פרוייקט גמר בקורס אנאליזה נומרית: אינטרפולאציית Neville

Neville's Method

אדם אהרוני מאור ולדמן

מכון טכנולוגי חולון HIT

יום חמישי, 01. מרץ, 2021.

פרולוג:

במסגרת פרוייקט זה, נסביר, נפרט, ונראה בפעולה על שיטת האינטרפולאצייה לפי Neville.
נוכיח גם איך היא עובדת, נראה דוגמאות, וכן נשווה מול שיטות אינטרפולאצייה אחרות שלמדנו בקורס.

במסגרת הפרוייקט הרחבנו את אופקינו בתחום האנאליזה הנומרית והאינטרפולאצייה.
הפרוייקט גרם לנו לחקור את שיטת Neville עד תום, ובכך נתן לנו מקור להשוואה עם שיטות אינטרפולאצייה אחרות.

פרולוג:

בהינתן $1 + n$ נקודות דאטה, קיים פולינום ייחודי מסדר $\leq n$ העובר דרך הנקודות הנ.ל..
באמצעות שיטת Neville ניתן להגיע לערך אותו פולינום בנקודה כלשהי.

- השיטה מבוססת על שיטת האינטרפולאצייה של ניוטון (Newton) ועל מציאת הפרשים מחולקים.

פולינום האינטרפולאצייה:

בהינתן $n+1$ נקודות דאטה $(x_k; y_k)$, $k=0;1;\ldots;n$, כאשר שיעורי ה-x שונים, פולינום האינטרפולאצייה הוא פולינום מסדר של לפחות n המקיים את:

$$\forall k = 0;1;\ldots;n:$$
$$p(x_k) = y_k$$

הפולינום הזה קיים והוא ייחודי. שיטת Neville מוצאת את ערך פולינום האינטרפולאצייה הנ.ל. בנקודה x כלשהי.

נגדיר פולינום $p_{i;j}(x)$ (כאשר מתקיים $i \leq j$) בתור פולינום מסדר של $j - i$ העובר דרך נקודות הדאטה הנתונות $(x_k; y_k)$ עבור $k = i; i+1; \ldots; j$.

תהליך מציאת הפולינום:

על הפולינומים לקיים את היחס הרקורסיבי:

$$\begin{cases} p_{i;\,i}(x) &= y_i \\ p_{i;\,j}(x) &= \dfrac{(x-x_j)p_{i;\,j-1}(x)-(x-x_i)p_{i+1;\,j}(x)}{x_i-x_j} \end{cases} ; \quad \begin{matrix} 0 \le i \le n \\ 0 \le i < j \le n \end{matrix}$$

נוסחת הנסיגה הזו יכולה לחשב את $p_{0;\,n}(x)$, שהוא הערך אותו אנו מחפשים.

הוכחת הנוסחה הרקורסיבית:

הביטוי $p_{i;\,i}(x) = y_i$ הוא מיידי, ואת הנוסחה הרקורסיבית נוכיח באמצעות אינדוקציה:

בהינתן $i;\ j$ עם פולינום $p_{i;\,j-1}(x)$ העובר דרך הנקודות $(x_k;\ y_k)$ עם $k = i;\ i+1; \ldots ;\ j-1$,
וכן הפולינום $p_{i+1;\,j}(x)$ העובר דרך הנקודות $(x_k;\ y_k)$ עם $k = i+1;\ i+2; \ldots ;\ j$.
לפי ההנחה הנ.ל. מתקיים:

$$\begin{cases} p_{i;\,j-1}(x_k) = y_k & i \le k \le j-1 \\ p_{i+1;\,j}(x_k) = y_k & i-1 \le k \le j \end{cases}$$

ולכן, עבור $i+1 \le k \le j-1$, מתקבל:

$$p_{i;j}(x_k) = \frac{(x_k - x_j)\,p_{i;j-1}(x_k) - (x_k - x_i)\,p_{i+1;j}(x_k)}{x_i - x_j} =$$
$$= \frac{(x_k - x_j)\,y_k - (x_k - x_i)\,y_k}{x_i - x_j} = y_k$$

וכן:

$$\begin{cases} p_{i;j}(x_i) & = \frac{(x_i - x_j)\,p_{i;j-1}(x_i)}{x_i - x_j} = y_i \\ p_{i;j}(x_j) & = \frac{-(x_j - x_i)\,p_{i+1;j}(x_j)}{x_i - x_j} = y_j \end{cases}$$

ולכן $p_{i;j}(x)$ עובר דרך כל הנקודות $(x_k;\ y_k)$ כאשר $k = i;\ i+1; \ldots ; j$.

