1. BÖLÜM
HATA ANALİZİ

GİRİŞ

Nümerik Analiz (diğer tabirle Sayısal yöntemler), matematiksel modellemeler (cebirsel denklem, trigonometri, logaritma, limit, türev, integral, lineer cebir v.s.) sayıları kullanarak yaklaşık olarak çözme tekniklerinin adıdır. Nümerik Analizin amacı, bu modellemeleri çeşitli yaklaşım metotlarıyla sayı değerleri (nümerik değerleri) elde etmeye çalışmadır. Böylece başta bilgisayar olmak üzere pek çok elektronik eşyaya sayıların programlarının yapılmasını sağlar. Mesela $\sqrt{2}$ sayısının değerini bir hesap makinesi algoritması nümerik analizde anlatılan yöntemlerle yapılması daha kolay olmaktadır.

NÜMERİK HATA KAVRAMI

Nümerik analizde bir takım yaklaşım metotları verilecektir. Bu yaklaşım metotların verdiği sonuçlar sayılar ile ifade edildiğinden gerçek sonuçlar ile karşılaştırıldığında hatalar barındırır. Çözümlerin kullanılabilirliği için bu hataların büyüklüğü önemlidir. Hataların büyüklüğünü iyi analiz edebilmek için hata kavramını sistematik bir şekilde ele alacağız.

1.1. Tanım: Yaklaşık bir değerin gerçek değer ile arasındaki farkına nümerik hata veya gerçek hata denir. Matematiksel model oluştururken yapılan basitleştirici kabullerin yol açtığı hatalar da bu tanıma ilave edilebilir.

1.1. Aksiyom: Gerçek değer p, yaklaşık değer p_1, gerçek hata E_t ile gösterilmek üzere;

$$E_t = |p - p_1|$$

şeklindedir. Burada E (eror) hatayı t (true) ise bu hatanın gerçek değer kullanarak hesaplandığını gösterir.

Örnek: Bir öğrenci 123 m^2 lik alanın ölçümünü yapan bir stajyer öğrenci 122 m^2 olarak elde ediyorsa

$$E_t = |p - p_1| = |123 - 122| = 1 \text{ br}$$

lik bir hata tespit etmiştir.

1.2. Tanım: Bir hatayı yüzdelik olarak elde etmeye bağı hata denir.

1.2. Aksiyom: Gerçek değer p, Yaklaşık değer p_1, Bağıl hata ε_t ile gösterilmek üzere;

$$\varepsilon_t = \left|\frac{p-p_1}{p}\right|, \qquad (p \neq 0)$$

şeklindedir.

Örnek: Bir sınavda 35 soru vardır. Bu soruların 24 sini doğru yapan adayın bağıl hatasını bulunuz.

Çözüm: $\varepsilon_t = \left|\frac{35-24}{35}\right| = 0{,}31$ yani %31 dir.

Örnek: 80 cm olarak yapılması gereken bir duvarın uzunluğunu inşaat mühendisinin ölçülmesi sonucu 78 cm olduğu tespit edilmiştir. Bu verilerdeki gerçek hatayı ve bağıl hatayı bulunuz.

Çözüm: Duvardaki gerçek hata

$$E_t = 80 - 78 = 2\,\text{cm}$$

Bağıl hata ise

$$\varepsilon_t = \frac{80-78}{80} = 0{,}025$$

%2,5

dir.

İTERASYON

Matematiksel bir denklemi veya matematiksel bir modellemeden sonuçlar elde edebilmek için aynı denklem veya modellemeyi birden fazla uygulamaya iterasyon adı verilir. İterasyondan amaç, artarda değerler verilerek daha sağlıklı yaklaşımlar bulmaktır. Bu gibi durumlar için, hata genellikle o andaki yaklaşık değer ile bir önceki yaklaşık değer arasındaki fark olarak tahmin edilir.

