情報システム工学科
伊達 章

2005 年 4 月 1 日
更新：2007 年 4 月 9 日

速修 正規分布

「$X_1, X_2, \cdots . X_{1000}$ を平均 μ, 分散 σ^2 の互いに独立なガウス分布に従う確率変数とする」というような表現がよく使われる．ここでは簡単な例を通して，この文の意味，特に,「互いに独立」,「ガウス分布」,「平均」,「分散」,「確率変数」などの概念について，少なくとも直感的に理解できるようにする．平均 μ，分散 σ^2 の正規分布の確率密度は次のように表せる．

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \tag{1}$$

$x' = (x-\mu)/\sigma$ と変換すれば，x' は平均 0，分散 1 の正規分布にしたがう（逆に言うと， x' をもとに $x = \sigma x' + \mu$ を生成すれば平均 μ，分散 σ^2 の正規分布にしたがうデータが得られる）．これを標準正規分布という．正規分布はガウス分布とも呼ばれ，以下のような形をしている．

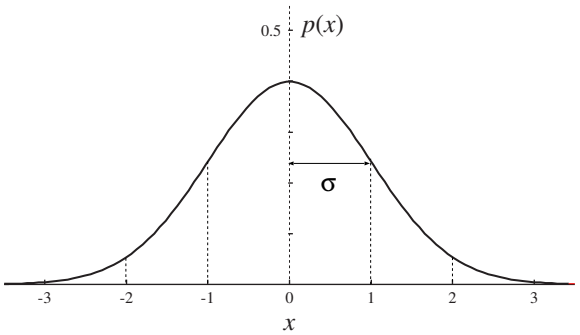

図 1: 平均 0，分散 1 の正規分布の確率密度

「$X_1, X_2, \cdots . X_{1000}$ を平均 0, 分散 1 の互いに独立なガウス分布に従う確率変数とする」というような表現を理解しよう．独立というのは例えば「x_1 の値は x_5 の値とは無関係に定まる」といったことである．また上の図では分布の広がりを σ と書いてあるが，これは標準偏差と呼ばれている量であり，分散は，その 2 乗，σ^2 の事である．あまり深く考えず，分布の広がりと思っていい．正規分布にしたがっていれば，この 1000 個の確率変数の実現値 $x_i, i = 1, \cdots, 1000$ のうち約 68.26% が $-1 < x_i < 1$ に含まれている．その根拠は

$$\int_{-1}^{1} p(x)dx = \frac{1}{\sqrt{2\pi}} \int_{-1}^{1} \exp\left\{-\frac{x^2}{2}\right\} dx = 0.6826 \tag{2}$$

の計算である．同じ様に，実現値のうち 95.44% が $-2 < x_i < 2$ の区間に, 99.74% が $-3 < x_i < 3$ の区間に含まれているはずである．また $x_1 = 100$ という値はめったにでないが，そのようなことがおこる確率は 0 ではない．おこりう事すべてを足しあわせた確率は

$$\int_{-\infty}^{\infty} p(x)dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left\{-\frac{x^2}{2}\right\} dx = 1 \tag{3}$$

となる．$p(x)$ は確率密度関数で確率ではない．上記の計算からもわかるように，標準正規分布にしたがうデータを 1 つとってきたとき，その値 x が，$a < x < b$ にある確率が $\Phi(b) - \Phi(a)$ と計算できる．ここで

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left\{-\frac{t^2}{2}\right\} dt \tag{4}$$

であり，ほとんどの確率・統計の教科書にこの関数の具体的な値が掲載されている．

sample0002.c

```
#include <stdio.h>
#include <stdlib.h> /* for srand48(), drand48() */
#include <math.h>   /* for sqrt(), log() */

double nrand();
double x;

main(){
         int i;
         int seed = 1234567; /* use any number as you like */
         double x;

         srand48( seed );

         for (i=0; i<1000; i++){
                x = nrand();
                printf("%.5lf\n",x);
         }
}

/*
  nrand() :  a random sample from standard normal distribution
  一様分布 drand48() から標準正規分布に従うデータを出力する関数
 */
double nrand()
{
        static int sw=0;
        static double r1,r2,s;

        if (sw==0){
                sw=1;
                do {
                        r1=2.0*drand48()-1.0;
                        r2=2.0*drand48()-1.0;
                        s=r1*r1+r2*r2;
                } while (s>1.0 || s==0.0);
                        s=sqrt(-2.0*log(s)/s);
                        return(r1*s);
        }
        else {
                sw=0;
                return(r2*s);
        }
}
```

```
% gcc sample0002.c -lm
% ./a.out > data001
```

とすれば標準正規分布から 1000 個の乱数がとりだせる．ヒストグラムを書いて確かめてみる．

```
% octave
octave:1> load data001
octave:2> hist(data001, 20)    # 20 の部分を大きい値にすると分布がより細かくみえる．
```

eps ファイルとして出力するには：

```
octave: > __gnuplot_set__ term postscript
octave: > __gnuplot_set__ output "file.eps"
```

もとにもどすには

```
octave: > __gnuplot_set__ term X11
octave: > exit  # octave の終了
% ggv file.eps   # 分布を見る
```