ガウス分布について

情報通信研究機構
ワイヤレスネットワーク研究所
主任研究員
有本 好徳

1. ガウス関数の定義

通信理論やレーザ光学系の設計・評価においてガウス分布(正規分布)、及び、その積分がよく用いられる。そこで、これらの定義式や近似式等について調べてみた。

(1) 定義

区間の片側で定義した場合の式は、

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \qquad \left(erf(\infty) = 1\right) \tag{1}$$

となるが、xが負の場合を含めて原点に対して対称な積分で表した場合は、

$$erf(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^x e^{-t^2} dt \tag{2}$$

この定義はガウスによるもので、これらの関数で定義した場合の分散は1/2になる。関数の全てのxにわたる積分が1に規格化されていることから、この積分は分布の中心付近の割合を示すものと考えられるが、反対に分布の裾の割合を示すものとして誤差補関数(erfc)が次の式で定義される。

$$erfc(x) = 1 - erf(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$$

一方、正規分布の分布関数の定義は、平均をμ、分散をσとすると、

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{3}$$

となる。この関数もxの全領域で積分すると1になるように規格化されている。

(2)近似式

通信方式の教科書ではビット誤り率(BER)の計算の際にQ関数を良く使う。zを受信機の符号判定点における信号対雑音比とすると、$BER = Q(z)$ となる。この近似式として、宮内一洋著「通信方式入門」には、以下の近似式が紹介されている。

$$Q(z) \approx \frac{1}{z \cdot \sqrt{2\pi}} \left(1 - \frac{0.7}{z^2}\right) e^{-\frac{z^2}{2}} \tag{4}$$

この式の近似精度はz>2.15で1%と説明されており、BERで10^{-2}よりも良い場合には良好な精度が得られる。BERを誤差補関数で表すと、

$$BER = \frac{1}{2} erfc\left(\frac{z}{\sqrt{2}}\right) = Q(z) \tag{5}$$

となるので、式(4)、(5)からerfcの近似式として、次の式が得られる。

$$erfc(z) = 2Q(z\cdot\sqrt{2}) \approx \frac{1}{z\cdot\sqrt{\pi}}\left(1-\frac{0.7}{2z^2}\right)e^{-z^2} \tag{6}$$

この近似式は以下の漸近展開をもとに最適化されたものと思われる。

$$erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt = 1-\frac{e^{-x^2}}{\sqrt{\pi}\cdot x}\left(1-\frac{1}{2x^2}+\frac{1\cdot 3}{2^2x^4}-\frac{1\cdot 3\cdot 5}{2^3x^6}+\dots\right) \tag{7}$$

この展開式自体は全ての項を計算すると発散するが、xの大きな領域で展開項の計算を途中までで打ち切った場合、テイラー展開よりも少ない項数で精度の高い結果が得られる。

式(7)以外の近似式は、例えば英文版Wikipediaで"Error function"で検索すると、次のような近似式が紹介されている。

$$erfc(x) = \begin{pmatrix} 1+7.05230784\cdot 10^{-2}x+4.22820123\cdot 10^{-2}x^2+9.2705272\cdot 10^{-3}x^3 \\ +1.520143\cdot 10^{-4}x^4+2.765672\cdot 10^{-4}x^5+4.30638\cdot 10^{-5}x^6 \end{pmatrix}^{-16} \tag{8}$$

この式の最大誤差は$3\text{x}10^{-7}$である。

$$erfc(x) = (a_1t+a_2t^2+a_3t^3+a_4t^4+a_5t^5)e^{-x^2} \qquad t=\frac{1}{1+px} \tag{9}$$

ここで、a_1=0.254829592, a_2=-0.284496736, a_3=1.421413741, a_4=-1.453152027, a_5=1.061405429, p=0.3275911である。この式の最大誤差は$1.5\text{x}10^{-7}$である。

$$erf(x) = \text{sgn}(x)\sqrt{1-\exp\left(-x^2\frac{4/\pi+ax^2}{1+ax^2}\right)} \qquad a=\frac{8(\pi-3)}{3\pi(4-\pi)}\approx 0.140012 \tag{10}$$