הוכח.

דוגמה:

ניקח את $n = 4$. נוכל להשתמש בנוסחת הרקורסייה על מנת למלא את התרשים משמאל לימין באופן הבא:

$$
\begin{array}{ccccccccc}
p_{0;\,0}(x) = y_0 & & & & & & & & \\
& p_{0;\,1}(x) & & & & & & & \\
p_{1;\,1}(x) = y_1 & & p_{0;\,2}(x) & & & & & & \\
& p_{1;\,2}(x) & & p_{0;\,3}(x) & & & & & \\
p_{2;\,2}(x) = y_2 & & p_{1;\,3}(x) & & \boxed{p_{0;\,4}(x)} & & & & \\
& p_{2;\,3}(x) & & p_{1;\,4}(x) & & & & & \\
p_{3;\,3}(x) = y_3 & & p_{2;\,4}(x) & & & & & & \\
& p_{3;\,4}(x) & & & & & & & \\
p_{4;\,4}(x) = y_4 & & & & & & & &
\end{array}
$$

דוגמה

התהליך מביא לנו את $p_{0;4}(x)$, שהוא בעצם ערך הפולינום העובר דרך כל $n+1$ נקודות הדאטה (מתנאי ההתחלה) בנקודה x כלשהי.

(ניתן גם לומר שאלגוריתם זה רץ בסיבוכיות של $O(n^2)$, הוכחה בהמשך)

זמן הריצה של האלגוריתם:

בהתחשב בכך שכדי למצוא את פולינום $p_{0;n}(x)$ נצטרך להשתמש ברקורסייה שבה מחשבים לראשונה n פולינומים, לאחר מכן $n-1$ פולינומים וכן הלאה...
מכאן, גודל הפולינומים שמחשבים מובא על ידיי:

$$n + (n-1) + (n-2) + \cdots + 1 =$$

$$= \sum_{i=1}^{n} i$$

מדובר בסכום סדרה חשבונית סטאנדארטית. לא קשה לראות שהסכום מקיים:

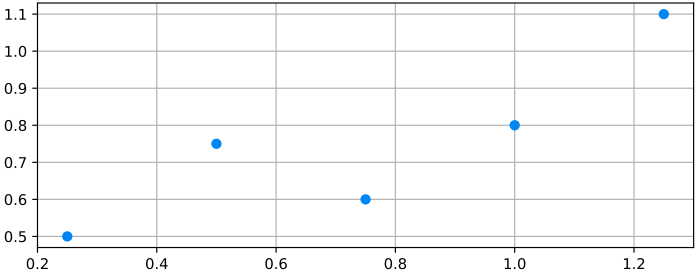
$$S=\sum_{i=1}^{n} i=\frac{n\left(n+1\right)}{2}=\frac{n^2}{2}+\frac{n}{2}=$$
$$=\boxed{O\left(n^2\right)}$$

ומכאן, הגענו לזמן ריצה של $O\left(n^2\right)$.

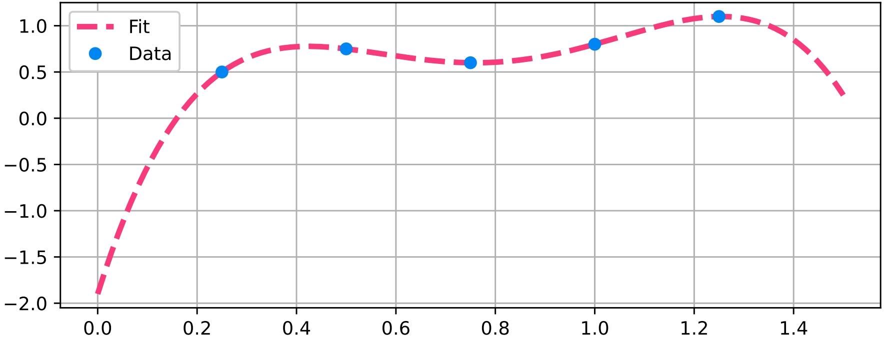
הוכח.

