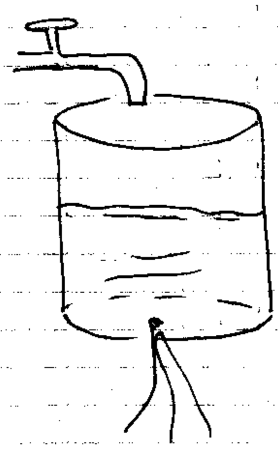
משוואה דיפרנציאלית היא משוואה שמגדירה יחס כלשהו בין פונקציה לבין הנגזרות שלה.

משוואה דיפרנציאלית היא רגילה אם מדובר בפונקציה במשתנה יחיד. סדר המשוואה הוא סדר הנגזרת הכי גבוהה שמופיעה במשוואה.

<u>דוגמה</u> $y=y(t)$ $\quad y^2=t^2+1+y'+y''$

זו משוואה דיפרנציאלית רגילה.

<u>דוגמה</u>: בדלי יש חור קטן ומברז שממנו נשפכים מים לדלי.



$y(t)$ = גובה המים בזמן t.

אפשר למדל את זה עם משוואה דיפרנציאלית:

$$y'(t)=A-B\cdot y(t)$$

כמות המים שנוספת לדלי מהברז ביח' זמן

קבוע קצב הריקון ליח' זמן ביחידות

זמן

זמן / חומר

וכמובן, צריך גם תנאי התחלה $y(t_0)=y_0$

למשוואות דיפ' רגילות יש <u>משפט קיום</u>: נתבונן במשוואה

$$y'(t)=f(t,y(t))$$

אם f רציפה בפרמטר הראשון t ואם f ליפשיצית בפרמטר השני y (כלומר קיים קבוע C כך ש $|f(t,y_1)-f(t,y_2)|\le C|y_1-y_2|$) אז קיימת פונק' יחידה $y(t)$ גזירה ברציפות ומקיימת את המשוואה עבור תנאי התחלה $y(t_0)=y_0$

אבל מה עם משוואות מסדרים גבוהים יותר ?
משוואה מסדר גבוה יותר ניתנת לרדוקציה למשוואות מסדר ראשון. נניח שיש לנו

$$y'' = y' + ay - 1$$

נגדיר שתי פונקציות

$$y_1 = y$$
$$y_2 = y' = (y_1')$$

ולכן יש לנו למעשה שתי משוואות

$$y_2' = y_2 + ay_1 - 1$$
$$y_1' = y_2$$

ועכשיו יש לנו שתי משוואות בשני נעלמים. כמובן, אפשר היה להכליל את זה ל-n משוואות ב-n נעלמים.
משפט הקיום למעשה נכון למע' משוואות מסדר ראשון ובזאת סיימנו את הרדוקציה.

יש משוואות מאוד פשוטות שאפשר לנחש פיתרונות שלהם ויש משוואות מסוימות מסויימים שיש להן שיטות לפתרון. אבל החיים לא כאלה יפים וסביר שלא נדע לפתור על הנייר. חוץ מזה יש משוואות שאפשר להוכיח שאנחנו לא יכולים לכתוב להן נוסחה סגורה.
אז כמובן נצטרך לפנות למחשב לעזרה.

אז יש לנו משוואה $y'(t) = f(y,t)$
נניח שאנחנו עובדים בדיסקרטיזציה h של הזמן.
אז יש לנו וקטור זמן $t_j = j \cdot h$
נסמן $y_j = y(jh)$ (הערכים האמיתיים)
ונסמן את הקירוב שלנו ב- $w_j \approx y_j$
אז מקירוב טיילור אנחנו נשתמש במשוואת הפרשים

$$\frac{w_{j+1} - w_j}{h} = f(w_j, t_j)$$

זה נקרא שיטת טיילור קדמית (forward)

(59) והיא מפורשת כי w_{j+1} תלוי בצורה מפורשת ב-w_j

שיטת אוילר האחורית נגדיר באופן סתום

$$\frac{w_{j+1}-w_j}{h} = f(w_{j+1}, t_j)$$

אם f פשוטה אולי פשוט נוכל להעביר אגפים אבל יכול להיות ש-f מסובכת.

עכשיו נדבר על single step methods - כאשר משוואות ההפרשים כוללת רק w_j, w_{j+1} (ולא w_{j-1} או w_{j+2} וכו'). תיאורטית זה מגביל אותנו אבל בפרקטיקה השיטות האלה מספיקות ונותנות תוצאות מאוד טובות.

<u>הגדרות:</u>

- שגיאת הקיטוע (truncation error)

  מודד כמה הפתרון האמיתי מקיים את משוואות ההפרשים

$$\tau_j = \frac{y_{j+1}-y_j}{h} - f(y_j, t_j)$$

- שגיאת קיטוע מקומית (local truncation error)

$$r_j = \tau_j \cdot h = y_{j+1} - y_j - hf(y_j, t_j)$$

- משוואות ההפרשים (scheme) הינן עקביות (consistent) אם $\tau_j \xrightarrow{h\to 0} 0$

מתקיים $y_{j+1} = y_j + h y_j' + \frac{h^2}{2} y_j'' + O(h^3)$ (קירוב טיילור)

$$\Rightarrow \tau_j = \frac{y_{j+1}-y_j}{h} - f(y_j, t_j) =$$

$$= \cancel{y_j'} + \frac{h}{2} y_j'' + O(h^2) - \cancel{f(y_j, t_j)} =$$

y_j' מצטמצם עם $f(y_j, t_j)$ כי y_j מסמל את הפתרון האמיתי

$$\frac{h}{2} y_j'' + O(h^2) \le \sup_{t\in[t_j, t_{j+1}]} \frac{y''}{2} h + O(h^2) \xrightarrow{h\to 0} 0$$

- שגיאה גלובלית $e_j = y_j - w_j$

שיטה היא מסדר s אם $e_j = O(h^s)$ עבור זמן t מסוים.

