Домашнее задание 1, ТВМС

Ковешников Глеб, M3238

3 марта 2020 г.

Метод Монте-Карло
Вариант №10

1 задание

1.1 Формулировка

Методом Монте-Карло оценить объем части тела $\{F(\tilde{x}) \leq c\}$, заключенной в k-мерном кубе с ребром $[0, 1]$. Функция имеет вид $F(\tilde{x}) = f(x_1) + f(x_2) + ... + f(x_k)$. Для выбранной надежности $\gamma \geq 0.95$ указать асимптотическую точность оценки 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.

1.2 Входные данные

- Функция имеет вид $f(x) = x^a$
- Куб размерностью $k = 10$
- Параметр $c = 2.21$
- Параметр $a = 3$

1.3 Исходный код программы

```
pkg load statistics;

function monte_carlo(n)
  k = 10;
  y = 0.95;
  c = 2.21;
  a = 3;
  Q = norminv((y + 1) / 2);
  X = rand(k, n);
  F_x = sum(X.^a);
  v = mean(F_x <= c);
  delta = Q * sqrt(v * (1 - v) / n);
  printf("%g %g %g\n\n", v, v - delta, v + delta);
  printf("Delta is %g\n", delta);
endfunction

monte_carlo(10^4);
monte_carlo(10^6);
```

1.4 Выходные данные

```
0.3878 0.37825 0.39735
Delta is 0.0095499

0.395959 0.395 0.396918
Delta is 0.000958532
```

1.5 Вывод

Доверительный интервал при $n = 10^6$ содержится в интервале при $n = 10^4$.
При увеличении числа итераций в 100 раз ширина доверительного интервала уменьшилось в 10 раз.

2 задание

2.1 Формулировка

Построить оценку интегралов (представить интеграл как математическое ожидание функции, зависящей от случайной величины с известной плотностью) и для выбранной надежности $\gamma \geq 0.95$ указать асимптотическую точность оценки и построить асимптотический доверительный интервал для истинного значения интеграла.

2.2 Интеграл 1

$$\int\limits_{-\infty}^{\infty} \sqrt{1+x^2} exp\left(\frac{-(x+2)^2}{4}\right) dx$$

2.2.1 Исходный код программы

```
pkg load statistics;

function res = g(x)
  res = sqrt(1 + x * x);
endfunction

function res = g1(x)
  res = g(x) * sqrt(2 * pi * 2);
endfunction

function res = f(x)
  res = g(x) * exp(-(x + 2) ^ 2 / 4);
endfunction

function calc_value(n)
  mu = -2;
  sigma = sqrt(2);
  y = 0.95;
  T = norminv((y + 1) / 2);
  X = normrnd(mu, sigma, 1, n);
  F_x = arrayfun(@g1, X);
  v = mean(F_x);
  delta = (std(F_x) * T) / sqrt(n);
  printf("%g %g %g\n", v, v - delta, v + delta);
  printf("Delta is %g\n\n", delta);
endfunction

printf("Sample answer = %g\n\n", quad(@f, -inf, inf));
calc_value(10^4);
calc_value(10^6);
```

2.2.2 Выходные данные

```
Sample answer = 8.54521

8.54964 8.47385 8.62544
Delta is 0.0757972

8.54044 8.53287 8.54801
Delta is 0.00756963
```

2.2.3 Вывод

Истинное значение интеграла содержится в доверительном интервале при $n = 10^4$ и $n = 10^6$. Значение, полученное методом Монте-Карло отличается от значения, полученного методом *quad*, на $7.5 \cdot 10^{-4}$. При увеличении числа итераций в 100 раз, ширина доверительного интервала уменьшилось в 10 раз.

2.3 Интеграл 2

$$\int\limits_0^5 \frac{\sin(x)}{x^2+1}dx$$

2.3.1 Исходный код программы

```
pkg load statistics;

function res = f(x)
  res = sin(x) / (x^2 + 1);
endfunction

function monte_carlo(n)
  l = 0;
  r = 5;
  y = 0.95;
  Q = norminv((y + 1) / 2);
  X = unifrnd(l, r, 1, n);
  F_x = arrayfun(@f, X) * (r - l);
  v = mean(F_x);
  delta = (std(F_x) * Q) / sqrt(n);
  printf("%g %g %g\n", v, v - delta, v + delta);
  printf("Delta is %g\n\n", delta);
endfunction

printf("Sample answer = %g\n\n", quad(@f, 0, 5));
monte_carlo(10^4);
monte_carlo(10^6);
```

2.3.2 Выходные данные

```
Sample answer = 0.648162

0.642251 0.625347 0.659155
Delta is 0.0169043

0.64845 0.646764 0.650136
Delta is 0.00168573
```

2.3.3 Вывод

Истинное значение интеграла содержится в доверительном интервале при $n = 10^4$ и $n = 10^6$. Значение, полученное методом Монте-Карло отличается от значения, полученного методом *quad*, на $1.6 \cdot 10^{-4}$. При увеличении числа итераций в 100 раз, ширина доверительного интервала уменьшилось в 10 раз.