1.3. Tanım: İlk terim herhangi bir reel değer olmak üzere, tanım kümesi bir önceki görüntüden oluşan dizilere iterasyon denir. Yani (a_n) herhangi bir dizi ise iterasyonun genel terimi

$$s_n = a_{n\circ(n-1)\circ\ldots\circ 3\circ 2\circ 1}$$

biçimindeki dizilerdir. O halde iterasyon terimlerin bileşkelerinin oluşturduğu özel bir dizidir.

Örnek: $a_1 = 3$ ve genel terimi $a_n = 2n$ olan bir dizinin iterasyonun 5. Terimi (adımını) bulunuz.

Çözüm: $s_1 = a_1 = 3$

$s_2 = a_2 \circ a_1 = (2n) \circ (3) = 6$

$s_3 = a_3 \circ a_2 \circ a_1 = (2n) \circ (2n) \circ (3) = 12$

$s_4 = a_4 \circ a_3 \circ a_2 \circ a_1 = (2n) \circ (2n) \circ (2n) \circ (3) = 36$

$s_5 = a_5 \circ a_4 \circ a_3 \circ a_2 \circ a_1 = (2n) \circ (2n) \circ (2n) \circ (2n) \circ (3) = 108$

1.2. Aksiyom: Mevcut yaklaşık değer p_i, önceki yaklaşık değer p_{i-1}, yaklaşık bağıl hata ε_a ile gösterilmek üzere;

$$\varepsilon_a = \left| \frac{p_i - p_{i-1}}{p_i} \right|, \qquad (p_i \neq 0)$$

şeklindedir.

Hata analizi yapıldığında hatanın pozitif ya da negatif çıkması önemli değildir. Daha çok hatanın tolere edilebilecek miktarda olması önemlidir. Tolere edilebilecek hata yaklaşık bağıl hatadan mutlak değerce küçük olmalıdır. Yani $\varepsilon_a < \varepsilon_t$ olmalıdır. Bu durum söz konusu oluncaya kadar hesaplamaya devam edilir.

Örnek: $e^{0,5} = 1,6487...$ olduğunu bilinmektedir.

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + ... + \frac{x^n}{n!} + ...$$

Taylor polinomu kullanarak $\varepsilon_a < \varepsilon_s$ oluncaya kadar iterasyon uygulayınız.

Çözüm:

1. adım: $e^x \approx 1$ için $e^{0,5} \approx 1$ olup

$$\varepsilon_t = \left| \frac{p - p_1}{p_1} \right| = \left| \frac{1,6487 - 1}{1,6487} \right| = 0,3935$$

%39,35 bağıl hata vardır.

2. adım: $e^x \approx 1 + x$ için $e^{0,5} \approx 1 + 0,5 = 1,5$ olup

$$\varepsilon_t = \left| \frac{1,6487 - 1,5}{1,6487} \right| = 0,0902$$

%9,02 bağıl hata vardır. Buna göre

$$\varepsilon_a = \left| \frac{p_i - p_{i-1}}{p_i} \right| = \left| \frac{1,5 - 1}{1,5} \right| = 0,3333$$

% 33,33 yaklaşık bağıl hata vardır.

3. adım: $e^{x} \approx 1 + x + \frac{x^2}{2!}$ için $e^{0,5} \approx 1 + 0,5 + \frac{0,5^2}{2!} = 1,625$ olup

$$\varepsilon_t = \left|\frac{1,6487 - 1,625}{1,6487}\right| = 0,0144$$

%1,44 bağıl hata vardır. Buna göre

$$\varepsilon_a = \left|\frac{p_i - p_{i-1}}{p_i}\right| = \left|\frac{1,625 - 1,5}{1,625}\right| = 0,0769$$

% 7,69 yaklaşık bağıl hata vardır.

4. adım: $e^{x} \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$ için $e^{0,5} \approx 1 + 0,5 + \frac{0,5^2}{2!} + \frac{0,5^3}{3!} = 1,6458$ olup

$$\varepsilon_t = \left|\frac{1,6487 - 1,6458}{1,6487}\right| = 0,0018$$

%0,18 bağıl hata vardır. Buna göre

$$\varepsilon_a = \left|\frac{p_i - p_{i-1}}{p_i}\right| = \left|\frac{1,6458 - 1,625}{1,6458}\right| = 0,0126$$

% 1,26 yaklaşık bağıl hata vardır.