דוגמה N°1:

הפונקצייה לפני אינטרפולאצייה:

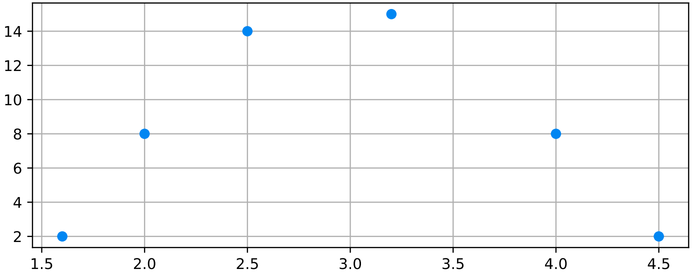


הפונקצייה אחרי אינטרפולאצייה:

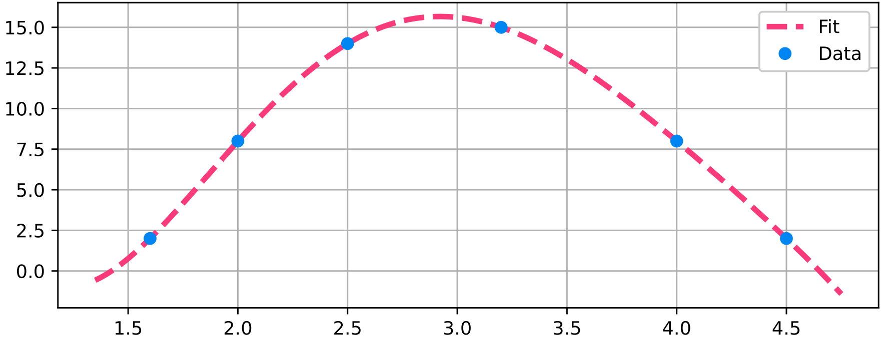


דוגמה N°2:

הפונקצייה לפני אינטרפולאצייה:

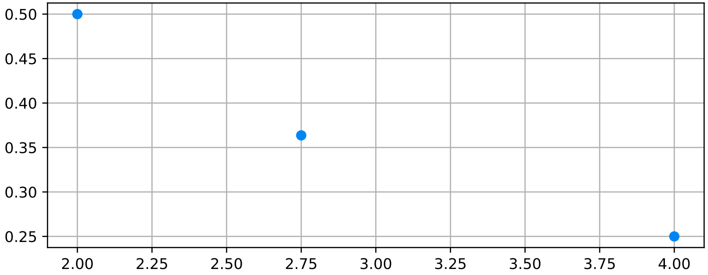


הפונקצייה אחרי אינטרפולאצייה:

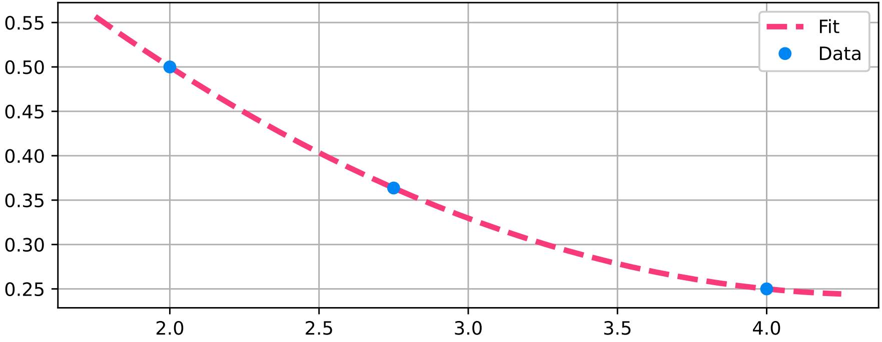


דוגמה N°3:

הפונקצייה לפני אינטרפולאצייה:

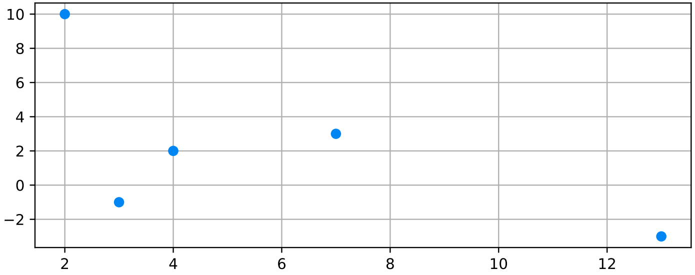


הפונקצייה אחרי אינטרפולאצייה:

