Python задание 1: моделирование вероятности

1 Частотная интерпретации вероятности

Теория вероятностей — это математическая дисциплина, которая изучает абстрактные модели, описанные аксиомами, сформулированными Андреем Николаевичем Колмогоровым. Математиков в этой области интересует только правильность выводов о вероятностях, а вот вопрос, что именно означает вероятность и как она связана с реальностью, для них вторичен. С точки зрения математики, если что-то соответствует вероятностным аксиомам, это можно считать вероятностью. Однако то, как это применить на практике, — это уже задача философии и других наук.

В этом задании мы будем полагаться на частотную (или фреквентистскую) интерпретацию вероятности. Представьте себе эксперимент, в результате которого событие A может либо произойти, либо нет. Нам интересно узнать вероятность того, что событие A произойдёт. Согласно частотной интерпретации, чтобы измерить вероятность A, нужно многократно повторить этот эксперимент и посчитать долю случаев, в которых событие A произошло:

$$\frac{\text{Количество случаев, когда А произошло}}{\text{Общее количество экспериментов}}.$$

В пределе это отношение стремится к истинной вероятности $\mathbb{P}(A)$. К счастью, компьютеры позволяют нам многократно повторять один и тот же вероятностный эксперимент, что делает эту задачу выполнимой на практике.

2 Вычисление числа π

В этом задании мы будем использовать понятие геометрической вероятности, чтобы оценить значение числа π.

2.1 Геометрическая вероятность

Геометрическая вероятность — это вероятность того, что случайно выбранная точка в ограниченной геометрической фигуре (например, квадрате, круге или другом теле) попадёт в заданную область внутри этой фигуры. Эта вероятность определяется как отношение меры (например, площади или объёма) интересующей нас области к мере всей фигуры. Геометрическая вероятность широко применяется в задачах, где требуется моделировать случайные события на основе геометрических свойств объектов.

2.2 Бросание точки в квадрат

Чтобы случайным образом выбрать точку внутри квадрата, необходимо использовать равномерный генератор случайных чисел. Равномерное распределение означает, что все значения внутри заданного диапазона имеют одинаковую вероятность быть выбраны.

Представьте, что у нас есть квадрат со сторонами, длина которых варьируется от 0 до 1. Чтобы случайным образом выбрать точку внутри этого квадрата, мы можем отдельно сгенерировать две координаты x и y для точки, где x и y — это случайные числа, равномерно распределённые на отрезке $[0; 1]$. Такой способ генерации обеспечит равномерное распределение точек по всему квадрату, так как каждая точка внутри квадрата имеет одинаковую вероятность быть выбранной.

Пример кода на Python для генерации такой точки:

```python
import random

x = random.uniform(0, 1)
y = random.uniform(0, 1)

print(f"Random point: ({x}, {y})")
```

В этом коде функция `random.uniform(0, 1)` возвращает случайное число, равномерно распределённое на отрезке $[0; 1]$. Соответственно, точка с координатами (x, y) будет равномерно распределена внутри квадрата.

2.3 Общая идея эксперимента

Идея нашего эксперимента состоит в следующем: мы будем многократно случайным образом "бросать"точки в квадрат и моделировать вероятность того, что выбранная точка попадёт в окружность, вписанную в этот квадрат. Вероятность того, что случайная точка, выбранная внутри квадрата, попадёт в окружность, пропорциональна отношению площади окружности к площади квадрата. Так как площадь окружности пропорциональна числу π, измеряя долю точек, попавших в окружность, мы можем оценить значение π.

Процесс можно описать следующим образом:

1. Мы генерируем множество точек внутри квадрата с помощью равномерного распределения.

2. Для каждой точки проверяем, попала ли она внутрь окружности.

3. Подсчитываем долю точек, попавших в окружность, по сравнению с общим числом точек.

4. Используя это соотношение, приближённо вычисляем значение числа π.

Ваша основная задача — подробно обдумать детали этого алгоритма и реализовать его.

Дополнительно вам необходимо построить график зависимости точности приближения числа π от количества использованных точек. Для этого не обязательно генерировать новые точки на каждом шаге, а можно пересчитывать вероятность, приближенное значение π и, соответственно, точность приближения после генерациикаждой новой точки. Это позволит визуализировать, как увеличивается точность с ростом числа использованных точек.