5. adım: $e^{x} \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$ için

$e^{0,5} \approx 1 + 0,5 + \frac{0,5^2}{2!} + \frac{0,5^3}{3!} + \frac{0,5^4}{4!} = 1,6484$ olup

$$\varepsilon_t = \left|\frac{1,6487 - 1,6484}{1,6487}\right| = 0,00018$$

%0,018 bağıl hata vardır. Buna göre

$$\varepsilon_a = \left|\frac{p_i - p_{i-1}}{p_i}\right| = \left|\frac{1,6484 - 1,6458}{1,6484}\right| = 0,00157$$

% 0,157 yaklaşık bağıl hata vardır.

6. adım: $e^{x} \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!}$ için

$e^{0,5} \approx 1 + 0,5 + \frac{0,5^2}{2!} + \frac{0,5^3}{3!} + \frac{0,5^4}{4!} + + \frac{x^5}{5!} = 1,6485$ olup

$$\varepsilon_t = \left|\frac{1,6487 - 1,6485}{1,6487}\right| = 0,00012$$

%0,012 bağıl hata vardır. Buna göre

$$\varepsilon_a = \left|\frac{p_i - p_{i-1}}{p_i}\right| = \left|\frac{1,6485-1,6484}{1,6485}\right| = 0,00006$$

% 0,006 yaklaşık bağıl hata vardır.

6. adımda yaklaşık bağıl hata %0,006, bağıl hatadan %0,012 den küçük kalır. Buna göre şu tablo çizilebilir.

Adımlar	Sonuç	ε_t	ε_a
1.	1	%39,35	-
2.	1,5	%9,02	%33,333
3.	1,625	%1,44	%7,69
4.	1,6458	%0,18	%1,26
5.	1,6484	%0,018	%0,157
6.	1,6485	%0,012	%0,006

HATAYA YOL AÇAN DURUMLAR

1. Yuvarlama Hataları

π, e, $\sqrt{2}$ gibi irrasyonel sayıların virgülden sonraki terimleri sonsuza gider ya da sabit sayıda anlamlı basamak yazılamaz. Bundan dolayı bilgisayarda tam olarak ifade edilemezler. Ayrıca bilgisayarlar binary (2 tabanlı) gösterim kullandıklarından bazı decimal (10 tabanlı) sayıları da hassas olarak ifade edemezler. Anlamlı basamakların bu şekilde ifade edilmesinden dolayı oluşan farka yuvarlama hatası denir.

2. Kesme Hataları

Sayısal işlemlerde kullanılan sonsuz seriler, belli bir yerde kesilip geri kalan terimlerinin atılması biçiminde kullanılır.

Örnek:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + ... + \frac{x^n}{n!} + ...$$

serisinden $e^{0,5}$ değerini elde etmek istediğimizde sonsuz seriyi alamayacağımızdan belli sayıda terim alınıp geri kalanı atmamız gerekir. Sayısal yöntemlerde bu tür hatalara kesme hatası denir.

Örnek: $\int_0^{0,5} \cos x \, dx$ integralini hem gerçek değerini bulunuz, hem de Taylor polinomu (Maclaurin serisi) 4 terim alarak elde ediniz. İki işlem arasındaki hata miktarını bulunuz.

Çözüm: Önce gerçek değerini bulalım.

$$\int_0^{0,5} \cos x \, dx = \sin x \Big|_0^{0,5} = \sin 0{,}5 = 0{,}008727$$

Şimdi Taylor polinomunu kullanalım

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \ldots$$

$$\int_0^{0,5} \cos x \, dx = \int_0^{0,5} \left(1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} \right) dx + K_n(x)$$

$$= \left(x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{1008} \right) \Bigg|_0^{0,5} + K_n(x)$$

$$= 0{,}5 - \frac{0{,}5^3}{6} + \frac{0{,}5^5}{120} - \frac{0{,}5^7}{1008} + K_n(x)$$

$$= 0{,}479419 + K_n(x)$$

bulunur. Buna göre hata miktarı

$$K_n(x) = |0{,}479419 - 0{,}008727| = 0{,}470692$$

olur.