(ה-j שעבורו jh זה הזמן שאנחנו מדברים עליו)

כך אפשר לומר ש- סדר השיטה = סדר של τ

- שיטה מתכנסת אם $e_j \xrightarrow[h\to 0]{} 0$ עבור זמן קבוע.

הערכת שגיאה גלובלית

לפי ההגדרה של r_j:

$$y_{j+1} = y_j + hf(y_j, t_j) + r_j$$

לפי הגדרת הסכמה

$$w_{j+1} = w_j + hf(w_j, t_j)$$

נחסר ונקבל

$$e_{j+1} = y_{j+1} - w_{j+1} = e_j + hf(y_j,t_j) - hf(w_j,t_j) + r_j$$

$$\Rightarrow |e_{j+1}| \le |e_j| + h\,|f(y_j,t_j) - f(w_j,t_j)| + |r_j| \le$$

ליפשיץ

$$\le |e_j| + h\cdot L\cdot |y_j - w_j| + |r_j| =$$

$$= (1+hL)|e_j| + |r_j|$$

$$\Rightarrow |e_{j+1}| \le \sum_{i=0}^{j} |r_i|(1+hL)^{j-i} \le$$

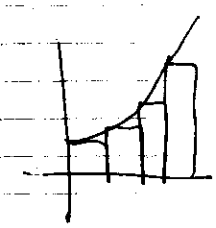
$$\le \sum_{i=0}^{j} |r_i|\, e^{hL(j-i)} = h\sum_{i=0}^{j} |\tau_j|\, e^{hL(j-i)} \le$$

$1+x \le e^x$

$$\le h\tau \sum_{i=0}^{j} e^{hL(j-i)} \le \tau \int_0^{t_j} e^{Lt}\,dt =$$

$\tau = \max_j \tau_j$

כי זה קירוב
רימן וזו פונק' מונוטונית
עולה ואם בוחרים
נכון את הנקודות זה יוצא חסם מלמעלה



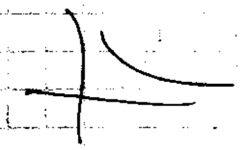
$$= \frac{\tau}{L} e^{t_j \cdot L} \left(\xrightarrow[h\to 0]{} 0 \quad \text{כי } \tau \to 0 \right)$$

- השגיאה עולה אקספוננציאלית אבל ככל שלוקחים h קטן יותר אפשר לנסות להתגבר על זה.
בכל אופן קירובים למשוואה בזמנים רחוקים הם פחות מוצלחים ולכן לא יודעים את התחזית לעוד יומיים).

דוגמה: $y' = -ay \quad 0<a$

• אוילר קדמי: $w_{j+1} = w_j - ahw_j = (1-ah)w_j$

אם $ah>2$ אז נקבל ש- $|w_j| \to \infty$ וזה איבר מחליף סימן. זה בניגוד שאנחנו יודעים שהפתרון $y=e^{-at}$ דועך רק



אז מקבלים דרוש ש- $|1-ah|<\frac{1}{2}$, כלומר $h<\frac{1}{2a}$

• אוילר אחורי: $w_{j+1} = w_j - ahw_{j+1}$

$$\Rightarrow \quad w_{j+1}(1+ah) = w_j$$

וכאן מקדם הניפוח נעשה בצד הטוב, מה שלא יהיה אנחנו שואפים לאפס.

נראה דוגמה למקרה שבו אוילר אחורי יותר טוב מהקדמי. אבל האמת היא שזה גם נכון באופן כללי.

Mid step (מקרה פרטי של Runge Kutta)

$$y_{j+1} = y_{j+\frac12} + \frac{h}{2}\cdot y'_{j+\frac12} + \frac{h^2}{8}y''_{j+\frac12} + O(h^3)$$

$$y_j = y_{j+\frac12} - \frac{h}{2}y'_{j+\frac12} + \frac{h^2}{8}y''_{j+\frac12} + O(h^3)$$

$$\Rightarrow \frac{y_{j+1}-y_j}{h} = y'_{j+\frac12} + O(h^2)$$

אז נשתמש במשוואת הפרשים

$$\frac{w_{j+1}-w_j}{h} = f(w_{j+\frac12}, t_{j+\frac12})$$

כאשר מגדירים $w_{j+\frac12} = w_j + \frac{h}{2}f(w_j,t_j)$

אז שגיאת הקיטוע היא

$$\tau_j = \underbrace{\frac{y_{j+1}-y_j}{h}}_{=\,y'_{j+\frac12}+O(h^2)} - f\left(y_j + \frac{h}{2}f(y_j,t_j),\ t_{j+\frac12}\right)$$

אנחנו רוצים שגם במחובר השני תהיה שגיאה
של $O(h^2)$

$$f\left(y_j + \tfrac{h}{2} f(y_j, t_j),\ t_{j+\frac{1}{2}}\right) =$$

טיילור $$= f(y_{j+\frac{1}{2}}, t_{j+\frac{1}{2}}) + \underbrace{\left(y_j + \tfrac{h}{2} f(y_j, t_j) - y_{j+\frac{1}{2}}\right)}_{h^2 \tau_j} \cdot f'(y_{j+\frac{1}{2}}, t_{j+\frac{1}{2}}) + O(-)$$

עכשיו בסכום של τ_j $y'_{j+\frac{1}{2}}$ מצטמצם עם
$f(y_{j+\frac{1}{2}}, t_{j+\frac{1}{2}})$ ונשארים עם משהו מסדר $O(h^2)$