Санкт-Петербургский Политехнический университет Петра Великого

Институт компьютерных наук и кибербезопасности

Высшая школа программной инженерии

Лабораторная работа №2

Дисциплина: Статистическое моделирование

Выполнил: студент группы 5130904/10102

Иванов К. А.

Преподаватель: Чуркин В. В.

2024

г. Санкт-Петербург

Цель работы:

1.Практическое освоение методов получения случайных величин, имеющих дискретный характер распределения.

2.Разработка программных датчиков дискретных случайных величин.

3.Исследование характеристик моделируемых датчиков:
3.1. Оценка точности моделирования: вычисление математического ожидания и дисперсии, сравнение полученных оценок с соответствующими теоретическими значениями.

4.Графическое представление функции плотности распределения и интегральной функции распределения.

Ход работы:

1. Дискретное распределение

Алгоритм 1:

$\boxed{r = (r\_up - r\_low + 1)*u + r\_low}$

Объем выборки: 10^4, low = 1, up = 100

Оценка	IRNUNI	Погрешность	Теоретическое значение
M	50.80176464183916	0.3017646418391635	50.5
D	836.6725036452414	3.4225036452413633	833.25

Рисунок 1 - Результаты равномерного распределения

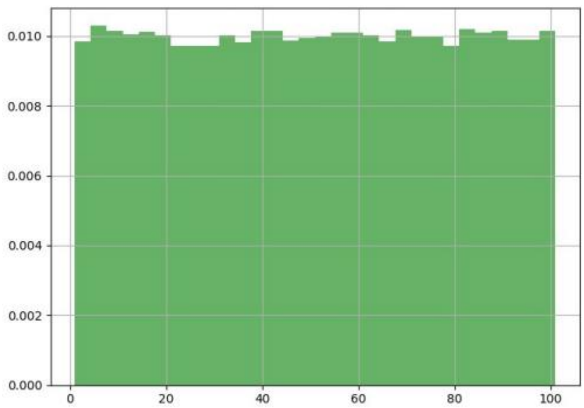

Рисунок 2 - Плотность распределения

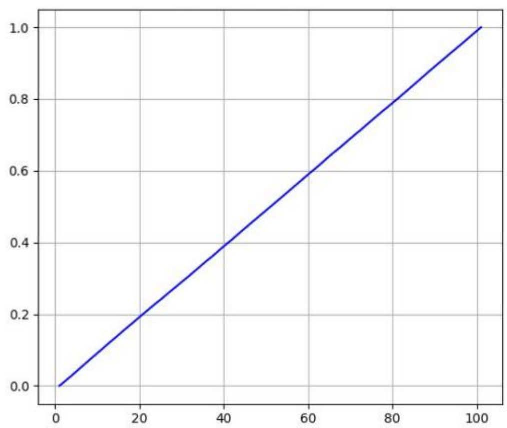

Рисунок 3 - Функция распределения

2. Биномиальное распределение

Объем выборки: 10^4, N=10, p=0,5

Алгоритм 1: рекуррентный метод

```
p(0)=(1-p)**N,
… … … … … … … … …
… … … … … … … … …
p(r)=p(r-1)*[((N-r)/(r+1))*(p/(1-p))]
```

Алгоритм 2: нормальная аппроксимация

```
IR = RNNORM(N*p,SQRT(N*p*(1.0-p))) + 0.5
```

Оценка	IRNBIN	IRNBNL	Теоретическое значение
M	4.9959	5.004344496592488	5.0
D	2.498863190000031	2.5028184140029017	2.5

Рисунок 4 - Результаты биномиального распределения

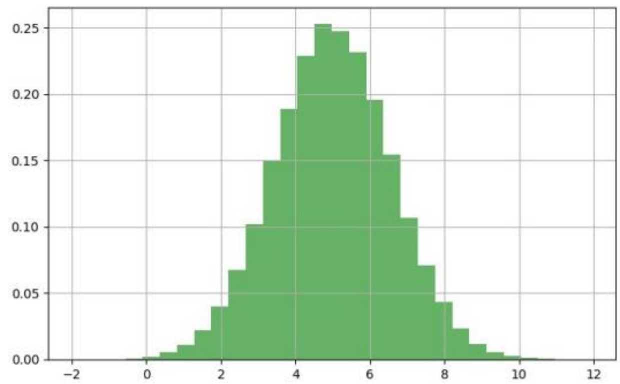

Рисунок 5 - Плотность распределения

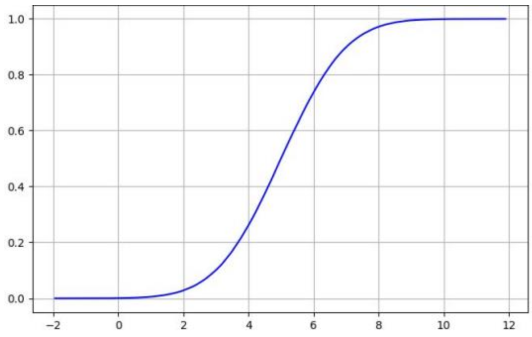

Рисунок 6 - Функция распределения

3. Геометрическое распределение

Объем выборки: 10^4, p=0,5

Алгоритм 1: рекуррентный метод

p(0) = p
… … … … …
p(r) =p(r-1)*(1-p), где r=1,2, …

Алгоритм 2: Прямой метод заключается в получении псевдослучайной последовательности равномерно распределенных случайных чисел u[1], u[2],... в интервале [0,1], до тех пор пока не найдется u[k] "успешный", который меньше или равен p.