3. Veri ve Formülasyon Hataları

Bu durumlar matematiksel modellemeyle ilgilidir. Modelin dayandığı fiziksel verilerin belirsizliği nedeniyle analizde bazen hatalar oluşur. Örneğin; düşen bir cisim modelinin denemek için bir cismi defalarca belli bir yükseklikten atarak belirli bir zaman sonra düşme hızını ölçtüğümüzde, bu ölçümlerde mutlaka belirsizlik olacaktır, çünkü cismin bazı düşmelerinde diğerlerine göre daha hızlı olacaktır. Bu gibi durumlara veri ve formülasyon Hataları denir.

BİLGİSAYAR ARİTMETİĞİ

Binary (İki Tabanlı) Makine Sayıları

Bilgisayarda tamsayılar ikili sayı sistemi kullanılarak ifade edilirler. Burada önemli olan bir nokta, sayıların ifade edilmesinde kullanılan bit sayısının, gösterilebilecek maksimum tamsayı değerini kısıtlamasıdır. Örneğin 32 bit kullanılarak yazılabilecek en büyük sayı $2^{32} = 4294967296$ olmaktadır. Eğer bu 32 bitin bir tanesini işaret için kullanarak sayıyı ifade etmeye kalkarsak $2^{32-1} = 2147483648$ değeri elde edilecektir. Eğer tamsayı gösteriminde işaret biti kullanılıyorsa işaretli tamsayı, kullanılmıyorsa işaretsiz tamsayı adını alır. Tamsayı gösteriminde limitler aşılırsa taşma problemi ile karşılaşılır.

Reel sayıların gösterimi için sayı sistemleri kullanılabilir olsa da bilgisayarda çok küçük ve çok büyük reel sayıların gösteriminde kayan nokta sayıları kullanılır. Kayan nokta sayıları, bir sayı ile o sayının 10'un bir kuvveti ile çarpımı şeklinde kullanılan bilimsel notasyona çok benzer. Günümüzde kayan nokta sayılarının gösteriminde yaygın olarak kullanılan yöntem IEEE 754 kayan nokta aritmetik standardıdır.

64 bit (ikili dijit) bir reel sayıyı göstermek için kullanılsın. s ile gösterilen ilk bit sayının işaretini ifade eder ve 1 ya da 0 değerini alır. Burada 1 sayının negatif, 0 ise pozitif olduğunu gösterir. s'yi 2 tabanında yazılan, c ile gösterilen ve karakteristik adı verilen 11 bitlik üst takip eder. c'nin ardından ise ½ tabanında yazılan, f ile gösterilen ve mantis adı verilen 52 bitlik kısım gelir. Bu sistem kullanarak

$$(-1)^s 2^{c-1023}(1+f)$$

formunda kayan nokta sayıları elde edilir.

Kayan nokta gösterimi, hem kesirlerin hem de çok büyük sayıların bilgisayarda gösterilmesine olanak tanır. Ancak bazı sakıncaları vardır. Örneğin, kayan nokta sayıları tamsayılara göre daha fazla yer tutar ve işlemler daha uzun sürer. Daha da önemlisi, mantis sadece sonlu sayıda anlamlı basamak tutabildiğinden, bir hata oluşturur. Böylece yuvarlama hatası ortaya çıkar.

Örnek: Aşağıdaki makine sayısı göz önüne alınsın:

0 10000000011 1011100100010000000000000000000000000000000000000000
00.