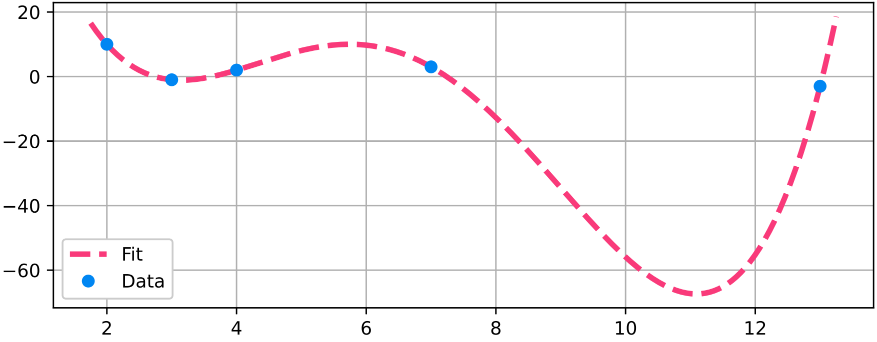


דוגמה N°4:

הפונקצייה לפני אינטרפולאצייה:

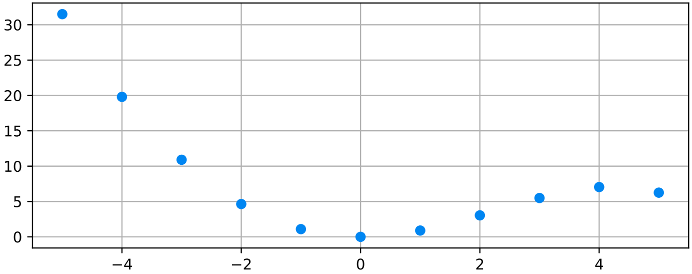


הפונקצייה אחרי אינטרפולאצייה:

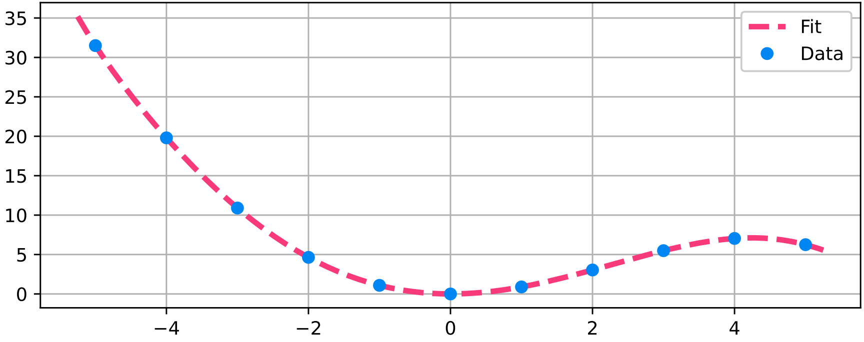


דוגמה N°5:

הפונקצייה לפני אינטרפולאצייה:

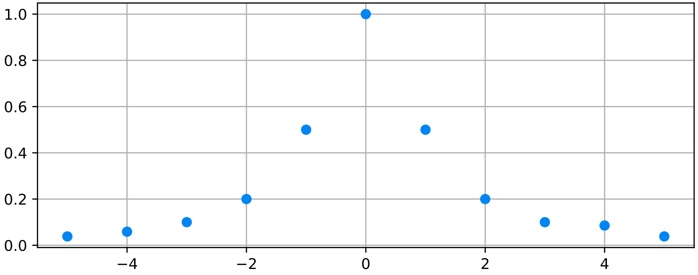


הפונקצייה אחרי אינטרפולאצייה:

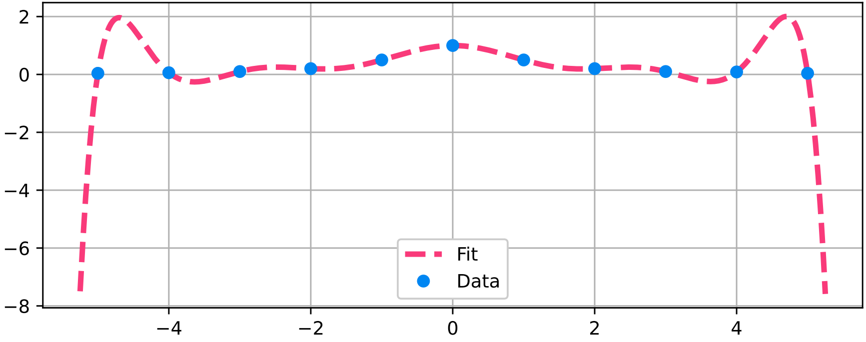


דוגמה N°6:

הפונקצייה לפני אינטרפולאצייה:



הפונקצייה אחרי אינטרפולאצייה:



השוואה עם טכניקות אחרות:

כעת, נערוך השוואה עם טכניקות אינטרפולאצייה אחרות כמו:

- אינטרפולאצייה לפי לאגראנז׳ (Lagrange)
- אינטרפולאצייה לפי `numpy.polyfit`