*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
| **Изображение выглядит как эмблема, герб, нашивка, символ  Автоматически созданное описание** | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(национальный исследовательский университет)***  ***(МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ФУНДАМЕНТАЛЬНЫЕ НАУКИ\_\_\_\_\_\_\_\_

**КАФЕДРА** ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И МАТЕМАТИЧЕСКАЯ ФИЗИКА (ФН11)\_

**НАПРАВЛЕНИЕ ПОДГОТОВКИ** МАТЕМАТИКА И КОМПЬЮТЕРНЫЕ НАУКИ (02.03.01)

**Отчет**

**по лабораторной работе № \_7\_**

**Название лабораторной работы: Критерий согласия для проверки простой непараметрической гипотезы**

**Вариант № 2**

**Дисциплина:** Теория вероятности и математическая статистика

Студент группы ФН11-52Б **\_\_\_\_\_\_\_\_\_\_\_\_\_ Кожемякин Г.А.**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_ Облакова Т.В.**

(Подпись, дата) (И.О. Фамилия)

Москва, 2024

**Задание**

Постройте с помощью стохастического эксперимента на основе указанной метрики приближенный критерий для проверки основной гипотезы. Найдите критические значения для трех уровней значимости .

Протестируйте критерий на трех-четырех примерах и сформулируйте выводы.

**Исходные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Вариант | Основная гипотеза | Метрика | Объем выборки |
| 22 |  |  |  |

**Основная гипотеза:**

(A=0) Выборка получена из распределения

()

-количество значений, попавших в -ый интервал группировки

-теоретическая вероятность попадания в -ый интервал группировки

**Решение задачи**

Смоделируем выборку, подчиняющуюся основной гипотезе , m=5000 раз, получим матрицу размера 5000 \* 500:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0.20846054 | 0.48168106 | 0.42053804 | … | 0.29423118 | 0.40776007 | 0.74514439 |
| 0.9023332 | 0.65206676 | 0.08046449 | … | 0.67659699 | 0.41244338 | 0.23062308 |
| 0.59529911 | 0.27612519 | 0.68172998 | … | 0.82754857 | 0.27678105 | 0.29894817 |
| … | … | … | … | … | … | … |
| 0.57489516 | 0.77782137 | 0.6915006 | … | 0.84629992 | 0.48255577 | 0.14937382 |
| 0.64272261 | 0.57195532 | 0.53934089 | … | 0.4154515 | 0.3381279 | 0.83200386 |
| 0.2557776 | 0.75631123 | 0.51107757 | … | 0.81500714 | 0.06380435 | 0.50284429 |

Сгруппируем полученные данные для каждой из выборок, число интервалов по формуле Стерджесса:

Интервалы имеют вид:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [0,1/9] | (1/9,2/9] | (2/9,1/3] | (1/3,4/9] | (4/9,5/9] | (5/9, 2/3] | (2/3, 7/9] | (7/9, 8/9] | (8/9, 1] |

Получим число элементов в полученных интервалах для каждой выборки:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [0,1/9] | (1/9,2/9] | (2/9,1/3] | (1/3,4/9] | (4/9,5/9] | (5/9, 2/3] | (2/3, 7/9] | (7/9, 8/9] | (8/9, 1] |
| 68 | 57 | 39 | 53 | 62 | 59 | 59 | 53 | 50 |
| 53 | 62 | 55 | 62 | 57 | 45 | 66 | 52 | 48 |
| 53 | 64 | 57 | 55 | 64 | 52 | 48 | 63 | 44 |
| … | … | … | … | … | … | … | … | … |
| 70 | 53 | 64 | 45 | 48 | 47 | 50 | 65 | 58 |

Для вычисления метрики найдем теоретические вероятности попадания случайной величины в каждый интервал. Так как основная гипотеза состоит в том, что случайная величина распределена равномерно на интервале [0,1], имеем:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | [0,1/9] | (1/9,2/9] | (2/9,1/3] | (1/3,4/9] | (4/9,5/9] | (5/9, 2/3] | (2/3, 7/9] | (7/9, 8/9] | (8/9, 1] |
| pi | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 |
| n\*pi | 55.56 | 55.56 | 55.56 | 55.56 | 55.56 | 55.56 | 55.56 | 55.56 | 55.56 |