En soldaki bit $s = 0$ olduğundan verilen sayının pozitif olduğu sonucu elde edilir. Sonraki 11 bit 10000000011, sayının karakteristiğidir ve ondalık olarak

$$c = 1.2^{10} + 0.2^9 + 0.2^8 + 0.2^7 + 0.2^6 + 0.2^5 + 0.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 1.2^1 = 1027$$

sayısına eşittir. Dolayısıyla sayının üst kısmı $2^{1027-1023} = 2^4 = 16$'dır. Son olarak 52 bitlik mantis kısmı

$$f = 1.\left(\frac{1}{2}\right)^1 + 1.\left(\frac{1}{2}\right)^3 + 1.\left(\frac{1}{2}\right)^4 + 1.\left(\frac{1}{2}\right)^5 + 1.\left(\frac{1}{2}\right)^8 + 1.\left(\frac{1}{2}\right)^{12}$$

şeklinde yazılır. Sonuç olarak verilen makine sayısı tam olarak

$$(-1)^s 2^{c-1023}(1+f)=(-1)^0 2^{1027-1023}\left(1+\left(\frac{1}{2}+\frac{1}{8}+\frac{1}{16}+\frac{1}{32}+\frac{1}{256}+\frac{1}{4096}\right)\right)$$
$$=27{,}56640625$$
ondalık sayısına eşittir. Diğer taraftan yukarıda verilen makine sayısından bir küçük sayı

0 10000000011 1011100100001111111111111111111111111111111111111
111

ve bir büyük sayı

0 10000000011 1011100100010000000000000000000000000000000000000
001

dir. Buna göre bizim orijinal makine sayımız sadece 27,56640625 değerini göstermekle kalmayıp yukarıda verilen kendisinden bir büyük ve bir küçük sayının işaret etmiş olduğu sayılarla sınırlı aralıkta yer alan tüm sayıları göstermektedir.
Bu formatta ifade edilebilecek

(i) en küçük pozitif sayı: s = 0, c = 1 ve f = 0 için
$$2^{-1022}(1+0)\approx 0{,}22251.10^{-307},$$

(ii) en büyük sayı: s = 0, c = 2046 ve $f=1-2^{-52}$ için
$$2^{1023}(2-2^{-52})\approx 0{,}17977.10^{309}$$
olduğundan eğer hesaplamalar sonucu elde edilen sayı $2^{-1022}(1+0)$'den büyük ise aşağı-taşma, $2^{1023}(2-2^{-52})$ sayısından büyük ise yukarı-taşma yaşanacağından hesaplamanın belirli bir hassaslık ile yapılmaması durumunda prosedür kesintiye uğrayacaktır. Ayrıca, sıfır sayısı için iki türlü gösterilim vardır: pozitif sıfır için s = 0, c = 0 ve f = 0; negatif sıfır için s = 1, c = 0 ve f = 0.

Decimal (Ondalık) Makine Sayıları

Bu kısımda, her i = 2, · · · , k için $1\le d_1\le 9$ ve $0\le d_i\le 9$ olmak üzere
$$\pm 0{,}d_1d_2\ldots d_k.10^2$$
olarak ifade edilen k-dijit normalize edilmiş ondalık kayan-nokta formundaki sayılar ile ilgileneceğiz.

Nümerik olarak makinede bir taşmaya neden olmayan, normalize edilmiş Pozitif
$$y=0{,}d_1d_2\ldots d_{k+1}d_{k+2}.10^n$$
sayısı göz önüne alınsın. y sayısının kayan nokta formu $f\ell(y)$ ile gösterilir ve ondalık kısmının sadece k-dijit olarak alınması ile elde edilir. Bu k-dijitin belirlen-

mesinde genel olarak iki yöntem takip edilir. Bunlardan ilki kesme denen ve $d_{k+1}d_{k+2}\ldots$ ondalıklarının atılması ile elde edilen

$$f\ell(y)=0,d_1d_2\ldots d_{k+1}d_{k+2}.10^n$$

sayıdır. Bir diğeri ise yuvarlama adı verilen ve $5.10^{n-(k+1)}$ sayısının y'ye eklenmesi sonrasında kesme yapılması ile elde edilen

$$f\ell(y)=0,\delta_1\delta_2\ldots\delta_k.10^n$$

sayısıdır. Kısaca yuvarlama yaparken $d_{k+1}\geq 5$ ise $f\ell(y)$'yi elde etmek için d_k sayısına 1 eklenirken (yukarı yuvarlama), $d_{k+1}\geq 5$ ise k-dijitten sonra kesme yapılır (aşağı yuvarlama). Eğer aşağı yuvarlama yapılıyor ise her i = 1, 2, · · · , k için $\delta_i=d_i$ olur. Bununla beraber, yukarı yuvarlama yapılması durumda tam kısım da dahil olmak üzere tüm ondalık basamaklar değişebilir.

