1. téma
Közelítő (numerikus) számítások, hibaforrások, hibabecslések

Bevezetés

Mindenekelőtt azt szeretnénk tisztázni, hogy miért van szükség olyan algoritmusok kidolgozására, amelyek egy adott matematikai probléma megoldását csak *közelítik*. Miért nem olyan eljárásokkal foglalkozunk, amelyekkel *pontosan* meg lehet oldani a problémákat? A válasz sok esetben természetes: mert bizonyos feladatok esetén ilyen eljárások bizonyítottan *nincsenek*. Tekintsük például a következő feladatokat!

1. Határozzuk meg egy adott *algebrai egyenlet gyökeit*! Algebrából tudjuk, hogy ha az egyenlet fokszáma 4-nél nagyobb, akkor a Ruffini-Abel tétel szerint nincs olyan gyökképlet, amellyel az összes gyököt egy *tetszőleges* egyenlet együtthatóiból *véges* lépésben *pontosan* meg tudnánk határozni. Gyakorlati szempontból nézve azonban a problémát, sok esetben erre nincs is szükség. Egy mérnök számára elegendő lehet, ha egy egyenlet gyökeit például négy tizedes jegy pontosságig ismeri. Az ő számára a gyökök egzakt megadása érdektelen. Természetesen merül fel a kérdés: Tudunk-e olyan algoritmust kidolgozni, amellyel az ő igényének megfelelhetünk?

2. Analízisből tanultuk *függvények határozott integráljának kiszámítását*. A Newton-Leibniz szabályt használva határozott integrálokat ki tudtunk számolni, miután a felhasználandó primitív függvényt felírtuk. Bizonyos esetekben azonban a primitív függvényt elemi függvényekkel nem tudjuk megadni (mert bizonyítottan *nem lehet*) vagy a felírása *nagyon komplikált*. Az alkalmazások számára azonban ekkor is sok esetben érdektelen, hogy *pontosan* mi az integrál értéke, elég annak egy *jó közelítése*. Hogyan adhatunk meg ilyeneket?

3. Lineáris algebrából mindenki megtanulta, hogyan lehet egy *lineáris egyenletrendszert megoldani*. Tanultuk a Cramer-szabályt, amely a gyököket determinánsok hányadosaként állítja elő. A gyakorlati életben, tapasztalt szakembernek azonban nem jutna eszébe, hogy egy általános nagyméretű lineáris egyenletrendszert a Cramer-szabállyal oldjon meg, a klasszikus determináns számolási szabállyal. Lineáris egyenletrendszerek megoldására léteznek az alkalmazások számára a Cramer-szabálynál hatékonyabban használható eliminációs és iterációs módszerek.

Hibaszámítás, hibabecslés

Ha egy eljárás egy feladatot nem pontosan old meg, hanem az eredménynek csak egy becslését, közelítését adja, akkor matematikai szempontból nézve az

eredmény hibás. Egy értékre adott becslés persze számunkra lehet nagyon jó, de lehet gyenge is. Természetesen merül fel, hogy az érték becslésén túl szükségünk van olyan mutatószámra is, amellyel valamiképpen jellemezhetjük a becslés jóságát.

Definíció: *Legyen az x valós számnak x' egy közelítése. A közelítés abszolút hibája az |x-x'| érték. Ha egy közelítés abszolút hibája nem nagyobb, mint egy nem negatív ε, akkor ε-t abszolút hibakorlátnak mondjuk.*

Példák különböző műveletek elvégzésekor adódó abszolút hibák becslésére:

Példa: Legyen $|x_i - x'_i| \le \varepsilon_i$ *(i=1,2)*. Milyen abszolút hibakorlátot tudunk adni az x_1 és x_2 összeadásakor fellépő abszolút hibára? Feladatunk tehát megadni egy olyan értéket, amelynek felírásában nem szerepelnek az (ismeretlen) x_1 és x_2 pontos értékek és az $|x_1 + x_2 - x_1' - x_2'|$ értéknél nem kisebb. A megoldás egyszerű:

$$|x_1 + x_2 - x_1' - x_2'| = \left|x_1 - x_1' + x_2 - x_2'\right| \le \left|x_1 - x_1'\right| + \left|x_2 - x_2'\right| \le \varepsilon_1 + \varepsilon_2$$

Kicsit nehezebb a dolgunk, ha ugyanezt a kérdést a szorzásra tesszük fel:

Példa: Legyen $|x_i - x'_i| \le \varepsilon_i$ *(i=1,2)*. Igazoljuk, hogy

$$\left|x_1 x_2 - x_1' x_2'\right| \le \varepsilon_1 \varepsilon_2 + \left|x_1'\right|\varepsilon_2 + \left|x_2'\right|\varepsilon_1 .$$

A megoldás során hamar kiderül, hogy a becsülendő különbséget alkalmas tagok hozzáadásával és levonásával kell ellátni. A kérdés csupán az, hogy melyek legyenek ezek a tagok, ha azt szeretnénk, hogy az abszolút hibakorlátban ne szerepeljenek az x_1 és x_2 ismeretlen értékek. Lehet például így csinálni:

$$\left|x_1 x_2 - x_1' x_2'\right| \le \left|(x_1 - x_1')(x_2 - x_2')\right| + \left|x_1'\right|\left|x_2 - x_2'\right| + \left|x_2'\right|\left|x_1 - x_1'\right| \le$$

$$\le \varepsilon_1 \varepsilon_2 + \left|x_1'\right|\varepsilon_2 + \left|x_2'\right|\varepsilon_1$$

A következő példa megoldása előtt elevenítsük fel a Lagrange-féle középérték tételt. *Lagrange-féle középérték tétel*: *Ha az f valós függvény az [a,b] intervallumon folytonos és az intervallum belsejében mindenütt deriválható, akkor az a és b között van legalább egy olyan c hely, hogy*

$$f(b) - f(a) = f'(c)(b - a) .$$

A következő példát meg lehetne oldani pusztán középiskolai ismeretekre támaszkodva. A Lagrange-féle középérték tétel segítségével azonban a megoldás csak egy sor.

Példa: Igazoljuk, hogy ha $\left|x-x^{'}\right| \le \varepsilon$, akkor $\left|\sin(x)-\sin(x^{'})\right| \le \varepsilon$.

Megoldás: A Lagrange-féle középérték tételt felhasználva:

$$\left|\sin(x)-\sin(x^{'})\right| = |\cos(\xi)|\,|x-x'| \le 1 \cdot \varepsilon = \varepsilon.$$

Az abszolút hiba mellett egy másik fontos fogalom a becslés relatív hibája.

Definíció: *Ha az x értéket x'-vel közelítjük, akkor a becslés relatív hibáján az* $\frac{|x-x'|}{|x|}$ *értéket értjük. Ha egy közelítés relatív hibája nem nagyobb, mint egy nem negatív ε, akkor ε-t relatív hibakorlátnak mondjuk.*

Házi feladat: Adjunk relatív hibakorlátot a szorzás műveletekor elkövetett relatív hibára.

Hibaforrások

Tekintsük át azokat a legfontosabb forrásokat, ahonnan a feladatmegoldás során hibák származhatnak.

Alapvetően két fő hibaforrás osztályt különböztetünk meg: vannak *öröklött hibák* és vannak *számítási hibák*.

Öröklött hibák:
a) a gyakorlati problémák matematikai modelljei legtöbbször csak *közelítései* a valóságnak, így a modellünk már eleve nem pontos,
b) a kiinduló (pl. mérésből származó) adatok pontatlansága, az *adathiba*.

Számítási hibák:
a) gépi vagy emberi *tévedés* (durva hiba),
b) *képlethiba* (erről akkor beszélünk, amikor nem a tényleges matematikai modellel megfogalmazott feladatot, hanem annak csak egy közelítését oldjuk meg),
c) *kerekítési hiba* (elég csak arra gondolni, hogy számítógéppel való munkák során a lebegőpontos számokkal való számolás rögzített számú tizedes jeggyel dolgozik).

Hibás eredményre nem feltétlenül csak hosszú és „bonyolult” számítások során juthatunk. Számos példát lehetne említeni olyan esetre, amikor viszonylag egyszerű feladatok során is hamis eredményt kaphatunk számítógéppel végzett számoláskor a hagyományos lebegőpontos aritmetikát használva.

Részlet E.Hansen és G.W. Walster: *Global Optimization Using Interval Analysis* (2004) című könyvéből. Rump példája.

ily valid. Using IEEE-754 computers, the following form (from Loh and Walster (2002)) of Rump's expression with $x_0 = 77617$ and $y_0 = 33096$ replicates his original IBM S/370 results.

$$f(x, y) = (333.75 - x^2)y^6 + x^2(11x^2y^2 - 121y^4 - 2) + 5.5y^8 + \frac{x}{2y} \qquad (1.4.1)$$

With round-to-nearest (the usual default) IEEE-754 arithmetic, the expression in (1.4.1) produces:

32-bit: $f(x_0, y_0) = 1.172604$
64-bit: $f(x_0, y_0) = 1.1726039400531786$
128-bit: $f(x_0, y_0) = 1.172603940053178631858834904520 1838$

All three results agree in the first seven decimal digits and thirteen digits agree in the last two results. Nevertheless, they are all completely incorrect. Even their sign is wrong.

