Метод Монте-Карло

Ерофей Башунов, M3336 (вариант 4)

16 июня 2020 г.

Задание №1

Условие

Методом Монте-Карло оценить объем части тела $\{F(\overline{x}) \le c\}$, заключѐнной в k-мерном кубе с ребром $[0, 1]$. Функция имеет вид $F(\overline{x}) = f(x_1) + f(x_2) + \dots + f(x_k)$. Для выбранной надежности $\gamma \ge 0.95$ указать асимптотическую точность оценивания и построить асимптотический доверительный интервал для истинного значения объёма.

Используя объём выборки $n = 10^4$ и $n = 10^6$, оценить скорость сходимости и показать, что доверительные интервалы пересекаются.

$$f(x) = \ln(ax + 1) \qquad k = 3 \qquad c = 2.8 \qquad a = 3$$

Решение

Следующий код на языке Octave является решением поставленной задачи.

```
pkg load nan
clc
clear

# parameter gamma
global gamma = 0.95;
global T = norminv((gamma + 1) / 2)

# initial conditions of variant 4
global k = 3;
global c = 2.8;
global a = 3;
function result = f(x)
    global a;
    result = log(a * x + 1);
endfunction

# confidence interval search function
function result = monteCarloCube(n)
    global T;
    global k;
    global c;
    x = rand(n, k);
    y = f(x);
    z = sum(y');
    p = mean(z <= c);
    d = T * sqrt(p * (1 - p ) / n);
    I = [p - d, p + d];
    result = I;
    printf("n = %d\n", n);
    printf("p = %f\n", p);
```

```
    printf("d = %f\n", d);
    printf("I = [%f, %f]\n", I)
    printf("\n")
endfunction

# interval intersection check function
function result = checkIntervals(I1, I2)
    if (I1(2) < I2(1))
        result = false;
    elseif (I1(1) > I2(2))
        result = false;
    else
        result = true;
    endif
endfunction

# solution
I1 = monteCarloCube(10^4);
I2 = monteCarloCube(10^6);
if (checkIntervals(I1, I2))
    printf("intervals intersect");
else
    printf("intervals do not intersect");
endif
```

n	10^4	10^6
p	0.637100	0.628023
d	0.009424	0.000947
I	$[0.627676, 0.646524]$	$[0.627076, 0.628970]$
$length(I)$	0.018848	0.001894

При увеличении выборки в 100 раз доверительный интервал уменьшился приблизительно в 10 раз. Доверительные интервалы для выборок пересекаются на промежутке $[0.627676, 0.628970]$.

Задание №2

Условие

Аналогично построить оценку интегралов (представить интеграл как математическое ожидание функции, зависящей от случайной величины с известной плотностью) и для выбранной надежности указать асимптотическую точность оценивания и построить асимптотический доверительный интервал для истинного значения интеграла.

$$\text{a.} \int_1^4 3^{-x^2} dx \qquad \text{b.} \int_0^{+\infty} x^{\frac{2}{3}} e^{-2x} dx$$

Решение

Пункт а.

```
pkg load nan
pkg load statistics
clc
clear

# parameter gamma
global gamma = 0.95;
global T = norminv((gamma + 1) / 2)

# initial conditions of variant 4
global k = 3;
global from = 1;
global to = 4;

# confidence interval search function
function result = monteCarloUnifrnd(n)
    global T;
    global k;
    global from;
    global to;
    x = unifrnd (from, to, n, k);
    z = (to - from) * (3 .^ (-(x .^ 2)));
    p = mean(mean(z));
    d = mean(T * std(z) / sqrt(n));
    I = [p - d, p + d];
    result = I;
    printf("n = %d\n", n);
    printf("p = %f\n", p);
    printf("d = %f\n", d);
    printf("I = [%f, %f]\n", I)
    printf("\n")
endfunction

# interval intersection check function
function result = checkIntervals(I1, I2)
    if (I1(2) < I2(1))
        result = false;
    elseif (I1(1) > I2(2))
        result = false;
    else
        result = true;
    endif
endfunction
```

```
# solution
I1 = monteCarloUnifrnd(10^4);
I2 = monteCarloUnifrnd(10^6);
R = quad(@(x) 3 .^ (-x .^ 2), from, to);
printf("real value = %f\n", R);
if (checkIntervals(I1, I2))
    printf("intervals intersect");
else
    printf("intervals do not intersect");
endif
```

n	10^4	10^6
p	0.115727	0.116991
d	0.004390	0.000443
I	$[0.111337, 0.120117]$	$[0.116548, 0.117434]$
$length(I)$	0.00878	0.000886

При увеличении выборки в 100 раз доверительный интервал уменьшился приблизительно в 10 раз. Доверительные интервалы для выборок пересекаются на промежутке $[0.116548, 0.117434]$. Реальное значение интеграла равно 0.116901 и лежит в пределах обоих доверительных интервалов.

Пункт b.

```
pkg load nan
pkg load statistics
clc
clear

# parameter gamma
global gamma = 0.95;
global T = norminv((gamma + 1) / 2)

# initial conditions of variant 4
global k = 3;
global lambda = 1/2;

# confidence interval search function
function result = monteCarloExprnd(n)
    global T;
    global k;
    global lambda;
    x = exprnd(lambda, n, k);
    z = lambda * (x .^ (2/3));
    p = mean(mean(z));
    d = mean(T * std(z) / sqrt(n));
    I = [p - d, p + d];
    result = I;
```

```
    printf("n = %d\n", n);
    printf("p = %f\n", p);
    printf("d = %f\n", d);
    printf("I = [%f, %f]\n", I)
    printf("\n")
endfunction

# interval intersection check function
function result = checkIntervals(I1, I2)
    if (I1(2) < I2(1))
        result = false;
    elseif (I1(1) > I2(2))
        result = false;
    else
        result = true;
    endif
endfunction

# solution
I1 = monteCarloExprnd(10^4);
I2 = monteCarloExprnd(10^6);
R = quad(@(x) (x .^ (2/3)) .* exp(-2 * x), 0, inf);
printf("real value = %f\n", R);
if (checkIntervals(I1, I2))
    printf("intervals intersect");
else
    printf("intervals do not intersect");
endif
```

n	10^4	10^6
p	0.284764	0.284422
d	0.003797	0.000379
I	$[0.280967, 0.288561]$	$[0.284043, 0.284801]$
$length(I)$	0.007594	0.000758

При увеличении выборки в 100 раз доверительный интервал уменьшился приблизительно в 10 раз. Доверительные интервалы для выборок пересекаются на промежутке $[0.284043, 0.284801]$. Реальное значение интеграла равно 0.284347 и лежит в пределах обоих доверительных интервалов.