Sayı	Normalize Edilmiş Ondalık Form
1222000	0.1222×10^{7}
-0.0000345	-0.345×10^{-4}
0.0027×10^{-3}	0.27×10^{-5}
-3687.487×10^{8}	-0.3687487×10^{12}
0.1001×10^{-5}	0.1001×10^{-5}
9.26539874×10^{-21}	$0.926539874\times10^{-20}$

Örnek: π sayısının beş-dijit
a) kesme değerlerini
b) yuvarlama değerlerini
elde ediniz.

Çözüm: π = 3.141592654 · · · şeklinde kendini tekrar etmeyen sonsuz ondalığa sahip π sayısı normalize edilerek

$$\pi=(0,3141592654\ldots).10^1$$

olarak yazılsın. Buna göre

a) beş-dijit kesme yapılarak π sayısının kayan-nokta formu

$$f\ell(\pi)=0,31415.10^1=3,14159$$

b) sayının altıncı basamağı 9 olduğundan beş-dijit yuvarlama yapılarak π sayısının kayan-nokta formu

$$f\ell(\pi)=(0,314159+0,00001).10^1=3,14159$$

olarak elde edilir.

Örnek: Aşağıda verilen p değerlerine yapılan p_1 yaklaşımlarında oluşan mutlak ve bağıl hataları hesaplayınız.

a) $p = 0{,}3000.10^{1}$, $p_1 = 0{,}3100.10^{1}$

b) $p = 0{,}3000.10^{-3}$, $p_1 = 0{,}2900.10^{-3}$

Çözüm:

a) Mutlak hata: $|p - p_1| = \left|0{,}3000.10^{1} - 0{,}3100.10^{1}\right| = 0{,}1$

Bağıl Hata: $\left|\dfrac{p-p_1}{p}\right| = \left|\dfrac{0{,}3000.10^{1} - 0{,}3100.10^{1}}{0{,}3000.10^{1}}\right| = 0{,}333333.10^{-1}$

b) Mutlak hata: $|p - p_1| = \left|0{,}3000.10^{-3} - 0{,}2900.10^{-3}\right| = 0{,}1.10^{-4}$

Bağıl Hata: $\left|\dfrac{p-p_1}{p}\right| = \left|\dfrac{0{,}3000.10^{-3} - 0{,}2900.10^{-3}}{0{,}3000.10^{-3}}\right| = 0{,}333333.10^{-1}$

Yukarıdaki örnekte görüldüğü üzere tüm yaklaşımlarda bağıl hata $0{,}333333.10^{-1}$ iken mutlak hata farklı değerler almaktadır. Yapılan yaklaşımda gerçek değere ne oranda yaklaşıldığını ölçmek bakımından bağıl hata mutlak hataya göre daha anlamlıdır. Örneğin 100 km'lik bir yolu 101 km ve 1 km'lik yolu 2 km ölçmede oluşan mutlak hatalar 1 km iken bu ölçümlerin hangisinin daha kabul edilebilir bir hata oranına sahip olduğunu anlamak için bağıl hatalarına bakmak gerekir.

Örnek: a) $f(x) = \cos x - (x+1)^2$ fonksiyonunun x = 0 civarında üçüncü Taylor polinomunu hata terimini göz ardı ederek hesaplayınız.

b) Eğer yukarıda bulunan polinom 5-dijit yuvarlama artimetiği ile f(0,05) değerini hesaplamak için kullanılırsa oluşacak mutlak ve bağıl hatayı tespit ediniz.