Loh and Walster (2001) show that both Rump's original and the expression for $f(x, y)$ in (1.4.1) reduce to:

$$f(x_0, y_0) = \frac{x_0}{2y_0} - 2, \qquad (1.4.2)$$

from which

$$f(x_0, y_0) = -0.827396059946821368141165095479816... \qquad (1.4.3)$$

with the above values for x_0 and y_0.

A számítástudománynak van egy ága, amit angolul *Reliable Computing*, magyarul *Megbízható számítások* néven aposztrofálnak és ennek éppen az a célja, hogy olyan módszereket dolgozzon ki, amelyekkel garantált megbízhatóságú számolásokat végezhetünk számítógéppel. Egy ilyen technika a több mint félévszázada tanulmányozott és használt *intervallum aritmetika*.

Intervallum aritmetika

A módszer alapötlete az, hogy egy valós szám helyett, olyan korlátos és zárt valós intervallummal dolgozunk, amelyről biztosan tudjuk, hogy abban az adott szám benne van. A hagyományos valós aritmetikát így felváltja az intervallumokkal számoló *intervallum aritmetika*, a valós analízist az *intervallum analízis*. A hagyományos matematika mellett szokás intervallum matematikáról is beszélni. A megbízhatóság kulcsfontosságú része, hogy egy ún. *kifelé kerekítés*nek nevezett eljárással garantálják, hogy a számítógéppel végzett műveletek során az eredményintervallumok inkább egy kicsit szélesebbek legyenek, de a végeredmény garantáltan az adott intervallumban legyen benne.
Az alapműveletek definiálása az intervallum aritmetikában:

$$[x_1, y_1]+[x_2, y_2]=[x_1+x_2, y_1+y_2]$$
$$[x_1, y_1]-[x_2, y_2]=[x_1-y_2, y_1-x_2]$$
$$[x_1, y_1]\times[x_2, y_2]=[\min(x_1x_2, x_1y_2, y_1x_2, y_1y_2), \max(x_1x_2, x_1y_2, y_1x_2, y_1y_2)]$$
$$1/[\mathrm{x}, \mathrm{y}]=[1/\mathrm{y}, 1/\mathrm{x}], \text{ ha } 0\notin[\mathrm{x}, \mathrm{y}]$$
$$[x_1, y_1]/[x_2, y_2]=[x_1, y_1]\times 1/[x_2, y_2]$$

Más algebrai szabályok érvényesülnek ebben az aritmetikában, mint a valós számoknál megszokott műveletek esetén. Például itt az összeadás és a kivonás, valamint a szorzás és az osztás nem inverz műveletei egymásnak.

Nem igaz, hogy a szorzás disztributív az összeadásra nézve, csak egy ún. *szubdisztribúciós szabály* teljesül: $A(B+C)\subseteq AB+AC$, ahol A,B,C valós intervallumok. Lássunk néhány további furcsaságot!

Példa: Határozzuk meg az $f(x)=x(1-x)$ függvénynek az értékkészletét a [0,1] intervallumon! A megoldás triviális: f([0,1])=[0,1/4]. Számoljunk azonban az intervallum aritmetika műveleteivel!

a) f([0,1])=[0,1]([1,1]-[0,1])=[0,1][0,1]=[0,1]. Így nem jön ki a pontos eredmény! Próbáljunk meg most átalakítani egy kicsit a függvényt, szorozzunk be *x*-szel! Tehát tekintsük az $f(x)=x-x^2$ ekvivalens alakot.

b) f([0,1])=[0,1]-$[0,1]^2$=[0,1]-[0,1]=[-1,1]. Most sem jött ki! Ráadásul más eredmény adódott, mint az első számoláskor.
c) Próbáljuk most előbb teljes négyzetté kiegészíteni a függvényt, tehát nézzük az $f(x)=-(x-1/2)^2+1/4$ alakot!

$$f([0,1])=-([0,1]-[1/2,1/2])^2+[1/4,1/4]=-[-1/2,1/2]^2+[1/4,1/4]=$$
$$=[-1/4,0]+[1/4,1/4]=[0,1/4]$$

Végre megvan! Igaz közben rájöttünk arra, hogy az intervallumos számolásnál $x\times x$ nem egyenlő x^2-tel. Vegyük észre, hogy mindhárom esetben a kapott intervallum tartalmazza a pontos eredményt!

2. téma
Eliminációs módszerek, trianguláris felbontások

Lineáris egyenletrendszer mátrixos alakban való felírása: *Ax=b*, ahol *A* valós mátrix, *b* valós vektor, *x* az ismeretleneket tartalmazó vektor. A továbbiakban mi csak a kvadratikus mátrixú lineáris egyenletrendszerekkel foglalkozunk.

Eliminációs módszerek, Gauss-elimináció

Az eliminációs módszerek lényege az, hogy az adott lineáris egyenletrendszer helyett egy *vele ekvivalens*, de már *könnyen megoldható* lineáris egyenletrendszert határozunk meg. Ez utóbbi egyenletrendszert egy eliminációs eljárás végrehajtása után kapjuk meg.

A Gauss-elimináció során a lineáris egyenletrendszer mátrixát egy *felső trianguláris* mátrixszá alakítjuk úgy, hogy az elimináció minden egyes lépésében a mátrix főátlójának rendre következő eleme alatt minden elemet kinullázunk. Tehát az

$$
\begin{aligned}
&a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = b_1\\
&a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n}x_n = b_2\\
&a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \ldots + a_{3n}x_n = b_3\\
&\ldots\\
&a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ldots + a_{nn}x_n = b_n
\end{aligned}
$$

kiinduló egyenletrendszerből az elimináció az alábbiak szerint jár el: Ha a_{11} nem 0, akkor képezzük rendre az a_{j1}/a_{11} hányadost, majd rendre megszorozva ezzel a hányadossal az első egyenletet a szorzatot levonjuk a *j*-dik egyenletből (*j=2,…,n)*. Ennek eredményeként az első oszlopban a második elemtől kezdve csupa 0 lesz. A következő *i*. lépésben, ha a kapott egyenletrendszer főátlójának *i*. eleme nem 0, akkor vele elosztjuk a *j*. sor *i*. elemét, majd ezzel a hányadossal az előbbi módon rendre kinullázzuk az *i*. oszlopban a főátló alatti értékeket (*i=2,...,n-1* és *j=i+1,..,n*).

A fenti eljárást végrehajtva végül egy olyan lineáris egyenletrendszert kapunk amelynek mátrixa *felső trianguláris* (vagyis a főátló alatt minden eleme 0). Az ilyen lineáris egyenletrendszereket a *visszahelyettesítés* módszerével már könnyű megoldani. Az utolsó egyenlet már csak egy egyváltozós lineáris egyenlet, amiből x_n könnyen adódik. Az x_n-t visszahelyettesítve az eggyel felette álló egyenletbe, szintén könnyen adódik az x_{n-1} értéke. Majd az x_n és x_{n-1} ismeretében az *n-2*. egyenletből x_{n-2} adódik, és így tovább.

A Gauss-elimináció műveletigénye: $\frac{2}{3}n^3 + O(n^2)$.

Jordan-elimináció

A Jordan-elimináció során a legutoljára kapott lineáris egyenletrendszer mátrixa az *egységmátrix*, így a megoldás rögtön leolvasható. Ez az eliminációs eljárás hasonló a Gauss-eliminációhoz, itt is minden lépésben elemeket nullázunk ki, de nemcsak a főátló alatt az adott oszlopban, hanem a fölötte lévő elemeket is. Mielőtt a kinullázást elkezdjük, elosztjuk az aktuális egyenletet a főátlóban lévő együtthatóval (ha az nem 0), hogy a főátlóban a végén csupa 1 legyen.
Az eljárás műveletigénye: $n^3 + O(n^2)$.

Az előbbi két algoritmus végrehajtása során világos, hogy az aktuális egyenletnek a főátlóban álló eleme, az ún. *főelem* fontos szerepet tölt be az eljárásban. Ha az rendre nem 0, akkor rendben megy az algoritmus, de ha 0, akkor 0-val nem tudunk osztani, így helyette új főelemet kell választani, ha tudunk. Ennek két módja használatos:
a) *részleges főelemkiválasztás*: sorcserével az aktuális egyenlet helyett olyant választunk, hogy a főelem ne legyen 0.
b) *teljes főelemkiválasztás*: sor és oszlopcserét is használhatunk ahhoz, hogy a főelem ne legyen 0. Ekkor az oszlopcserék miatt az algoritmus során, és a visszahelyettesítéssel kapott vektorban vissza kell állítani az ismeretlenek helyes sorrendjét.

A Jordan-eliminációt használhatjuk reguláris mátrixok invertálására is. Az invertálás művelete felfogható úgy is, *mint n darab lineáris egyenletrendszer szimultán megoldása.* Meg lehet mutatni, hogy $O(n^3)$ művelettel egy reguláris mátrix inverze Jordan-eliminációval meghatározható.

Trianguláris felbontások

Definíció. Az *A* kvadratikus mátrix *<u>trianguláris felbontás</u>*án (dekompozícióján) az *A=MR* alakú előállítást értjük, ahol *R* felső trianguláris mátrix. Ha *M*

- alsó egység-trianguláris mátrix (vagyis olyan mátrix, amelynek főátlójában minden elem 1 és a főátló fölött minden elem 0), akkor *<u>LR felbontás</u>*ról beszélünk (más jelölésben LU-felbontás).
- R^T, akkor *<u>Cholesky-felbontás</u>*ról beszélünk.

