

F-ðāñī ðāāāēāí èā

Í áí çí à÷áí èā	$F(x v_1, v_2)$
Í áēāñöü çí à÷áí èé	$0 \leq x < \infty$
Ī àðāì àòðû	Ēî èè÷āñöāā ñðāī áí áé ñāí áí āû – öāēû ā í î ēī æèðāēūí ū ā ÷ēñēā v_1 è v_2 , ĩ àðāì àòðû ô î ðī ū.
Ī ēī ā í ñöü	$\frac{\Gamma\left(\frac{v_1 + v_2}{2}\right) (v_1/v_2)^{v_1/2} x^{\frac{v_1-2}{2}}}{\Gamma\left(\frac{v_1}{2}\right) \Gamma\left(\frac{v_2}{2}\right) [1 + (v_1/v_2)x]^{\frac{v_1+v_2}{2}}}$
Ī àòāì àòè÷āñēí ā í æāāí èā	$v_1/(v_2 - 2), v_2 > 2$
Āēñī āðñēý	$2v_2^2(v_1 + v_2 - 2)/v_1(v_2 - 2)^2(v_2 - 4), v_2 > 2$
Óóí èöèý ðāñī ðāāāēāí èý	Í ā āû ðāæāāðñý ā ýēāì áí ðāðí ū ð ó í èöèýð

Ī î èāçí ū ā ñāî éñöāà

- ðāñī ðāāāēāí èā $F(x|v_1, v_2)$ è áāðā-ðāñī ðāāāēāí èā $B(x|v_1, v_2)$ ñāýçāí ū ñēāāóþ ù èì ñî ð-í î ø áí èāì : $F(x|v_1, v_2) = B\left(\frac{xv_1}{v_2 + xv_1} \middle| \frac{v_1}{2}, \frac{v_2}{2}\right)$.
- Çí à÷áí èā $(1-\alpha)$ -ēāāí ðèèý ðāñī ðāāāēāí èý $F(x|v_1, v_2)$ āñöü āāèè÷èí à, í áðāóí àý α -ēāāí ðèèþ ðāñī ðāāāēāí èý $F(x|v_2, v_1)$. Òī æā ñāì í ā ô î ðī óēí é:

$$F^{-1}(1 - \alpha|v_1, v_2) = 1/F^{-1}(\alpha|v_2, v_1),$$

çāāñü, èāè è āñþ äó -1 ā èā÷āñöāā āāðóí āāí èí āāēñā āí āí ðèð í á í áðāð í í é ó ó í èöèè.
- Āñēè $\chi_{v_1}^2, \chi_{v_2}^2$ āāā í àçāāèñèì ū ā ñēó÷āéí ū ā āāèè÷èí ū ñ v_1 è v_2 ñðāī áí ýì è ñāí áí āû ñî ð-āāðñöāāí í í, òí $\frac{v_2\chi_{v_1}^2}{v_1\chi_{v_2}^2} \sim F(x|v_1, v_2)$. Ýòí ñî ðí ðí ø áí èā – í äèí èç í ñí í áí ū ð èñòí ÷í èēí ā í ðèì áí áí èý F-ðāñī ðāāāēāí èý, à ðāèæā í ðè÷èí à āû ā í áí í āí āāí í àçāāí èý, ðāñī ðāāāēāí èā àēñī āðñèí í í í āí í ð í í ø áí èý: āāāü í ðè ÷í ðèì áí áí èè àēñī āðñèí í í í āí áí àèèçā í ðèí ýòí ñ÷è-ðāòü, ÷òí èñòí áí ū ā í ááèþ āāí èý ðāñī ðāāāēāí ū í í ðī àēūí í è, çí à÷èð, àēñī āðñèè í í ä÷è-í ýþ ðñý ðāñī ðāāāēāí èþ χ^2 .

Āí ð ñòāðèñðè÷āñēāý í áðāóí ðī óēèðí àēā í í ñēāāí āāí ñāî éñöāà. Ī óñöü āāí ū āāā í ááí ðā í àçāāè-ñèì ū ð ā ñî āí èóí í í ñðè í í ðī àēūí ū ð ñēó÷āéí ū ð āāèè÷èí: $x_i \in N(\mu_x, \sigma_x), i=1..n_x$,

$y_i \in N(\mu_y, \sigma_y)$, $i=1..n_y$. Í ðáááéèì ñéó÷áéí ù á ááèè÷èí ù \bar{x} , \bar{y} , s_x^2 è s_y^2 ñéááóp ù èí íá-ðàçíì :

$$\bar{x} = \sum_{i=1}^{n_x} x_i, \bar{y} = \sum_{i=1}^{n_y} y_i, s_x^2 = \sum_{i=1}^{n_x} (x_i - \bar{x})^2, s_y^2 = \sum_{i=1}^{n_y} (y_i - \bar{y})^2.$$

Òí ááà ñéó÷áéí ày ááèè÷èí à $\left[\frac{n_x s_x^2}{(n_x - 1) \sigma_x^2} / \frac{n_y s_y^2}{(n_y - 1) \sigma_y^2} \right]$ í í á÷èí yáðñy F-ðáñí ðáááéáí èp ñ í à-ðàì áððàì è n_x è n_y .