Çözüm: a) Hata terimi yazılmayacağından üçüncü Taylor polinomunu hesaplamak için aşağıdaki işlemler yapılır:

$f(x) = \cos x - (x+1)^2$ ise $f(0) = \cos 0 - (0+1)^2 = 0$

$f'(x) = -\sin x - 2(x+1)$ ise $f'(0) = -\sin 0 - 2(0+1) = -2$

$f''(x) = -\cos x - 2$ ise $f''(0) = -\cos 0 - 2 = -3$

$f'''(x) = \sin x$ ise $f'''(0) = \sin 0 = 0$

Buna göre;

$$f_3(x) = f(0) + \frac{f'(0)}{1!}(x-0) + \frac{f''(0)}{2!}(x-0)^2 + \frac{f'''(0)}{3!}(x-0)^3$$

$$= 0 + \frac{-2}{1!}x + \frac{-3}{2!}x^2 + \frac{0}{3!}x^3$$

$$= -2x - \frac{3}{2}x^2$$

olarak üçüncü Taylor polinomunu hesaplanır.

b) 5-dijit yuvarlama artimetiği kullanılarak yapılan bu yaklaşımda

$$f_3(0{,}05) = -2(0{,}05.) - \frac{3}{2}(0{,}05)^2 = -0{,}10375$$

olarak elde edilir. Buna göre mutlak hata

$$\left|f(0{,}05) - f_3(0{,}05)\right| = \left|\cos 0{,}05 - (0{,}05+1)^2 - (-0{,}10375)\right| = 2{,}6039.10^{-7}$$

ve bağıl hata da

$$\left|\frac{f(0{,}05) - f_3(0{,}05)}{f(0{,}05)}\right| = \left|\frac{\cos 0{,}05 - (0{,}05+1)^2 - (-0{,}10375)}{\cos 0{,}05 - (0{,}05+1)^2}\right| = 2{,}5098.10^{-6}$$

şeklinde bulunur. Mutlak hatanın bağıl hatadan daha küçük çıkmasının nedeni yapılan yaklaşımın gerçek fonksiyon değerine çok yakın olmasıdır.

Tanım: $\left|\frac{p - p_1}{p}\right| \leq 5.10^{-t}$ eşitsizliğini sağlayan, negatif olmayan en büyük t pozitif tamsayısı için p_1, p'ye t anlamlı basamakta bir yaklaşımdır denir.

Verilen bir sayının anlamlı basamakları aşağıdaki şekilde tespit edilir:

- Tüm sıfırdan farklı sayılar anlamlı basamaktır. Mesela:
  5 sayısının bir anlamlı basamağı vardır.
  9,2 sayısının iki anlamlı basamağı vardır.
  470 sayısının iki anlamlı basamağı vardır.
  6 327,554 sayısının yedi anlamlı basamağı vardır.

- Tüm anlamlı basamaklar arasında kalan sıfırlar anlamlı basamaklardır. Mesela:
  703 sayısının üç anlamlı basamağı vardır.
  2,006 sayısının dört anlamlı basamağı vardır.
  60,003 sayısının beş anlamlı basamağı vardır.

- Ondalıklı bir sayıda ondalık noktasının sağında yer alan tüm sayılar anlamlı basamaklardır. Mesela:
  0,20 sayısının iki anlamlı basamağı vardır (ilk sıfır anlamlı değildir, fakat sondaki sıfır anlamlıdır).
  0,0050 sayısının iki anlamlı basamağı vardır (sondaki bir ve sıfır).
  5,80 sayısının üç anlamlı basamağı vardır.
  24,6000 sayısının altı anlamlı basamağı vardır.

100,00 sayısının beş anlamlı basamağı vardır (ondalık noktasının solunda yer alan iki sıfır anlamlı olduğu bilinen 4 sayısının sağında olduğundan anlamlıdır, ayrıca ondalık noktasının sağında yer alan sıfırlar 4 ile sıfır arasında yer aldığından anlamlıdır).

x sayısının k-dijit kayan nokta formu $f\ell(x)$ ile gösterilsin. $f\ell(x)$ sayısının yuvarlama aritmetiği kullanılarak elde edilmesi durumunda yaklaşımda oluşan bağıl hatanın bir üst sınırı

$$\left|\frac{x-f\ell(x)}{x}\right| \leq 0{,}5.10^{-k+1}$$

dir. Gerçekten: $d_1 \neq 0$, $n \in \mathbb{Z}$ olmak üzere x sayısı

$$x=(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n$$

şeklinde verilsin.