この式の絶対誤差は全てのxについて0.00035以下である。

これらの近似式についてNetlibのFortranのソースコード(W. J. Cody)をもとにした倍精度の計算結果との比較を行ってみた。以下の表でr-diffは相対誤差、a-diffは絶対誤差を示す。

式(8)についての比較結果:

x	erfc6(x)	derfc(x)	r-diff(%)	a-diff
0.00	1.0000000000000000000	1.0000000000000000	0.00	0.00000000
0.50	0.4795003017179373	0.4795001221869535	0.00	0.00000018
1.00	0.1572989536661082	0.1572992070502851	-0.00	-0.00000025
1.50	0.0338950786634172	0.0338948535246893	0.00	0.00000023
2.00	0.0046774829249957	0.0046777349810473	-0.01	-0.00000025
2.50	0.0004071590209555	0.0004069520174450	0.05	0.00000021
3.00	0.0000222644075132	0.0000220904969986	0.79	0.00000017
3.50	0.0000007748050616	0.0000007430983723	4.27	0.00000003
4.00	0.0000000177921089	0.0000000154172579	15.40	0.00000000
4.50	0.0000000002863435	0.0000000001966160	45.64	0.00000000
5.00	0.0000000000034922	0.0000000000015375	127.14	0.00000000

xが3.5以下(BERで10^{-6}以上)であれば相対精度10%以下の精度が出ている。

式(9)についての比較結果:

x	erfc6(x)	derfc(x)	r-diff(%)	a-diff
0.00	1.00000000000000000	1.0000000000000000	0.00	0.00000000
0.50	0.4794999836952530	0.4795001221869535	-0.00	-0.00000014
1.00	0.1572993102524100	0.1572992070502851	0.00	0.00000010
1.50	0.0338947335970280	0.0338948535246893	-0.00	-0.00000012
2.00	0.0046778604187811	0.0046777349810473	0.00	0.00000013
2.50	0.0004070354633940	0.0004069520174450	0.02	0.00000008
3.00	0.0000221051488978	0.0000220904969986	0.07	0.00000001
3.50	0.0000007442171360	0.0000007430983723	0.15	0.00000000
4.00	0.0000000154602958	0.0000000154172579	0.28	0.00000000
4.50	0.0000000001975086	0.0000000001966160	0.45	0.00000000
5.00	0.0000000000015478	0.0000000000015375	0.67	0.00000000

全てのxについて絶対精度7ケタの精度があり、相対精度も良好で、特にBERの小さな領域で精度が高い。

式(10)についての結果:

x	erfc6(x)	derfc(x)	r-diff(%)	a-diff
0.00	1.00000000000000000	1.0000000000000000	0.00	0.00000000
0.50	0.4794807110317622	0.4795001221869535	-0.00	-0.00001941
1.00	0.1570744177568533	0.1572992070502851	-0.14	-0.00022479
1.50	0.0335833981411652	0.0338948535246893	-0.92	-0.00031146
2.00	0.0045553196116770	0.0046777349810473	-2.62	-0.00012242
2.50	0.0003883067024652	0.0004069520174450	-4.58	-0.00001865
3.00	0.0000207864989325	0.0000220904969986	-5.90	-0.00000130
3.50	0.0000006973845483	0.0000007430983723	-6.15	-0.00000005
4.00	0.0000000145977519	0.0000000154172579	-5.32	-0.00000000
4.50	0.0000000001896490	0.0000000001966160	-3.54	-0.00000000
5.00	0.0000000000015219	0.0000000000015375	-1.01	-0.00000000

式(8)、(9)に比べると精度が悪いが、全てのxについて10%の精度で結果が得られる。

以上の評価の結果、単精度(8桁)程度の簡易計算には式(9)が、計算量の少ない式が必要なら式(8)、あるいは式(10)が適していることが分かった。実際には、上記の誤差評価と計算の複雑さを考え合わせて最適な式を選択する必要があると思われる。最近の数値計算ライブラリ(C99やFortran2008)では倍精度の誤差関数が整備されている。これらのライブラリではW. J. Codyのアルゴリズムが用いられているものと思われる。また、エクセルの計算(少なくとも2003年版以降)については正確な計算結果が得られるようである。■