Ãáí áðàöèy ñéó÷áéí ù ò ÷èñáè

Í áðí áú, ñí áöèðè ÷áñèèá áey ááí í í áí ðáñí ðáááéáí èy, ì í á í áèçááñòí ù.

Âú ÷èñéáí èá ò óí êöèè ðáñí ðáááéáí èy è áá êááí òèèáé

Í ðí ù á áñááí èñí í èüçí áàðü óéàçáí í óp ñáyçü ñ ááðà-ðáñí ðáááéáí èáì .

Áey áú ÷èñéáí èy ñ í í í í ù ùp í èæáñéááóp ù èð êí áí á í ððááóp ðñy ò áééú betaDF.h, betaDF.cpp, logGamma.h, logGamma.cpp.

Òàéè Fdistr.h

```
#ifndef __FDF_H__ /* To prevent redefinition
*/

#define ENTRY extern
#define LOCAL static

double FDF(double n1, double n2, double x);
/* Áú÷èñéyáð ááðíyðííñðü ðíáí, ÷ðí ñéó÷áéíày ááèè÷èíà,
* ííá÷èíyðüàyñy òáíððáèüíííó F-ðáñíðáááéáíèp ñ íàðàìáððàìè n1 è n2,
* íá íðááíñòíáèð çáááíííáí çíà÷áíèy x.
*/
double inv_FDF(double n1, double n2, double p);
/* Èùáð ðí÷éó, á êíðíðíé óóíéöèy F-ðáñíðáááéáíèy
* ñ íàðàìáððàìè n1 è n2 ðááíyáðñy p
*/

#define __FDF_H__ /* Prevents redefinition
*/
```

```
#endif /* Ends #ifndef __FDF_H__
*/
```

Ôàéë Fdistr.cpp

```
#include <math.h>
#include <assert.h>

#include "betaDF.h"

ENTRY double
FDF(double n1, double n2, double x)
// Áú÷ēñÿàðñÿ āāðíÿðíññòü òíāí, ÷òí ñēó÷āéíāÿ āāēē÷ēíā,
// íā÷ēíÿpùāÿñÿ òāíððāēüíííó F-ðāñīðāāāēāíēp ñ íāðāíāððāìè n1 è n2,
// íā íðāāíññōíāèð çāāāíííāí çíā÷āíēÿ x.
{
    assert(x >= 0.0);
    return 1.0 - BetaDF(0.5 * n2, 0.5 * n1).value(n2 / (n2 + n1 * x));
} /*FDF*/

ENTRY double
inv_FDF(double n1, double n2, double p)
// Èüāð òí÷ēó, ā ēíðíðíé óóíēòēÿ F-ðāñīðāāāēāíēÿ
// ñ íāðāíāððāìè n1 è n2 ðāāíÿàðñÿ p
{
    double y=BetaDF(0.5 * n1, 0.5 * n2).inv(p);
    return n2*y/n1/(1-y);
} /*inv_FDF*/

#ifdef TEST_IT

#include <stdio.h>

LOCAL void
tryF(void)
{
    double m, n, x, p;
    double alpha=0.95;

    while (1) {
        printf("\n\n\rEnter integer n1: ");
        scanf("%lg", &n);
        if (n <= 0)
            break;
        printf("\rEnter integer n2: ");
        scanf("%lg", &m);
        printf("\rEnter F: ");
        scanf("%lg", &p);
        x = FDF(n, m, p);
        printf("\nF(%.0lf, %.0lf, %lg) = %lg", n, m, p, x);
    }
} /*tryF*/

LOCAL void
try_invF(void)
{
    double n1, n2, bound;
```

```
double alpha=0.95;

while (1) {
    printf("\n\n\rEnter integer n1: ");
    scanf("%lg", &n1);
    if (n1 <= 0)
        break;
    printf("\rEnter integer n2: ");
    scanf("%lg", &n2);
    bound = inv_FDF(n1,n2,alpha);
    printf("\n%.2f bound = %lg", alpha, bound);
}

}/*try_invF*/

void main(void)
{
    tryF();
    try_invF();
}

#endif          /* Ends #ifdef TEST_IT */
```