

Обязательные для сдачи задания

Метод Монте-Карло может использоваться для вычисления интеграла (площади под кривой).

Будем считать, что требуется вычислить интеграл на интервале $[0, 1]$ и значение функции также лежит в таком же интервале. В этом случае график функции помещается в единичный квадрат (рис. 1).

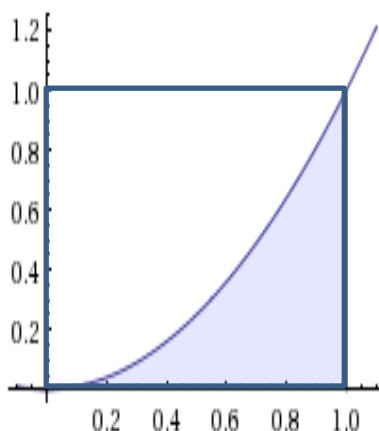


Рис. 1. Функция и единичный квадрат

Допустим, мы будем случайным образом ставить точку внутри единичного квадрата. В этом случае, вероятность попадания точки под кривую зависит от площади, ограниченной этой кривой: чем больше площадь, тем «легче» попасть в соответствующую область. Более точно, отношение числа попавших под кривую точек к общему числу точек стремится к отношению площади под кривой к общей площади квадрата.

Пусть было проведено N случайных испытаний, в ходе которых n случайно выбранных точек попало в область под кривой. В этом случае

$$n/N \rightarrow S/S_{square},$$

где S – площадь под кривой, S_{square} – площадь квадрата, в котором проводились случайные испытания. В рассматриваемом случае $S_{square} = 1$ и $n/N \rightarrow S$. Обратите внимание, что стоящее слева выражение – это оценка вероятности наступления события «точка попала под кривую» после проведения n испытаний.

Таким образом, метод Монте-Карло для вычисления интеграла реализуется следующим образом.

- Проводятся случайные испытания, **исходами** которых являются случайные точки внутри единичного квадрата.
- В качестве **события** рассматривается попадание точки в область под графиком функции.
- На основании проведения большого числа испытаний оценивают ожидаемое значение вероятности наступления такого события. Найденная оценка используется как приближенная оценка интеграла.

В первых трёх заданиях требуется реализовать этот метод вычисления площади для функции $\sin(x)$ на отрезке $[0, 1]$ (рис. 2). Точное значение этого интеграла равно $1 - \cos(1) \approx 0.45970$.

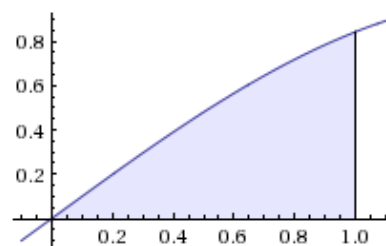


Рис. 2. Интеграл функции $\sin(x)$ на отрезке $[0, 1]$

Задание 1

Реализуйте функцию `compute_trial()`, проводящую случайное испытание и возвращающую его исход.

Исходом должны быть координаты точки, выбранной случайным образом внутри единичного квадрата.

Координаты должны представляться кортежем вида (x, y) .

Вызовите функцию `compute_trial()` несколько раз, чтобы убедиться, что она работает правильно.

Задание 2

Реализуйте функцию `check_event()`, проверяющую наступление события «точка под графиком».

Параметром функции `check_event()` является `point` – точка (кортеж вида (x, y)).

Функция должна проверить, находится ли точка не выше графика функции $\sin(x)$ и вернуть `True`, если это так.

Проверьте работу функции `check_event()` с помощью `doctest`'ов.

Задание 3

Реализуйте функцию `mc_area()`, вычисляющую площадь под кривой внутри единичного квадрата.

Параметром этой функции является `trials` – число испытаний.

Функция должна провести заданное число испытаний, оценить вероятность наступления события и вернуть эту вероятность. При реализации этой функции используйте функции `compute_trial()` и `check_event()` из предыдущих заданий.

Оценка обязательной части

- | | | |
|---|--|---------|
| 1 | Испытание и исход | - 20 %. |
| 2 | Событие (при наличии <code>doctest</code> 'ов) | - 20 %. |
| 3 | Вычисление площади | - 60 %. |

Дополнительные задания

Задание 4

Реализуйте вычисление приближённого значения числа π методом Монте-Карло.

Для этого:

- придумайте геометрическую фигуру (или функцию), площадь которой выражается через число π ;
- реализуйте событие, соответствующее попаданию точки внутрь этой фигуры (функции);
- рассчитайте площадь фигуры методом Монте-Карло;
- вычислите π через полученную оценку площади.

Вычисление реализуйте в функции `mc_pi(trials)`, параметром которой является число экспериментов для метода Монте-Карло.

Оценка дополнительной части

Задание 4 – 100 %