LR-felbontás

Egy kvadratikus mátrixnak nincs, pontosan 1, vagy végtelen sok LR-felbontása is létezhet.

a) Az $A=\begin{pmatrix}0 & 0\\ 1 & 1\end{pmatrix}$ mátrixnak nincs LR-felbontása.

b) $A\ B=\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}=\begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}$ a mátrix egyetlen LR-felbontása.

c) $A\ C=\begin{pmatrix}0 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}1 & 0\\ x & 1\end{pmatrix}\begin{pmatrix}0 & 1\\ 0 & 1-x\end{pmatrix}$ bármilyen valós *x*-re teljesül.

Állítás. *Reguláris mátrix LR felbontása egyértelmű.*

Bizonyítás. Ha lenne két különböző felbontása *A*-nak pl. $A=L_1R_1=L_2R_2$, akkor mivel A reguláris, így a determinánsok szorzástétele alapján L_1, L_2, R_1, R_2 szintén reguláris mátrixok, vagyis mind invertálhatóak. Ezek szerint teljesülnie kell az

$$L_2^{-1}L_1=R_2R_1^{-1}$$

összefüggésnek is. A baloldalon egy alsó egység-trianguláris mátrix áll, a jobb oldalon egy felső trianguláris mátrix, és ez csak úgy lehet, hogy ha mindkét oldalon az egységmátrix van. Ez azonban azt jelentené, hogy $L_1=L_2$, $R_1=R_2$.

Egy kvadratikus mátrix LR felbontása megkapható a Gauss-elimináció segítségével is. Az alábbi egyenletrendszer megoldása: x=(3,2,-1,1)

$$\begin{aligned}x_1-x_2+3x_3+2x_4&=0\\ 2x_1+8x_3+5x_4&=3\\ -x_1+7x_2+4x_3+4x_4&=11\\ 3x_1-x_2+13x_3+12x_4&=6\end{aligned}$$

A Gauss-eliminációval való megoldás az alábbi lépéseken keresztül jut el a végső felső trianguláris mátrixú lineáris egyenletrendszerig:

$$\underset{A}{\left(\begin{array}{cccc|c}1 & -1 & 3 & 2 & 0\\ 2 & 0 & 8 & 5 & 3\\ -1 & 7 & 4 & 4 & 11\\ 3 & -1 & 13 & 12 & 6\end{array}\right)}\sim\underset{A_1}{\left(\begin{array}{cccc|c}1 & -1 & 3 & 2 & 0\\ 0 & 2 & 2 & 1 & 3\\ 0 & 6 & 7 & 6 & 11\\ 0 & 2 & 4 & 6 & 6\end{array}\right)}\sim\underset{A_2}{\left(\begin{array}{cccc|c}1 & -1 & 3 & 2 & 0\\ 0 & 2 & 2 & 1 & 3\\ 0 & 0 & 1 & 3 & 2\\ 0 & 0 & 2 & 5 & 3\end{array}\right)}\sim\underset{A_3}{\left(\begin{array}{cccc|c}1 & -1 & 3 & 2 & 0\\ 0 & 2 & 2 & 1 & 3\\ 0 & 0 & 1 & 3 & 2\\ 0 & 0 & 0 & -1 & -1\end{array}\right)}$$

Az egyes lépések algebrailag is leírhatók egy-egy ún. *eliminációs mátrix*szal való beszorzással:

$$M_1=\begin{pmatrix}1 & 0 & 0 & 0\\ -2 & 1 & 0 & 0\\ 1 & 0 & 1 & 0\\ -3 & 0 & 0 & 1\end{pmatrix}\qquad M_2=\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & -3 & 1 & 0\\ 0 & -1 & 0 & 1\end{pmatrix}\qquad M_3=\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & -2 & 1\end{pmatrix}$$

$$A_1=M_1A,\qquad A_2=M_2A_1,\qquad A_3=M_3A_2\qquad (*)$$

Egy rövid kitérőként nézzük meg az eliminációs mátrixokkal való számolás néhány tulajdonságát.

Definíció: *Az* $M_j \in R^{n\times n}$ *mátrix <u>eliminációs mátrix</u>, ha* $M_j = I + m^{(j)}e_j^T$, *ahol* $m^{(j)}$ *egy olyan valós vektor, amelynek az első j komponense* 0 $(1 \le j \le n)$.

Az M_j *eliminációs mátrix inverze* $M_j = I - m^{(j)}e_j^T$. *Az* M_j *és az* M_k $(j<k)$ *eliminációs mátrixok szorzata* $I + m^{(j)}e_j^T + m^k e_k^T$.

Visszatérve az előbb három (*) egyenlőséghez $A_3 = M_3A_2 = M_3M_2A_1 = M_3M_2M_1A$ adódik. Mivel egy eliminációs mátrix mindig invertálható (hiszen determinánsa 1), így igaz az $A = M_1^{-1}M_2^{-1}M_3^{-1}A_3$ egyenlőség is. Láthattuk, hogy eliminációs mátrixokat könnyű invertálni és összeszorozni. Az $M_1^{-1}M_2^{-1}M_3^{-1}$ szorzat alsó egység-trianguláris mátrix lesz, így az előbbi egyenlőség megadja A egy LR dekompozícióját is:

$$\begin{pmatrix} 1 & -1 & 3 & 2 \\ 2 & 0 & 8 & 5 \\ -1 & 7 & 4 & 4 \\ 3 & -1 & 13 & 12 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ -1 & 3 & 1 & 0 \\ 3 & 1 & 2 & 1 \end{pmatrix}\begin{pmatrix} 1 & -1 & 3 & 2 \\ 0 & 2 & 2 & 1 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Van azonban olyan algoritmus is, amellyel rendezettebben kiszámolhatjuk az LR felbontást. Ezt az eljárást szokás „**parketta algoritmusnak**” is hívni. Az eljárás formalizálható két formulában:

$$r_{jk} = a_{jk} - \sum_{p=1}^{j-1} l_{jp}r_{pk} \qquad (k=j,j+1,\ldots,n)$$

$$l_{ij} = \frac{1}{r_{jj}}(a_{ij} - \sum_{p=1}^{j-1} l_{ip}r_{pj}) \qquad (i=j+1,\ldots,n) \qquad (j=1,2,\ldots,n)$$

Az eljárás úgy működik, hogy először a fenti formulák alapján meghatározzuk az *R* mátrix első sorát, majd az *L* első oszlopát, utána az *R* második sorát, majd *L* második oszlopát és így tovább (mintha parkettákat tennénk le). **Példa.** Határozzuk meg az előbbi mátrix LR-felbontását a parketta algoritmussal is!

Cholesky-felbontás.

Csak szimmetrikus mátrixnak létezik Cholesky-felbontása. Valóban, hiszen ha $A = R^TR$, akkor $A^T = (R^TR)^T = R^T(R^T)^T = R^TR = A$, tehát *A* szimmetrikus.

Állítás: *Tetszőleges A szimmetrikus reguláris mátrix Cholesky-felbontása az R sorainak előjelétől eltekintve egyértelmű.*

Bizonyítás. Tegyük fel, hogy A-nak van két különböző Cholesky-felbontása: $A = R_1^T R_1 = R_2^T R_2$. Mivel a felbontásban szereplő minden mátrix is reguláris, így $\left(R_2^T\right)^{-1} R_1^T = R_2 \left(R_1\right)^{-1}$. Itt a bal oldalon alsó, a jobb oldalon felső trianguláris mátrix áll, ami nyilván csak úgy lehet, ha mind a kettő egy D diagonális mátrixszal azonos. Ekkor $D^2 = DD^T = \left(R_2^T\right)^{-1} R_1^T \left(R_2 \left(R_1\right)^{-1}\right)^T = \left(R_2^T\right)^{-1} R_1^T \left(R_1^T\right)^{-1} R_2^T = I$, ami azt jelenti, hogy $d_{jj} = \pm 1$. Tehát $R_2 = DR_1$ és $R_2^T = R_1^T D$.

Algoritmus pozitív definit mátrix Cholesky-felbontásának megadására:

Cholesky(A,R)

1. A a felbontandó mátrix
2. R a jobboldali felső trianguláris mátrix
3. *n*=Msor(A)
4. for *j*=1 to *n*
5. do
6. $r_{jj} = a_{jj}$
7. for *l*=1 to *j*-1
8. do
9. $r_{jj} = r_{jj} - r_{lj}^2$
10. od
11. $r_{jj} = sqrt(r_{jj})$
12. for *k*=*j*+1 to *n*
13. do
14. $r_{jk} = a_{jk}$
15. for *l*=1 to *j*-1
16. do
17. $r_{jk} = r_{jk} - r_{lj} r_{lk}$
18. od
19. $r_{jk} = r_{jk} / r_{jj}$
20. od
21. od
22. return R

Példa. Az algoritmust használva az alábbi felbontást nyerhetjük:

$$\begin{pmatrix} 4 & 2 & -2 & 4 \\ 2 & 10 & 5 & 2 \\ -2 & 5 & 6 & -3 \\ 4 & 2 & -3 & 6 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ -1 & 2 & 1 & 0 \\ 2 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & -1 & 2 \\ 0 & 3 & 2 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

3. téma
Mátrixok sajátértékei és sajátvektorai, közelítő módszerek

A matematika több területén (pl. differenciálegyenletek megoldásakor) hasznos egy $x \to Ax$ leképezés (ahol $A \in C^{n\times n}, x \in C^n$) olyan vektorainak ismerete, amelyeket a leképezés a *saját* hatásvonalán hagy meg. Az ilyen vektorokat az *A* mátrix *sajátvektor*ainak hívjuk. Az iránya és a nagysága változhat a vektornak, így csak azt követeljük meg, hogy eleget tegyen az $A\underline{v} = \lambda\underline{v}$ egyenlőségnek, valamilyen λ komplex számmal. Pontosabban tehát

Definíció. *Az $A \in C^{n\times n}$ mátrixnak $\lambda \in C$ sajátértéke, ha létezik olyan $\underline{v} \in C^n$ zéróvektortól különböző vektor, amelyre*

$$A\underline{v} = \lambda\underline{v}.$$

A $\underline{v}$ vektort ekkor az A mátrix λ sajátértékéhez tartozó sajátvektornak mondjuk.