Алгоритм 3: вариация первого алгоритма

k = int[ln(u)/ln(q)]+1

Оценка	IRNGEO_1	IRNGEO_2	IRNGEO_3	Теоретическое значение
M	1.98988	2.00475	2.00209	2.0
D	1.9704975855988898	2.0102874374993487	2.009405631899556	2.0

Рисунок 7 - Результаты геометрического распределения

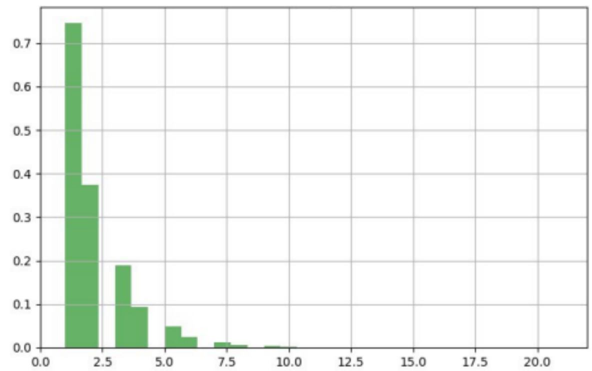

Рисунок 8 - Плотность распределения

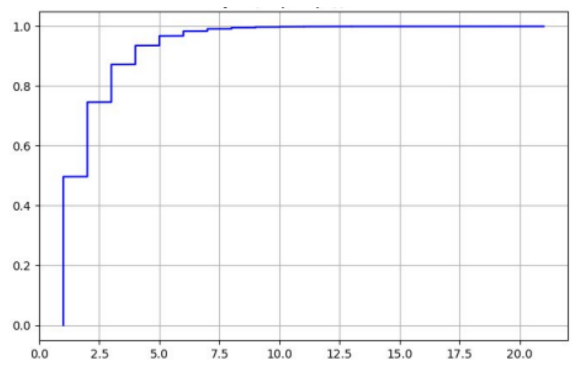

Рисунок 9 - Функция распределения

4. Распределение Пуассона

Объем выборки: 10^4, mu=10

Алгоритм 1: рекуррентный метод

p(0) = e **(-mu)
… … … … … … … … … …
P(r) = P(r-1)*mu/r для r = 1,2,…

Алгоритм 2: перемножении равномерно распределенных случайных чисел до тех пор, пока выполняется условие

Ro -mu П x[i] >= e i=0

Оценка	IRNPOI	IRNPSN	Теоретическое значение
M	9.99058	10.032733613485187	10.0
D	10.070291263598396	9.996152381913673	10.0

Рисунок 10 - Результаты распределения Пуассона

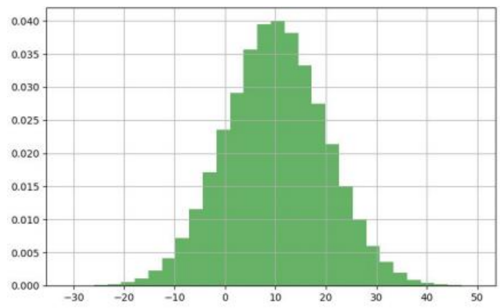

Рисунок 11 - Плотность распределения

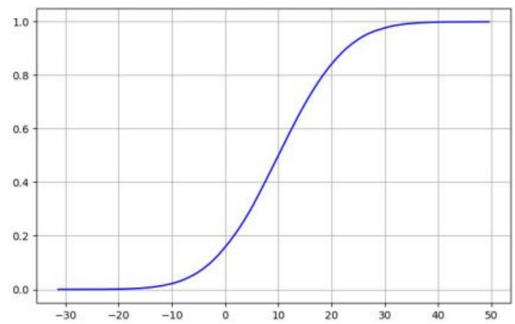

Рисунок 12 - Функция распределения

Индивидуальное задание, Вариант 9

Цель:

Реализовать один из тестов на случайность из набора тестов Д. Кнута – проверка интервалов. Описание теста приведено на стр. 140 [3], проверить сгенерированную выборку, разрядность слов и длина выбирается в соответствии с рекомендациями в [3].

Ход работы:

Проверка интервалов. Данный тест проверяет равномерность распределения символов в исследуемой последовательности, анализируя длины подпоследовательностей, все элементы которых принадлежат определенному числовому интервалу.

Пусть $\varepsilon = \varepsilon_1\varepsilon_2\ldots\varepsilon_n$ – последовательность *m*-разрядных чисел. Пусть α и β – два целых числа, таких, что $0 \le \alpha < \beta \le 2^m - 1$. Подсчитываются длины интервалов между числами, лежащими в промежутке $[\alpha; \beta]$. После этого определяется число интервалов ν_i, $i = \overline{0, t}$, длины 0, 1, 2, …, t и рассчитывается статистика:

$$\chi^2(obs) = \sum_{i=0}^{t} \frac{\left[\nu_i - \eta\frac{\beta-\alpha}{2^m}\left(1-\frac{\beta-\alpha}{2^m}\right)^i\right]^2}{\eta\frac{\beta-\alpha}{2^m}\left(1-\frac{\beta-\alpha}{2^m}\right)^i},$$

где $\eta = \sum_{i=0}^{t}\nu_i$ – общее число интервалов.

Полученный результат анализируется при помощи критерия χ^2 с числом степеней свободы, равным t.

[5, 6, 5, 5, 6, 6, 6, 7, 6, 7, 6, 6, 5, 4, 5, 3, 8, 3, 4, 5, 2, 5, 4, 5, 5, 4, 7, 5, 4, 5, 5, 3, 7, 7, 5, 3, 6, 8, 6, 5, 3, 2, 5, 8, 4, 4, 7, 5, 6, 2]

[0, 3, 6]

[0, 0, 0]

хи2 = 0

хи2_крит = 5.991464547107979

Таким образом, проверка интервалов прошла тест

GitHub: https://github.com/bpllqd/matstat/tree/main/Lab2