Отчет о проделанной работе (Лабораторная работа 4, Вариант 2)

Теория вероятностей и математическая статистика

Михаил Басанец, Ян Кордияко

Белорусский Государственный Университет
Факультет Прикладной Математики и Информатики
2020

1 Условия

Рассматривается выборка объемом n = 100. Необходимо проверить с помощью критерия хи-квадрат гипотезы $\Pi(1)$, $Bi(4, 0.5)$.

2 Используемые формулы

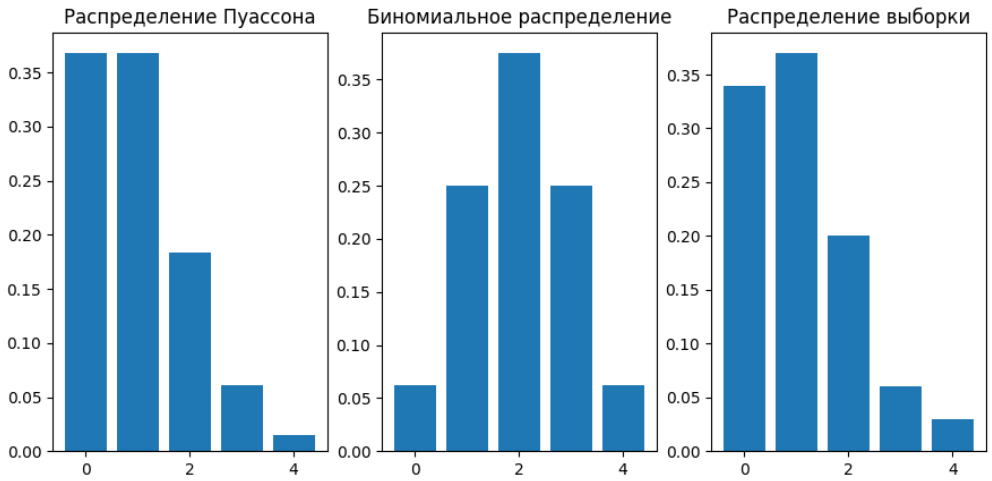
$$P_{\text{Пуассона}}(k) = \frac{\lambda^k}{k!}e^{-\lambda} \text{ – функция вероятности р. Пуассона,} \tag{1}$$

$$P_{\text{биномиальное}}(k) = \binom{n}{k}p^k(1-p)^{n-k} \text{ – функция вероятности биномиального р.,} \tag{2}$$

$$\widetilde{\chi^2_n} = n\sum_{j=1}^{n}\frac{(q_j - p_j(\widetilde{\theta}))^2}{p_j(\widetilde{\theta})} \text{ – статистика хи-квадрат.} \tag{3}$$

3 Ход работы

Сначала нарисуем гистограммы гипотетических и выборочного распределений:



Как мы можем наблюдать на гистограммах, распределение выборки довольно сильно напоминает распределение Пуассона, в то время как оно значительно отличается от биномиального распределения. Проверим наши наблюдения с помощью критерия хи-квадрат.

Для начала разделим выборку на интервалы. В качестве интервалов возьмём:

$$[0, 1), [1, 2), [2, 3), [3, 4), [4, 5).$$

Так как количество элементов, попавших в последний интервал менее 5, то объединим его с предыдущим. Таким образом, получим $l = 4$ интервала:

$$\Delta_1 = [0, 1), \Delta_2 = [1, 2), \Delta_3 = [2, 3), \Delta_4 = [3, 5).$$

Теперь подсчитаем эмпирические вероятности попадания в соответствующие интервалы:

$$q_1 = 0.34, q_2 = 0.37, q_3 = 0.2, q_4 = 0.09.$$

Затем для подсчитаем теоритические вероятности попадания в соответствующие интервалы для распределения Пуассона с помощью (1):

$$p_1 = 0.36787944117144233,\ p_2 = 0.36787944117144233,$$

$$p_3 = 0.18393972058572114,\ p_4 = 0.07664155024405049.$$

Теперь подсчитаем статистику хи-квадрат (3):

$$\widetilde{\chi_n^2} = 0.5855658625776508.$$

Распределение Пуассона имеет $k = 1$ степень свободы. Также возьмём уровень значимости $\alpha = 0.05$. Следовательно, находим в соответствующей таблице пороговое значение статистики хи-квадрат:

$$C_{0.05} = \chi^2_{0.05,\ l-k-1} = 5.991.$$

Очевидно, что $\widetilde{\chi_n^2} < C_{0.05}$, следовательно данная гипотеза принимается.

Теперь проверим критерий хи-квадрат для биномиального распределения. Сначала подсчитаем теоритические вероятности попадания в соответствующие интервалы для этого распределения с помощью (2):

$$p_1 = 0.0625,\ p_2 = 0.25000000000000006,$$

$$p_3 = 0.3750000000000001,\ p_4 = 0.31250000000000006.$$

Теперь подсчитаем статистику хи-квадрат (3):

$$\widetilde{\chi_n^2} = 152.97866666666673.$$

Биномиальное распределение имеет $k = 2$ степени свободы. Также возьмём уровень значимости $\alpha = 0.05$. Следовательно, находим в соответствующей таблице пороговое значение статистики хи-квадрат:

$$C_{0.05} = \chi^2_{0.05,\ l-k-1} = 3.841.$$

Очевидно, что $\widetilde{\chi_n^2} > C_{0.05}$, причём понятно, что при любом другом адекватном уровне значимости α неравенство сохранит знак. Следовательно, данная гипотеза отвергается.