- $d_{k+1} < 5$ olsun. Bu durumda $f\ell(x)=(0{,}d_1 d_2 d_3 ... d_k ...)10^n$ olduğundan

$$\left|\frac{x-f\ell(x)}{x}\right| \leq 0{,}5.10^{-k+1}$$

$$\left|\frac{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n-(0{,}d_1 d_2 d_3 ... d_k)10^n}{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n}\right| \leq \frac{0{,}5.10^{-k}}{0{,}1}$$

$$\left|\frac{(0{,}d_{k+1} d_{k+2} d_{k+3} ...)10^{n-k}}{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n}\right| \leq 0{,}5.10^{-k+1}$$

bulunur.

- $d_{k+1} \geq 5$ olsun. Bu durumda $f\ell(x)=(0{,}d_1 d_2 d_3 ... d_k ...)10^n - 10^{n-k}$ olduğundan

$$\left|\frac{x-f\ell(x)}{x}\right| \leq 0{,}5.10^{-k+1}$$

$$\left|\frac{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n-(0{,}d_1 d_2 d_3 ... d_k)10^n}{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n}\right| \leq \frac{0{,}5.10^{-k}}{0{,}1}$$

$$\left|\frac{(1-0{,}d_{k+1} d_{k+2} d_{k+3} ...)10^{n-k}}{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n}\right| \leq 0{,}5.10^{-k+1}$$

elde edilir.

Diğer taraftan k-dijit kesme aritmetiği kullanılarak $f\ell(x)$ sayısının elde edilmesi durumunda oluşacak bağıl hata için bir üst sınır aşağıdaki şekilde verilir:

$$\left|\frac{x-f\ell(x)}{x}\right| = \left|\frac{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n-(0{,}d_1 d_2 d_3 ... d_k)10^n}{(0{,}d_1 d_2 d_3 ... d_k d_{k+1} ...)10^n}\right|$$

$$= \left| \frac{(0,d_{k+1}d_{k+2}d_{k+3}\ldots)10^{n-k}}{(0,d_1d_2d_3\ldots d_kd_{k+1}\ldots)10^{n}} \right|$$

$$= \left| \frac{(0,d_{k+1}d_{k+2}d_{k+3}\ldots}{0,d_1d_2d_3\ldots d_kd_{k+1}\ldots} \right| 10^{-k}$$

Diğer taraftan $d_1 \neq 0$ olduğundan paydanın alabileceği minimum değer 0,1'dir. Ayrıca pay üstten 1 ile sınırlı olduğundan

$$\left| \frac{x - f\ell(x)}{x} \right| \leq \frac{1}{0,1}.10^{-k} = 10^{-k+1}$$

şeklinde bir üst sınır elde edilir.

KAYNAKÇA

1. Prof. Dr. İbrahim Uzun, Nümerik Analiz, Beta Basın A.Ş., 2012, İstanbul.
2. Prof. Dr. Nuri Özalp – Doç. Dr. Elif Demirci, Nümerik Analiz, Ward Cheney, David Kincaid, Gazi Yayınları, 2012, Ankara.
3. Doç. Dr. Arzu Erdem, Sayısal Analiz, Kocaeli Üniversitesi, Ders notları, 2013, Kocaeli.
4. Doç. Dr. Zekeriya Girgin, Sayısal Analiz, Pamukkale Üniversitesi, Ders notları, 2016, Denizli.
5. Yrd. Doç. Dr. İhsan Temuçin Dolapçı, Yrd. Doç. Dr. Yiğit Aksoy, Sayısal Yöntemler, Celal Bayar Üniversitesi, Ders notları, 2015, Manisa.
6. Yrd. Doç. Dr. Emel Yavuz Duman, Nümerik Analiz, İstanbul Kültür Üniversitesi, Ders notları, 2013, İstanbul.