Példa. A $\begin{pmatrix} 2 & 1 \\ 2 & 3 \end{pmatrix}$ mátrixnak két sajátértéke van: $\lambda_1 = 1$ és $\lambda_2 = 4$.

A λ_1-hez tartozó sajátvektor lehet minden olyan $\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ vektor, amelyre $v_1 + v_2 = 0$.

A λ_2-höz tartozó sajátvektor lehet minden olyan $\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ vektor, amelyre $v_2 = 2v_1$.

Természetesen leszámítva az előbbi két esetben a zéróvektort.

Tétel. *A $\lambda \in C$ akkor és csak akkor sajátértéke az $A \in C^{n\times n}$ mátrixnak, ha a*

$$\det(A - \lambda I) = 0.$$

Bizonyítás

a) Ha $\lambda \in C$ sajátértéke az $A \in C^{n\times n}$ mátrixnak, akkor van olyan $\underline{v} \neq \underline{0}$ vektor, amelyre $A\underline{v} = \lambda\underline{v}$. Mindkét oldalból levonva $\lambda\underline{v}$-t, kapjuk az $A\underline{v} - \lambda\underline{v} = \underline{0}$ összefüggést, majd $\underline{v}$-t kiemelve $(A - \lambda I)\underline{v} = \underline{0}$ adódik. A feltételünk szerint ennek a homogén lineáris egyenletrendszernek a $\underline{v}$ nem triviális megoldása, így az egyenletrendszer mátrixnak determinánsa 0 kell hogy legyen.

b) Ha a tételben szereplő determináns 0 valamilyen $\lambda \in C$ értékre, akkor az $(A - \lambda I)\underline{v} = \underline{0}$ homogén lineáris egyenletrendszernek van nem triviális megoldása. A λ azonban egyben sajátértéke is az A-nak, hiszen az előbbi átalakításokat visszafelé végrehajtva éppen az $A\underline{v} = \lambda\underline{v}$ összefüggés adódik egy $\underline{v} \neq \underline{0}$ vektorra.

Definíció. *A* $\det(A-\lambda I)$ *λ-ra nézve n-edfokú polinomot* $p_A(\lambda)$*-val jelöljük és A karakterisztikus polinomjának nevezzük.*

Példa. A $\begin{pmatrix}2 & 1\\ 2 & 3\end{pmatrix}$ mátrixnak a $\det\begin{pmatrix}2-\lambda & 1\\ 2 & 3-\lambda\end{pmatrix}=\lambda^2-5\lambda+4$ a karakterisztikus polinomja.

Megjegyzések.
a) Egy mátrix sajátértékei a karakterisztikus polinomjának gyökei.
b) Az algebra alaptételéből következik, hogy (multiplicitással számolva) egy $n\times n$-es mátrixnak pontosan *n* sajátértéke van a komplex számtestben.
c) A Ruffini-Abel tétel szerint $n\geq 5$ esetén nincs általános megoldó képlet egy polinom gyökeinek algebrai megadására. Nem létezik olyan *véges* algoritmus, amellyel egy *tetszőleges* mátrix sajátértékeit *pontosan* meg lehetne határozni.

Tétel. *Egy felső vagy alsó trianguláris mátrix sajátértékei megegyeznek a mátrix főátlójában álló elemekkel.*

Bizonyítás. Ha *A* alsó vagy felső trianguláris mátrix, akkor $\det(A-\lambda I)$ értéke éppen a főátlóbeli elemek szorzata, amely egyben a karakterisztikus polinom gyöktényezős előállítása is, így gyökei az *A* főátlóbeli elemei.

Definíció.

a) *Az* A *mátrix nyoma a* $tr\,A=\sum_{i=1}^{n}a_{ii}$ *érték.* (A tr jelölés az angol trace-ből ered.)

b) *Az* A *mátrix spektrálsugara* $\rho(A)=\max_{1\leq i\leq n}|\lambda_i|$.

c) *Az* A *mátrix spektruma* A *sajátértékeinek halmaza.*

Példa. A $\begin{pmatrix}2 & 1\\ 2 & 3\end{pmatrix}$ mátrix nyoma 5, spektrálsugara 4.

Tetszőleges $A\in C^{n\times n}$ mátrixra $\mathrm{tr}\,A=\sum_{i=1}^{n}\lambda_i$ és $\det A=\prod_{i=1}^{n}\lambda_i$.

Vektornormák

Definíció: *Egy* $||.||: R^n\to R$ *leképezést vektornormának mondunk, ha*
(i) $||v||>0$ *, minden* $v\neq \underline{0}$*-ra.*
(ii) $||\alpha v||=|\alpha|\,||v||$*, bármely* α *valós számra és v vektorra.*
(iii) $\|v_1+v_2\|\leq\|v_1\|+\|v_2\|$ *bármely* v_1, v_2 *vektorra.*

A vektornormák egyik fajtája a **p-norma** (*p* pozitív egész szám)**.**

$$\|v\|_p = \left(\sum_{i=1}^{n} |v_i|^p \right)^{1/p}$$

A p=1 esetben $\|v\|_1 = \sum_{i=}^{n} |v_i|$

A p=2 esetben **euklideszi normáról** szoktunk beszélni.

Gyakran használatos még a

$$\|v\|_\infty = \max_{1 \le i \le n} |v_i| \text{ vektornorma is.}$$

Mátrixnormák

Definíció. *Egy* $||.||: R^{n \times n} \to R$ *leképezést* <u>*mátrixnormának*</u> *mondunk, ha*
(i) $\|A\| > 0$, *ha A nem a zérómátrix.*
(ii) $\|\alpha A\| = |\alpha| \|A\|$, *ahol α tetszőleges valós szám.*
(iii) $\|A_1 + A_2\| \le \|A_1\| + \|A_2\|$, *bármely* A_1, A_2 *valós mátrixokra.*
(iv) $\|AB\| \le \|A\| \|B\|$, *bármely A,B valós mátrixokra*.

Minden vektornormából származtathatunk mátrixnormát az alábbi módon:

$$\|A\| = \sup_{x \neq 0} \frac{\|Ax\|_v}{\|x\|_v} \quad \left(= \max_{x \neq 0} \frac{\|Ax\|_v}{\|x\|_v} = \max_{\|x\|_v = 1} \|Ax\|_v \right)$$

Könnyen látható, hogy az I egységmátrixhoz minden származtatott mátrixnormában az 1 érték tartozik: $\|I\| = \max_{\|x\|_v = 1} \|x\|_v = 1$.

Nevezetes mátrixnormák

$\|A\|_1 = \max_{\|x\|_1 = 1} \|Ax\|_1 \left(= \max_{1 \le j \le n} \sum_{i=1}^{n} |a_{ij}| \right)$ (oszlopnorma)

$\|A\|_\infty = \max_{\|x\|_\infty = 1} \|Ax\|_\infty \left(= \max_{1 \le i \le n} \sum_{j=1}^{n} |a_{ij}| \right)$ (sornorma)

$\|A\|_F = \left(\sum_{i=1}^{n} \sum_{k=1}^{n} |a_{ik}|^2 \right)^{1/2}$ (Frobenius-norma)

$\|A\|_M = n \max_{1 \le i,k \le n} |a_{ik}|$ (maximum-norma)

Definíció: *Egy* $\|.\|_m$ *mátrixnormát* <u>*kompatibilisnek*</u> *mondunk egy* $\|.\|_v$ *vektornormával, ha minden A valós mátrixra és x vektorra teljesül az*

$$\|Ax\|_v \le \|A\|_m \|x\|_v \text{ egyenlőtlenség.}$$

Tétel. *Bármely mátrixnormához megadható vele kompatibilis vektornorma.*

Bizonyítás. Legyen $\|.\|_m$ mátrixnorma. Egy vele kompatibilis vektornormát így definiálhatunk: $\|x\|_v = \|X\|_m$, ahol $X = xe_1^T$. Tényleg vektormát kapunk így, mivel

a) $\|x\| > 0$, ha $x \neq 0$.

b) $\|\alpha x\|_v = \|\alpha(xe_1^T)\|_m = |\alpha|\|xe_1^T\|_m = |\alpha|\|x\|_v$

c) $\|x+y\|_v = \|xe_1^T + ye_1^T\|_m \leq \|xe_1^T\|_m + \|ye_1^T\|_m = \|x\|_v + \|y\|_v$

A bevezetett vektornorma kompatibilis lesz az adott mátrixnormával, mivel $\|Ax\|_v = \|Axe_1^T\|_m \leq \|A\|_m \|xe_1^T\|_m = \|A\|_m \|x\|_v$.