4 Выводы

Критерий проверки гипотезы Пирсона подтвердил наши наблюдения, основанные на сравнительных гистограмах. Распределение выборки оказалось близким к распределению Пуассона, но совсем не похожим на биномиальное распределение.

5 Исходный код программы

```
import matplotlib.pyplot as plt
from scipy.stats import poisson, binom
import numpy as np
# параметры гипотетических распределений:
# lmbd — для распределения Пуассона
# n_bin, pr_bin — для биномиального распределения
lmbd = 1
n_bin = 4
pr_bin = 0.5
# выборка
X = [0, 0, 3, 1, 1, 0, 1, 1, 0, 1, 4, 1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 3,
     3, 1, 1, 1, 2, 0, 1, 0, 2, 1, 1, 0, 2, 0, 2, 1, 1, 0, 2, 1, 2, 0, 2,
     0, 0, 1, 0, 0, 0, 0, 1, 1, 4, 1, 2, 2, 0, 0, 0, 1, 1, 1, 1, 3, 3, 2,
     1, 2, 1, 0, 2, 1, 2, 0, 0, 1, 2, 2, 2, 0, 4, 1, 1, 2, 3, 2, 0, 1, 1,
     0, 0, 0, 2, 1, 0, 2, 0]
# размер выборки
n = len(X)
# интервалы, на которые разбита выборка: в силу особенностей выборки
# будем считать, что 0 соотвествует интервалу [0, 1),
# 1 — интервалу [1, 2), ..., i — интервалу [i, i + 1), ...
intervals = np.sort(np.unique(np.array(X)))
# число интервалов
k = len(intervals)
# массивы, в которых будем хранить теоритические вероятности попадания
# в соответствующий интервал для каждого распределения: p_poisson —
# для распределения Пуассона, p_binomial — для биномиального
p_poisson = []
p_binomial = []
# массив, в i-м элементе котором будем хранить количество элементов
# выборки, попавших в [i, i + 1)
intervalFrequencies = [0.0]*len(intervals)
# циклом проходим по всей выборке
for x in X:
    # увеличиваем соответствующее количество элементов
    # в данном интервале
    intervalFrequencies[x] += 1
# циклом проходим по всем интервалам
for interval in intervals:
    # подсчитываем функцию вероятности Пуассона (1) и биномиальную (2)
    # для данного значения (интервала)
    p_poisson.append(poisson.pmf(interval, lmbd))
    p_binomial.append(binom.pmf(interval, n_bin,
                                pr_bin))
# нормируем массив с частотой попадания в интервалы
q = [nu / n for nu in intervalFrequencies]
# рисуем гистограммы
plt.subplot(1, 3, 1)
plt.title('Распределение Пуассона')
plt.bar(intervals, np.array(p_poisson))
plt.subplot(1, 3, 2)
plt.title('Биномиальное распределение')
plt.bar(intervals, np.array(p_binomial))
plt.subplot(1, 3, 3)
```

```
plt.title('Распределение выборки')
plt.bar(intervals, np.array(q))
plt.show()
# здесь объединяем интервалы, в которых находится меньше 5 элементов
threshold = 5.0 / n
i = 0
while i < len(q) - 1:
    if(q[i] < threshold):
        q[i] += q[i + 1]
        del q[i + 1]
        p_poisson[i] += p_poisson[i + 1]
        p_binomial[i] += p_binomial[i + 1]
        del p_poisson[i + 1]
        del p_binomial[i + 1]
    else:
        i += 1
i = len(q) - 1
while i > 0:
    if(q[i] < threshold):
        q[i] += q[i - 1]
        del q[i - 1]
        p_poisson[i] += p_poisson[i - 1]
        p_binomial[i] += p_binomial[i - 1]
        del p_poisson[i - 1]
        del p_binomial[i - 1]
    i -= 1
# выведем значения эмпирических вероятностей
# попадания в интервалы
print(q)
# выведем значения теоритических вероятностей попадания в
# интервалы для р. Пуассона
print(p_poisson)
# вычисляем значение статистики хи-квадрат для выборки и
# распределения Пуассона по формуле (3)
chiSquarePoisson = sum([n*(q[x] - p_poisson[x])*(q[x] - p_poisson[x]) /
                        p_poisson[x] for x in range(len(q))])
print(chiSquarePoisson)
# выведем значения теоритических вероятностей попадания
# в интервалы для биномиального р.
print(p_binomial)
# вычисляем значение статистики хи-квадрат для выборки и
# биномиального распределения по формуле (3)
chiSquareBinomial = sum([n*(q[x] - p_binomial[x])*(q[x] - p_binomial[x]) /
                         p_binomial[x] for x in range(len(q))])
print(chiSquareBinomial)
```