Вычислим и отсортируем метрику D для сгруппированных выборок, получим найденные значения для каждой выборки:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0.024444 | 0.028889 | 0.031111 | … | 0.207111 | 0.211111 | 0.219111 |

Вычислим эмпирические квантили для различных уровней значимости, используя функцию np.quantile, они будут равны:

|  |  |  |  |
| --- | --- | --- | --- |
| α | 0.1 | 0.05 | 0.01 |
| Dcr | 0.1369 | 0.1471 | 0.1689 |

**Тестирование критерия для различных распределений**

1. Выборка из n случайных величин из исходного распределения R[0,1]

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Получим D=0.0987, то есть , следовательно, основная гипотеза о распределении выборки принимается во всех трех случаях.

2. Выборка из n случайных величин, заданная законом

***Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание***

Получим D=0.5209, то есть , следовательно, основная гипотеза о распределении выборки отклоняется во всех трех случаях.

3. Выборка из n случайных величин, заданная

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Получим D=0.5129, то есть , следовательно, основная гипотеза о распределении выборки отклоняется во всех трех случаях.

4. Выборка из n случайных величин бэта-распределения с параметрами 0.25, 0.75

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

Получим D=0.7316, то есть , следовательно, основная гипотеза о распределении выборки отклоняется во всех трех случаях.

**Выводы**

В ходе проделанной лабораторной работы был проведен стохастический эксперимент, на основе указанной метрики построен приближенный критерий для проверки основной гипотезы, а также найдены критические значения для трех уровней значимости . Построенный критерий был протестирован на четырех выборках из разных законов распределения и показал свою эффективность.

**Приложение**

В приложении представлен программный код на языке Python.

**import** numpy **as** np  
np.random.seed(22)  
n = 500  
m = 5000  
data = np.random.uniform(0, 1, size=(m, n))  
data.reshape(m, n)  
  
sample = np.zeros((m, 9))  
intervals = np.linspace(0, 1, 10)  
print(intervals)  
**for** i **in** range(m):  
 counts, \_ = np.histogram(data[i], bins=intervals)  
 sample[i] = counts  
 **if** i **in** [0, 1, 2, 4999]:  
 print(counts)  
  
print(sample)  
  
p = [1 / 9 **for** i **in** range(9)]  
mcounts = [p[i] \* n **for** i **in** range(9)]  
mcounts  
  
D = (1 / n) \* np.sum(np.abs(sample - 55.555555556), axis=1)  
np.sort(D)  
  
alpha\_ = [0.1, 0.05, 0.01]  
quantiles\_alpha = np.quantile(D, [1 - alpha **for** alpha **in** alpha\_ ], method =**'nearest'**)  
quantiles\_alpha  
  
np.random.seed(100)  
data1 = np.random.uniform(0, 1, n)  
counts1, \_ = np.histogram(data1, bins=intervals)  
D1 = (1 / n) \* np.sum(np.abs(counts1 - 55.555555556))  
D1  
  
np.random.seed(99)  
ksi = np.random.uniform(0, 1, n)  
data2 = np.sqrt(ksi)  
counts2, \_ = np.histogram(data2, bins=intervals )  
D2 = (1 / n) \* np.sum(np.abs(counts2 - 55.555555556))  
D2  
  
np.random.seed(98)  
ksi = np.random.uniform(0, 1, n)  
data3 = 1 - np.sqrt(ksi)  
counts3, \_ = np.histogram(data3, bins = intervals )  
D3 = (1 / n) \* np.sum(np.abs(counts3 - 55.555555556))  
D3  
  
np.random.seed(97)  
data4 = np.random.beta(0.25, 0.75, n)  
counts4, \_ = np.histogram(data4, bins=intervals )  
D4 = (1 / n) \* np.sum(np.abs(counts4 - 55.555555556))  
D4