Tétel. *Egy A mátrix spektrálsugara nem lehet nagyobb valamely mátrixnormának A-ban felvett értékénél, vagyis* $\rho(A) \leq \|A\|$.

Bizonyítás. Ha azt igazoljuk, hogy bármely $A \in C^{n\times n}$ mátrix bármely λ sajátértékére $|\lambda| \leq \|A\|$, akkor készen vagyunk, hiszen ekkor speciálisan a spektrálsugárra is fennáll az egyenlőtlenség. Ez viszont igaz, hiszen az előbbi tétel szerint minden mátrixnormához megadható vele kompatibilis vektornorma, így $|\lambda|\|x\|_v = \|\lambda x\|_v = \|Ax\|_v \leq \|A\|\|x\|_v$, amit $\|x\|_v > 0$-val leosztva a bizonyítandó állítást kapjuk.

Tetszőleges valós A kvadratikus mátrixra $\|A\|_2 = \left(\rho\left(A^T A\right)\right)^{1/2}$, ahol $\|A\|_2$ származtatott mátrixnorma. Ha A szimmetrikus, akkor $\|A\|_2 = \rho(A)$, így a $\|.\|_2$ normát szokás *spektrálnormá*nak is mondani.

A sajátértékeknek a komplex számsíkon való elhelyezkedését jellemzi a **Gersgorin tétel.** *Legyen* $A \in C^{n\times n}$ *tetszőleges mátrix,* $r_i = \sum_{j=1, j\neq i}^{n} |a_{ij}|$ $(i = 1,...,n)$ és $D_i = \{z \in C : |z - a_{ii}| \leq r_i\}$ $(i = 1,...,n)$. Ekkor az *A* mátrix minden sajátértékére fennáll, hogy $\lambda \in \bigcup_{i=1}^{n} D_i$. A D_i köröket Gersgorin-köröknek nevezik.

Definíció. *Legyen* $A \in C^{n\times n}$ *tetszőleges mátrix, T pedig tetszőleges reguláris mátrix. Az* $A \to T^{-1}AT$ *hozzárendelést <u>hasonlósági transzformációnak</u> nevezzük. Az A mátrix <u>hasonló</u> a B-hez (jelölés A~B), ha létezik olyan reguláris T mátrix, hogy*

$$B = T^{-1}AT .$$

Tétel. *Ha A~B, akkor* $p_A(\lambda) = p_B(\lambda)$ *, így sajátértékeik is megegyeznek.*

Bizonyítás: Ha *A~B,* akkor létezik olyan *T* reguláris mátrix, amelyre $B = T^{-1}AT$. Ekkor $p_A(\lambda) = \det(A - \lambda I) = \det(TT^{-1}ATT^{-1} - \lambda ITT^{-1}) = \det(T(T^{-1}AT - \lambda I)T^{-1}) =$
$\det(T)\det(T^{-1}AT - \lambda I)\det(T^{-1}) = \det(TT^{-1})\det(T^{-1}AT - \lambda I) = \det(B - \lambda I) = p_B(\lambda).$

Az előbbi tétel megfordítása nem feltétlenül teljesül, hiszen a $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ és a $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ mátrixok karakterisztikus polinomjaik megegyeznek, mégsem hasonlók.

Mivel nem létezik olyan algoritmus, amely minden mátrixnak valamennyi sajátértékét véges számú lépésben pontosan meghatározza, így különösen fontosak az olyan eljárások, amelyekkel jól tudjuk közelíteni őket.

Hogyan tudnánk ilyen algoritmusokat kitalálni?

Az alapötlet lehet hasonló ahhoz, mint amit lineáris egyenletrendszereknél használtunk: Próbáljuk meg olyan alakra hozni a mátrixot, hogy a sajátértékek onnan már könnyen leolvashatók legyenek.

Tipikusan a trianguláris mátrixok olyanok, amelyekről ránézésre le tudjuk olvasni a sajátékeket, hiszen azok megegyeznek a főátlóban álló elemekkel. Lineáris egyenletrendszerek eliminációval való megoldásánál fontos szerepet kapott, hogy meg tudtunk adni a kiinduló egyenletrendszerrel *ekvivalens* olyan lineáris egyenletrendszert, amelynek megoldása már egyszerű volt. Ehhez a felső trianguláris mátrixú egyenletrendszerhez egy olyan egyenletrendszer-sorozaton keresztül jutottunk, amelyben bármely két egyenletrendszer ekvivalens volt egymással.

A sajátérték probléma közelítő megoldásánál az előbbi ekvivalencia fogalmat a hasonlóság helyettesíti. Itt is definiálhatunk – jó esetben – egy mátrix sorozatot, ahol a sorozat bármely két eleme hasonló egymáshoz és a sorozat határértéke egy felső trianguláris mátrix, amelynek főátlójában a kiindulási mátrix sajátértékei vannak.

A sorozatot a mátrix trianguláris felbontásaival definiálhatjuk:

$$A = M_1 R_1$$
$$A_1 = R_1 M_1 = M_2 R_2$$
$$A_2 = R_2 M_2 = M_3 R_3$$
$$\dots$$

Tétel. Az LR-algoritmus konvergencia-tétele (bizonyítás nélkül):

Legyen $A_1 \in R^{n\times n}$ *olyan reguláris mátrix, amelynek sajátértékei valósak és egyszeresek. Tegyük fel, hogy elő tudjuk állítani az alábbi sorozatot, vagyis minden lépésben létezik a mátrix LR trianguláris felbontása:*

$$A_k = L_k R_k$$
$$A_{k+1} = R_k L_k .$$

Legyen az X mátrix A_1*-t diagonizáló, vagyis* $X^{-1} A_1 X = diag(\lambda_1, \lambda_2, ..., \lambda_n)$ *és létezzen az X-nek és az* X^{-1}*-nek LR trianguláris felbontása.*

Ekkor az $\{A_k\}_{k=1}^{\infty}$, $\{R_k\}_{k=1}^{\infty}$ *és* $\{L_k\}_{k=1}^{\infty}$ *mátrixsorozatok konvergensek és*

$$\lim_{k\to\infty} A_k = \lim_{k\to\infty} R_k = \begin{pmatrix} \lambda_1 & * & * & * & * \\ 0 & \lambda_2 & * & * & * \\ 0 & 0 & \lambda_3 & * & * \\ ... & ... & ... & ... & * \\ 0 & 0 & ... & 0 & \lambda_n \end{pmatrix}, \qquad \lim_{k\to\infty} L_k = I .$$

Tétel. Az $R^T R$ - algoritmus konvergencia-tétele (bizonyítás nélkül)**:**

Legyen $A_1 \in R^{n\times n}$ *pozitív definit mátrix. A Cholesky-felbontáson alapuló alábbi sorozat*

$$A_k = R_k^T R_k$$
$$A_{k+1} = R_k R_k^T$$

konvergál egy olyan diagonális mátrixhoz, amely A_1 *sajátértékeit tartalmazza, mindegyiket annyiszor amennyi a multiplicitása.*

Megjegyzés.

A trianguláris felbontásoknak egyéb alkalmazásaik is vannak.

a) Ha A-nak ismert az LR felbontása, akkor az Ax=b lineáris egyenletrendszer helyett tekinthetjük az Ly=b, Rx=y egyenletrendszereket, amelyeket már pusztán visszahelyettesítéssel megoldhatunk.

b) Ha A-nak ismert az LR felbontása, akkor A determinánsa is könnyen meghatározható, hiszen az nem lesz más, mint az R főátlójában lévő elemek szorzata.

4. téma
Egyenletek és egyenletrendszerek közelítő megoldása iterációs módszerekkel

Skaláriteráció

Definíció: *Legyen* x_0 *valós szám, f valós egyváltozós függvény! Az*

$$x_1 = f(x_0)$$
$$x_2 = f(x_1)$$
$$x_3 = f(x_2)$$
$$\dots$$
$$x_n = f(x_{n-1})$$
$$\dots$$

sorozatot az f függvény x_0 *kezdőponthoz tartozó iterációsorozatának nevezzük.*

Lássunk egy egyszerű példát! Legyen $f(x) = ax + b$ és x_0 valós szám. Ekkor

$$x_1 = ax_0 + b$$
$$x_2 = a^2 x_0 + ab + b$$
$$x_3 = a^3 x_0 + a^2 b + ab + b$$
$$\dots$$
$$x_n = a^n x_0 + (a^{n-1} + a^{n-2} + \dots + a^2 + a + 1)b$$
$$\dots$$

Tehát $x_n = \begin{cases} a^n x_0 + \dfrac{1-a^n}{1-a} b, & ha \quad a \neq 1 \\ x_0 + nb, & ha \quad a = 1 \end{cases}$

Természetesen merül fel a kérdés, hogy mikor lesz konvergens az előbbi sorozat, ha *n* tart a végtelenbe. Nem nehéz belátni a következőt:

$$\lim_{n\to\infty} x_n = \begin{cases} \dfrac{1}{1-a} b, & ha \quad |a| < 1 \\ x_0, & ha \quad a = 1, b = 0 \\ nincs \quad véges \quad határérték, & különben \end{cases}$$

Amire különösen érdemes felfigyelni az az, hogy ha |a|<1, akkor a határérték független az x_0-tól. Tehát bármilyen kezdőpontból is indul ki a sorozat, az mindig egy konstanshoz, az $s = \dfrac{1}{1-a} b$-hez fog tartani. Az (*s*,*s*) pont az y=x és y=ax+b egyenesek metszéspontja, tehát megoldása az x=f(x) egyenletnek.

