ГУАП

КАФЕДРА № 41

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Е. К. Григорьев |
| должн., уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 3. |
| МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ВЕЛИЧИН С ЗАДАННЫМ ЗАКОНОМ РАСПРЕДЕЛЕНИЯ МЕТОДОМ ОБРАТНОЙ ФУНКЦИИ |
| по курсу: МОДЕЛИРОВАНИЕ. |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4217 |  |  |  | | У. А. Мазориев |
|  |  |  | подпись, дата | |  | инициалы, фамилия |

**Цель работы:**

Получить навыки моделирования наиболее известных генераторов равномерно распределенных псевдослучайных чисел в программной среде Python, а также первичной оценки качества полученных псевдослучайных чисел.

**Результат выполнения работы**

Программный код можно посмотреть, перейдя по ссылке на GitHub: <https://github.com/Mrx112426/Modelirovanie/tree/main>

**Ход выполнения работы**

Распределение Рэлея — распределение вероятностей X случайной величины

f(x;σ)=xσ2exp⁡(−x22σ2),x⩾0,σ>0, Изображение выглядит как текст, Шрифт, рукописный текст, белый

Контент, сгенерированный ИИ, может содержать ошибки.

где — параметр масштаба. Соответствующая функция распределения имеет вид

P(X⩽x)=∫0xf(ξ)dξ=1−exp⁡(−x22σ2),x⩾0. Изображение выглядит как Шрифт, рукописный текст, текст, белый

Контент, сгенерированный ИИ, может содержать ошибки.

На рисунке 1 показан график плотности распределения вероятностей распределение Рэлея.

Изображение выглядит как текст, линия, диаграмма, График

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1 - Плотность распределения вероятностей

На рисунке 2 показан график функции распределения.

Изображение выглядит как текст, линия, График, диаграмма

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2 - График функции распределения

Метод обратной функции - метод для генерации случайных величин с заданным распределением на основе того, что функция распределения известна. Суть метода заключается в том, чтобы найти обратную функцию от распределения и применить её для получения случайных чисел с нужной функцией распределения.

Если U - случайная величина, распределенная равномерно на [0,1], то значение X, распределённое по можно получить из уравнения:

Здесь - функция распределения случайной величины X.

После реализуется скрипт на Python, результаты для N = [1000, 5000, 10000], результаты показаны на рисунках 3 - 5.

Изображение выглядит как текст, диаграмма, График, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 3 - Визуализация графиков N = 1000



Рисунок 4 - Визуализация графиков N = 5000

Изображение выглядит как диаграмма, линия, График, текст

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 5 - Визуализация графиков N = 10000

Также выводятся основные характеристики (рис. 6).

Изображение выглядит как текст, снимок экрана, меню, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 6 - Подсчитанные характеристики для N = [1000, 5000, 10000]

Проделанная работа показывает, что генератор псевдослучайных чисел работает хорошо, с увеличением объёма выборки наблюдается сходимость выборочных характеристик к теоретическим значениям, что подтверждается уменьшением разницы между расчетными и ожидаемыми средними, дисперсией и стандартным отклонением. С ростом выборки увеличивается размах данных, что связано с большей вероятностью появления редких значений в более крупных выборках. Коэффициенты асимметрии и эксцесса остаются положительными, что свидетельствует о скошенности распределения вправо и его более вытянутой форме по сравнению с нормальным распределением.

**Вывод**

В ходе лабораторной работы были получены навыки моделирования случайных величин с заданным законом распределения методом обратной функции в программной среде Python, а также первичной оценки качества полученных псевдослучайных чисел.