Описанный здесь метод для вычисления числа π называется также методом Монте-Карло. Этот метод основан на повторении экспериментов для получения статистически значимых результатов, которые позволяют оценить числовые характеристики и вероятности.

3 Вычисление вероятности перколяции

Предыдущее задание было слишком искуственным, ведь есть гораздо лучшие способы подсчёта числа π. Поэтому теперь мы попробуем искать вероятности, для которых моделирование уже необходимо.

В данном задании вы исследуете концепцию перколяции, которая находит применение в самых разных областях — от распространения эпидемий до проникновения жидкости в пористые материалы. Перколяция описывает процесс образования связного пути через систему, которая состоит из случайно распределенных узлов или элементов.

Мы будем рассматривать частный случай модели, представляющей собой прямоугольное поле, разделенное на мелкие одинаковые квадраты. Каждый квадратик может быть либо проходимым, либо непроходимым. Проходимость каждой клетки определяется согласно заранее заданной вероятности.

Основная задача состоит в проведении множества экспериментов, в каждом из которых будет генерироваться новое поле согласно заданной вероятности проходимости. В каждом эксперименте вы проверите существование пути, соединяющего верхнюю и нижнюю стороны поля, проходящего только через проходимые клетки. На основе этих данных вы оцените *вероятность перколяции*, то есть вероятность наличия такого пути. От вас требуется построить график, отображающий зависимость вероятности перколяции от вероятности проходимости каждой клетки.

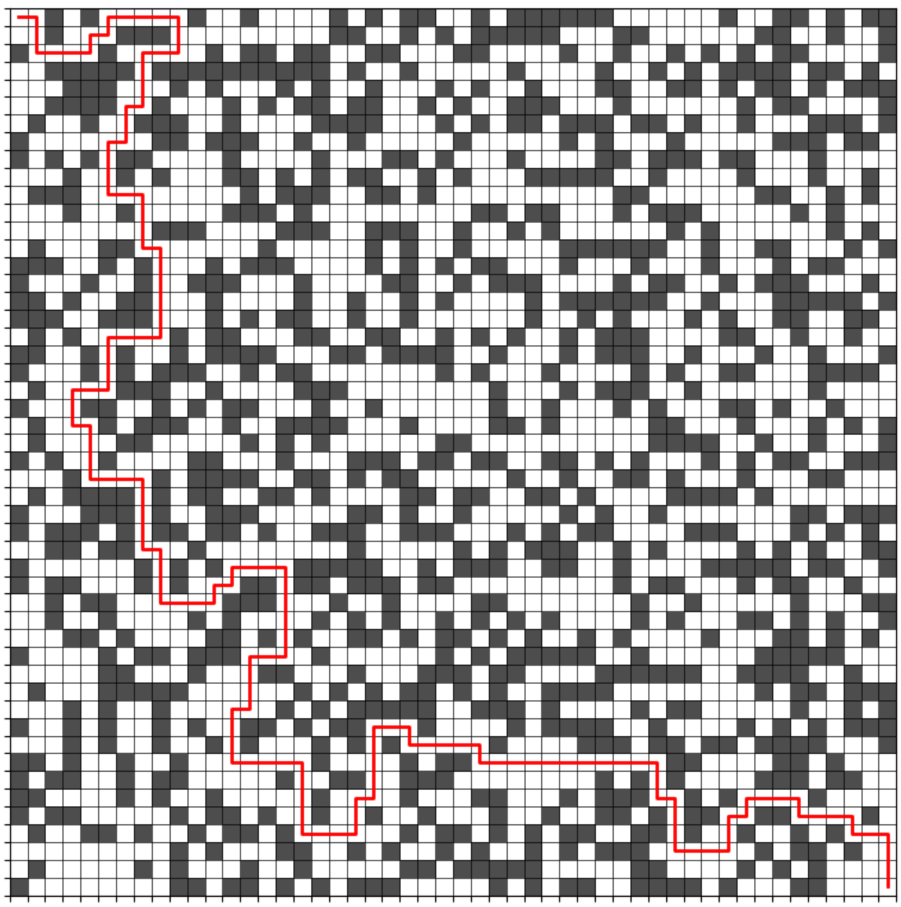

Рис. 1: Иллюстрация задачи о перколяции.