Definíció: *Az x=f(x) egyenlet megoldásait f fixpontjainak nevezzük.*

Tétel: *Legyen konvergens az f valós függvény egy iteráció-sorozata és tartson s-hez. Tegyük fel, hogy f folytonos és értelmezési tartománya zárt. Ekkor s fixpontja lesz az f-nek!*

Bizonyítás: $s = \lim_{n\to\infty} x_n = \lim_{n\to\infty} f(x_{n-1}) = \lim_{n\to\infty} f(x_n) = f(\lim_{n\to\infty} x_n) = f(s)$.

Természetesen felmerülő kérdések:

1. Létezik-e fixpontja egy *f* függvénynek?
2. Egyértelmű-e a fixpontja *f*-nek?
3. Bármely kezdőpontból kiindulva a fixponthoz konvergál-e az *f* iteráció-sorozata?

Ahhoz, hogy biztosítsuk, hogy az iteráció-sorozata biztosan létezzék az *f*-nek, feltesszük, hogy $R_f \subseteq D_f$. Az előbbi állításból kitűnt, hogy szükségünk van *f* folytonosságára és arra is, hogy D_f zárt legyen. Ha feltennénk azt is, hogy D_f korlátos, rögtön következne az, hogy *f*-nek biztosan létezik fixpontja. Elég csak meggondolni a korlátos zárt intervallumon folytonos függvények tulajdonságait.

Tétel: *Az* $f:[a,b]\to[a,b]$ *folytonos függvénynek létezik fixpontja.*

Bizonyítás: Legyen g(x)=x-f(x). Mivel $a \le f(a)$ és $f(b) \le b$, így $g(a) \le 0$ és $g(b) \ge 0$. Ha g(a)=0, vagy g(b)=0 akkor készen vagyunk, hiszen akkor vagy *a* vagy *b* fixpontja lesz *f*-nek. Tegyük fel, hogy g(*a*)<0 és g(*b*)>0. A Bolzano-tétel szerint azonban ekkor létezik olyan *c* pont *a* és *b* között, hogy g(*c*)=0. Ez viszont pontosan azt jelenti, hogy *c* fixpontja *f*-nek.

Ha nem tesszük fel D_f korlátosságát, akkor nem biztos, hogy létezik fixpontja *f*-nek. Ha a D_f korlátosságát elvetjük, de előírjuk, hogy *f* teljesítse a Lipschitz-feltételt, akkor viszont az előbbi három kérdésre igennel válaszolhatunk. Vagyis:

Ha az f folytonos függvényre $R_f \subseteq D_f$, D_f *zárt és f teljesíti az* $|f(x_1) - f(x_2)| \le L\,|x_1 - x_2|$ *Lipschitz-feltételt, ahol* $0 \le L < 1$, *és* $x_1, x_2 \in D_f$, *akkor f-nek pontosan 1 fixpontja van, és bármely* $x_0 \in D_f$*-re f iterációsorozata a fixponthoz konvergál.*

Nemlineáris egyenletek megoldása fixpontiterációval

Az egyenletek és egyenletrendszerek gyökkeresésének problémája visszavezethető alkalmas függvények fixpontjának meghatározására.

Tétel: *Az f(x)=0 egyenletnek s gyöke, akkor és csak akkor, ha a g(x)=x+h(x)f(x) függvénynek s fixpontja, ahol h egy olyan valós függvény, amelynek nincs valós zérushelye.*

Bizonyítás: Ha *s* gyöke az f(*x*)=0 egyenletnek, akkor f(*s*)=0, így g(*s*)=*s*+h(*s*)0=*s*, tehát *s* fixpontja *g*-nek. Ha *s* fixpontja *g*-nek, akkor *s*=g(*s*)=*s*+h(*s*)f(*s*), tehát f(*s*)=0. Így *s* zérushelye *f*-nek.

Ha a *h* függvénynek az *f* deriváltjának a reciprokának az ellentettjét választjuk, akkor az érintőmódszerhez jutunk. Ha a derivált helyett azt az *f* egy differenciahányadosával helyettesítjük, akkor jutunk el a szelőmódszerhez.

Érintőmódszer (Newton-módszer)

Az $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ sorozattal közelíthetjük az *f(x)=0* egyenlet egy gyökét.

Tétel: *Az* x_{n+1} *az f függvényhez az* $(x_n, f(x_n))$ *pontban húzott érintőnek és az x-tengely metszéspontjának abszcisszája.*

Bizonyítás: Legyen az érintő egyenlete *y=mx+b*. Az egyenlet meredeksége $f'(x_n)$, így $y = f'(x_n)x + b$. Mivel az egyenes átmegy az $(x_n, f(x_n))$ ponton, így $f(x_n) = f'(x_n)x_n + b$. Tehát $b = f(x_n) - f'(x_n)x_n$. Az érintő egyenlete így $y = f'(x_n)x + f(x_n) - f'(x_n)x_n$. Az x_{n+1} zérushelye eleget kell, hogy tegyen az $0 = f'(x_n)x_{n+1} + f(x_n) - f'(x_n)x_n$ összefüggésnek, amiből $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ adódik.

Példa: Négyzetgyökvonás Newton-módszerrel

A *c* pozitív szám négyzetgyökét meghatározni, nem más, mint az $f(x) = x^2 - c$ pozitív gyökét kiszámolni. Az $x_0 = 1$ kezdőértéket és c=2-t választva

$$x_{n+1} = x_n - \frac{x_n^2 - c}{2x_n} = \frac{1}{2}\left(x_n + \frac{c}{x_n}\right)$$ iteráció-sorozattal az

$x_0 = 1.00000000$ $x_1 = 1.50000000$ $x_2 = 1.41666667$ $x_3 = 1.41421569$
$x_4 = 1.41421356$ itt már minden kiírt jegy pontos.

Szelőmódszer

$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$. Itt az $(x_n, f(x_n))$ és $(x_{n-1}, f(x_{n-1}))$ pontokat összekötő szelőnek az x-tengellyel vett metszéspontjának abszcisszája lesz x_{n+1}.

Lineáris egyenletrendszerek megoldása iterációval

Az $Ax=a$ négyzetes mátrixú lineáris egyenletrendszert ekvivalens átalakításokkal hozzuk $x = Bx + b$ alakra. Válasszunk alkalmas x_0 kezdővektort majd képezzük az $x_{n+1} = Bx_n + b$ iterációsorozatot.

Tétel (bizonyítás nélkül): *Az $x_{n+1} = Bx_n + b$ iteráció globálisan konvergál az $Ax=a$ egy megoldásához, ha $\rho(B) < 1$. (Az előbbi feltételt lehet azzal is helyettesíteni, hogy van olyan ||.|| mátrixnorma, hogy ||B||<1.)*

Kérdés: Hogyan állítsuk elő az $Ax=a$ lineáris egyenletrendszerből a vele ekvivalens $x=Bx+b$ lineáris egyenletrendszert?

Tekintsük A-nak egy $A=R-S$ reguláris szétvágását, ahol R reguláris mátrix. Ekkor

$$(R-S)x=a$$
$$Rx=Sx+a$$
$$x = R^{-1}Sx + R^{-1}a$$

Így

$$B = R^{-1}S\,,\ b = R^{-1}a\,.$$

A továbbiakban feltesszük, hogy A erősen reguláris, vagyis A reguláris és a főátlójában nincs nulla. Hogyan kaphatjuk meg A egy reguláris szétvágását? Tekintsük az $A=D-L-U$, ahol D diagonális mátrix, L alsó trianguláris, U felső trianguláris mátrixok.

$$D = \begin{pmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn} \end{pmatrix} \qquad L = \begin{pmatrix} 0 & 0 & \dots & 0 \\ -a_{21} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -a_{n1} & -a_{n2} & \dots & 0 \end{pmatrix} \qquad U = \begin{pmatrix} 0 & -a_{12} & \dots & -a_{1n} \\ 0 & 0 & \dots & -a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

Jacobi-iteráció: R=D, S=L+U

Ekkor

$$B_J = D^{-1}(L+U)$$
$$b_J = D^{-1}a$$
$$x_n = D^{-1}(L+U)x_{n-1} + D^{-1}a$$

Gauss-Seidel-iteráció: R=D-L, S=U

Ekkor

$$B_S = (D-L)^{-1}U$$
$$b_S = (D-L)^{-1}a$$
$$x_n = (D-L)^{-1}Ux_{n-1} + (D-L)^{-1}a$$

5. téma
Polinomok zérushelyei

1. Feladat: Adjunk meg olyan síkbeli tartományt, amely tartalmazza egy adott polinom *legalább egy* gyökét!

Tétel: *Legyen p(x) egy polinom és legyen z nem zérushelye a p deriváltjának. Ekkor p-nek van legalább egy gyöke a z körüli* $n\frac{|p(z)|}{|p'(z)|}$ *sugarú körben.*

Bizonyítás: *p* gyöktényezős alakját differenciálva kapjuk a

$p'(z)=\frac{p(z)}{z-x_1}+\frac{p(z)}{z-x_2}+...+\frac{p(z)}{z-x_n}$ összefüggést, amit felhasználva

$$\left|\frac{p'(z)}{p(z)}\right|=\left|\frac{1}{z-x_1}+\frac{1}{z-x_2}+...+\frac{1}{z-x_n}\right|\le\left|\frac{1}{z-x_1}\right|+\left|\frac{1}{z-x_2}\right|+...+\left|\frac{1}{z-x_n}\right|\le n\frac{1}{\min\limits_{1\le i\le n}|z-x_i|}$$

Ebből adódóan már következik az állítás:

$$\min_{1\le i\le n}|z-x_i|\le n\left|\frac{p(z)}{p'(z)}\right|.$$

Példa: A $p(x)=x^4+x^3+2x^2+4x-8$ polinomnak az

$$x_1=1, x_2=-2, x_3=2i, x_4=-2i$$

a gyökei, amelyek közül egy benne van egy origó körüli 8 sugarú körben.

2. Feladat: Adjunk meg olyan síkbeli tartományt, amely tartalmazza egy adott polinom *összes gyökét*!

Tétel: *Jelölje a* $p(x)=a_0x^n+a_1x^{n-1}+...+a_{n-1}x+a_n$ *polinom gyökeit* $x_1,x_2,...,x_n$.
Ekkor $\mu=\max\limits_{1\le i\le n}|x_i|\le\max\left\{1+\left|\frac{a_1}{a_0}\right|,1+\left|\frac{a_2}{a_0}\right|,...,1+\left|\frac{a_{n-1}}{a_0}\right|,\left|\frac{a_n}{a_0}\right|\right\}=\delta$.

Bizonyítás: Minden gyökre igaz, hogy $x_i^n=-\frac{a_1}{a_0}x_i^{n-1}-\frac{a_2}{a_0}x_i^{n-2}-...-\frac{a_{n-1}}{a_0}x_i-\frac{a_n}{a_0}$,

így igaz a $\mu^n\le\left|\frac{a_1}{a_0}\right|\mu^{n-1}+\left|\frac{a_2}{a_0}\right|\mu^{n-2}+...+\left|\frac{a_{n-1}}{a_0}\right|\mu+\left|\frac{a_n}{a_0}\right|$.

Tudjuk, hogy $\left|\frac{a_i}{a_0}\right| \le \delta - 1$ (i=1,…,n-1) és $\left|\frac{a_n}{a_0}\right| \le \delta$.

Tegyük fel, hogy $\mu > 1$, ellenkező esetben a bizonyítás triviális.

Felhasználva az előbbi eredményeket, így

$$\mu^n \le (\delta-1)\mu^{n-1} + (\delta-1)\mu^{n-2} + \ldots + (\delta-1)\mu + (\delta-1) + 1$$

$\mu^n \le (\delta-1)\frac{\mu^n - 1}{\mu - 1} + 1$**,** amiből már könnyen adódik az állítás.

Példa: Az előbbi példabeli polinomra a tétel azt adja, hogy minden gyök benne van egy origó körüli max(2,3,5,8)=8 sugarú körben.

Ezek a tételek arra is használhatók, hogy ha egy polinom zérushelyeinek meghatározására iterációs módszert használunk, akkor a sorozat kezdőelemének kiválasztásában segítségünkre lehetnek az olyan tartományok, amelyekről biztosan tudjuk, hogy a polinomnak van zérushelye.

6. téma
Függvényközelítések, Lagrange interpoláció, legkisebb négyzetek módszere

Ebben a témakörben a *polinom-interpolációs feladattal* foglalkozunk.

Probléma: Adott n darab páronként különböző $\{x_1, x_2, ..., x_n\}$ *ún. interpolációs alappont és az* $\{f_1, f_2, ..., f_n\}$ *értékek. Határozzuk meg azt a legfeljebb n-1-ed fokú* $P_n(x)$ *interpolációs polinomot, amely eleget tesz a* $P_n(x_i) = f_i$ *interpolációs feltételnek.*

Lagrange interpoláció

Az előbbi feladat egyértelműen megoldható.

Definíció: *Minden* $1 \le i \le n$*-re definiáljuk az*

$$L_i(x) = \frac{(x - x_1)(x - x_2)\ldots(x - x_{i-1})(x - x_{i+1})\ldots(x - x_n)}{(x_i - x_1)\ldots(x_i - x_{i-1})(x_i - x_{i+1})\ldots(x_i - x_n)}$$

ún. Lagrange-féle bázispolinomokat.

Tétel:
(i) *Minden* L_i *pontosan n-1-ed fokú polinom.*
(ii) *Az alappontokban felvett értékekre* $L_i(x_k) = \delta_{ik}$ *teljesül.*

Bizonyítás:
(i) A nevező konstans, míg a számlálóban egy *n-1*-ed fokú polinom van.
(ii) Ha $i=k$, akkor $L_i(x_i) = 1$, különben 0.

Tétel. *Az interpolációs probléma megoldható és egy lehetséges megoldása a*

$$P_n(x) = \sum_{i=1}^{n} f_i L_i(x)$$

Lagrange-féle interpolációs polinom.

Bizonyítás. Az előbbi állítás (i)-ből következik, hogy $P_n(x)$ legfeljebb *n-1*-ed fokú polinom. Az előbbi állítás (ii)-ből következik, hogy $P_n(x_i) = f_i$.

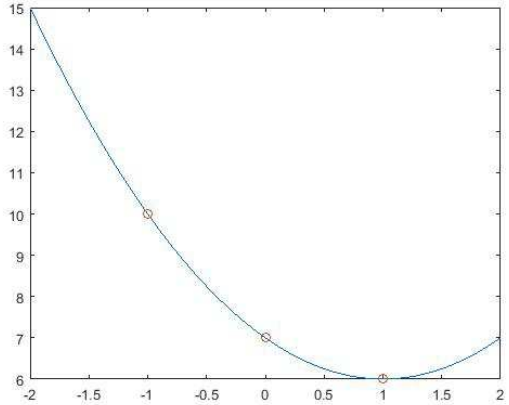
Tétel. *Tetszőlegesen választott alappontok és függvényértékek esetén az interpolációs probléma megoldása egyértelmű.*

Bizonyítás. Tegyük fel indirekt módon, hogy két különböző megoldás létezik *P* és *P'*. Képezve az *S=P-P'* polinomot világos, hogy *S*-nek minden alappontban 0 az értéke. Ez azt jelenti, hogy legalább *n* zérushelye van, ami azt jelenti, hogy legalább *n* a fokszáma. Ez azonban ellentmond S konstrukciójának, mivel abból meg az következik, hogy *S* legfeljebb *n-1*-ed fokú.

Példa. Legyenek adva a (-1,10), (0,7) és (1,6) pontok. Keressük meg hozzájuk a Lagrange-féle interpolációs polinomot.

$$L_1(x) = \frac{(x-0)(x-1)}{(-1-0)(-1-1)} \qquad L_2(x) = \frac{(x+1)(x-1)}{(0+1)(0-1)} \qquad L_3(x) = \frac{(x+1)(x-0)}{(1+1)(1-0)}$$

$$P_3(x) = 10\frac{(x-0)(x-1)}{(-1-0)(-1-1)} + 7\frac{(x+1)(x-1)}{(0+1)(0-1)} + 6\frac{(x+1)(x-0)}{(1+1)(1-0)} = x^2 - 2x + 7$$



Legkisebb négyzetek módszere

Adott *m* pontpár (x_i, y_i) *(i=1,…,m)*. Szeretnénk ezeket a pontokat a Descartes-féle koordinátarendszerben egy *f* függvény grafikonjával jól közelíteni. Az *f* függvény itt *n* számú lineárisan független g_j függvény lineáris kombinációjának kell lennie. Tehát

$$f = \sum_{j=1}^{n} \alpha_j g_j \text{ , ahol } \alpha_j \in R .$$

Feladatunk így az α_j értékek alkalmas megválasztása. A legkisebb négyzetek módszerének lényege, úgy megválasztani az α_j-ket, hogy

$$\sqrt{\sum_{i=1}^{m}\left(\sum_{j=1}^{n}\alpha_j g_j(x_i) - y_i\right)^2} \text{ minimális legyen.}$$

Ehhez az szükséges, hogy minden $1 \le i \le n$-re

$$\frac{\partial}{\partial \alpha_i}\sum_{i=1}^{m}\left(\sum_{j=1}^{n}\alpha_j g_j(x_i) - y_i\right)^2 = 0$$

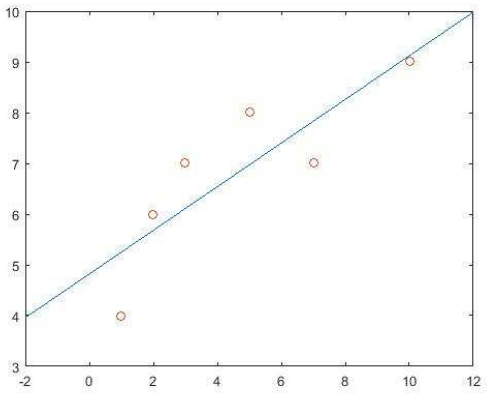
Ez utóbbi egy lineáris egyenletrendszer az α_j-kre, amit megoldva felírhatjuk a keresett *f* függvényt. A legegyszerűbb eset az *n=2* , $g_1(x)=1$ és $g_2(x)=x$, tehát amikor lineáris függvénnyel közelítünk. Ekkor (vizsgára nem kell tudni fejből ezeket a képleteket)

$$\alpha_1 = \frac{\sum_{i=1}^{m} x_i^2 \sum_{i=1}^{m} y_i - \sum_{i=1}^{m} x_i \sum_{i=1}^{m} x_i y_i}{m\sum_{i=1}^{m} x_i^2 - \left(\sum_{i=1}^{m} x_i\right)^2} \qquad \alpha_2 = \frac{m\sum_{i=1}^{m} x_i y_i - \sum_{i=1}^{m} x_i \sum_{i=1}^{m} y_i}{m\sum_{i=1}^{m} x_i^2 - \left(\sum_{i=1}^{m} x_i\right)^2}$$

Példa: Legyenek adva az (1,4), (2,6), (3,7), (5,8), (7,7) és (10,9) pontok. A legkisebb négyzetek módszere az előbbi speciális függvényválasztással egy egyenessel, az ún. regressziós egyenessel közelít. Itt

$$\alpha_1 = \frac{415}{86} \approx 4.82 \qquad \alpha_2 = \frac{37}{86} \approx 0.43$$

Az egyenes egyenlete így y=4.82+0.43x.

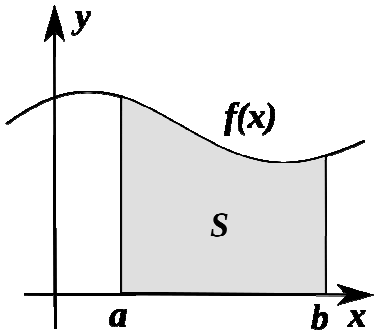


7. téma
Numerikus integrálás, interpolációs kvadratúraformulák, Newton-Cotes formulák

A klasszikus analízisben egy határozott Riemann-integrált a Newton-Leibniz szabállyal szoktunk kiszámolni, a jól ismert alapképlet segítségével:

$$\int_a^b f(x)dx = F(b) - F(a)$$

ahol F az f ún. primitív függvénye.



Előfordulhat azonban olyan eset is, amikor F meghatározása nagyon nehéz feladat, vagy éppen F nem is írható fel zárt alakban. Továbbra is az analízis eszköztárán belül maradva vannak olyan „trükkök” amelyeket néha sikerrel alkalmazhatunk ilyen esetekben is:

- az integrálandó függvény sorba fejtése, majd a sor tagonkénti integrálása,
- parciális integrálás,
- helyettesítéses integrálás, stb.

A gyakorlati életben azonban előfordulhatnak olyan feladatok is, hogy nemcsak az F, de maga az integrálandó függvény képlete sincs meghatározva, hanem helyette pl. egy gépi szubrutin van csak adva. Világos, hogy ilyenkor az analízis előbbi eszközei helyett más eljárásokat kell kitalálni. Fontos szerepet játszik az is, hogy a gyakorlat számára általában nincs szükség a pontos értékre elég annak egy nagyon jó közelítése is.

Definíció: *Tegyük fel, hogy létezik az integrálja az f-nek az [a,b] intervallumon. A* $Q_n(f) = \sum_{i=1}^{n} w_i f(x_i)$ *súlyozott összeget kvadratúra-formulának hívjuk. A benne szereplő* w_i *számokat a formula súlyainak, az* x_i *helyeket kvadratúra-alappontoknak hívjuk.*

A kvadratúra-alappontokról mindig feltesszük, hogy az [a,b] intervallumban vannak és páronként különbözőek.

Interpolációs kvadratúra-formulák

Alapötlet: Közelítsük az *f* függvényt polinommal (hiszen azok jól integrálhatók), így az *f* integrálját is polinom integráljával.

Vegyünk fel *n* alappontot az [a,b]-ben és közelítsünk az ezekhez tartozó $P_n(x)$ Lagrange-féle interpolációs polinommal. Az integrál ekkor az alábbi módon alakul:

$$\int_a^b f(x)dx \approx \int_a^b P_n(x)dx = \int_a^b \sum_{i=1}^{n} f(x_i)L_i(x)dx = \sum_{i=1}^{n} f(x_i)\int_a^b L_i(x)dx$$

Ha most bevezetjük az $w_i = \int_a^b L_i(x)dx$ súlyokat, akkor kiderül, hogy tényleg egy kvadratúra formulát kaptunk.

Definíció: Egy $Q_n(f) = \sum_{i=1}^{n} w_i f(x_i)$ *kvadratúra formula <u>interpolációs kvadratúra-formula</u>, ha megkapható a fenti módon, vagyis az alappontjaira felírt Lagrange-interpolációs polinom kiintegrálásával.*

Newton-Cotes-formulák

A Newton-Cotes-formulák speciális interpolációs kvadratúra-formulák, méghozzá azt a speciális esetet jelentik, amikor *n* darab *ekvidisztáns* alapponthoz tartozó Lagrange-polinom integráljával közelítünk.

Legyen a kiválasztott *n* alappont $a \le x_0 < x_1 < ... < x_{n-2} < x_{n-1} \le b$.

Attól függően, hogy az integrálás határai szerepelnek-e az interpolációs alappontok között vagy sem, két esetet különböztetünk meg:

A *nyitott formulák* esetén a határok nem alappontok, ekkor

$$h = \frac{b-a}{n+1}, \qquad a = x_0 - h, \qquad b = x_{n-1} + h \qquad \text{és} \qquad x_i = x_0 + ih \qquad 0 \le i \le n-1$$

A *zárt formulák* esetében a határok is alappontok, ekkor

$$h = \frac{b-a}{n-1}, \qquad a = x_0, \qquad b = x_{n-1} \qquad \text{és} \quad x_i = x_0 + ih \qquad 0 \le i \le n-1.$$

Mielőtt definiálnánk pontosan a Newton-Cotes formulákat, szólnunk kell a véges differenciákról is.

Definíció: *Adott* x_k *alappontokhoz és* f_k *függvényértékekhez tartozó* $\Delta^i f_k$ *i-ed rendű véges differenciákat a következő rekurzióval értelmezzük:*

(i) $\Delta^0 f_k = f_k$

(ii) $\Delta^i f_k = \Delta^{i-1} f_{k+1} - \Delta^{i-1} f_k$

Igazolható, hogy $\Delta^i f_k = \sum_{j=0}^{i} (-1)^j \binom{i}{j} f_{i+k-j}$

f_0

Δf_0

f_1 $\Delta^2 f_0$

Δf_1

f_2

Tétel (bizonyítás nélkül): *A Lagrange-féle interpolációs polinomot véges differenciákkal az alábbi módon írható fel:*

$$P_n(x) = P_n(x_0 + th) = \sum_{i=0}^{n-1} \binom{t}{i} \Delta^i f_0$$

$$\int_a^b P_n(x_0 + th)dx = \int_a^b \sum_{i=0}^{n-1} \binom{t}{i} \Delta^i f_0 dx = \sum_{i=0}^{n-1} \Delta^i f_0 \int_a^b \binom{t}{i} dx$$

Ha *t* szerinti integrálásra térünk át, akkor két különböző integrált kell felírni:

(i) nyitott formulák esetén az $x = x_0 + th$ helyettesítés után

$$\sum_{i=0}^{n-1} \Delta^i f_0 \int_a^b \binom{t}{i} dx = h \sum_{i=0}^{n-1} \Delta^i f_0 \int_{-1}^{n} \binom{t}{i} dt$$

(ii) zárt formulák esetén szintén az $x = x_0 + th$ helyettesítést elvégezve

$$\sum_{i=0}^{n-1} \Delta^i f_0 \int_a^b \binom{t}{i} dx = h \sum_{i=0}^{n-1} \Delta^i f_0 \int_{0}^{n-1} \binom{t}{i} dt$$

Néhány speciális formula

Nyitott formula

n=1 (érintő-formula)

$$h = \frac{b-a}{2} \qquad a < x_0 = \frac{a+b}{2} < b$$

$$\int_a^b P_1(x_0 + th)dx = h \sum_{i=0}^{0} \Delta^i f_0 \int_{-1}^{1} \binom{t}{i} dt = \frac{b-a}{2} f(x_0) \int_{-1}^{1} 1 dt = \frac{b-a}{2} f\left(\frac{a+b}{2}\right) 2 = (b-a) f\left(\frac{a+b}{2}\right)$$

Zárt formula

n=2 (trapéz-formula)

$h = b - a \qquad a = x_0 < x_1 = b$

$$\int_a^b P_2(x_0 + th)dx = h\sum_{i=0}^{1} \Delta^i f_0 \int_0^1 \binom{t}{i} dt = (b-a)\left(f(x_0)\int_0^1 1dt + \Delta f_0 \int_0^1 \binom{t}{1} dt \right) =$$

$$= (b-a)\left(f(x_0) + \frac{1}{2}\Delta f_0 \right) = (b-a)\left(\frac{1}{2} f(a) + \frac{1}{2} f(b) \right).$$