

# Наумов П. 3413

Тема. Законы распределения дискретных случайных величин

Привести примеры (сформулировать условие задачи, нельзя брать задачи из лекций и рассмотренные на практике из учебника) дискретных случайных величин, распределенных по: биномиальному закону распределения; закону распределения Пуассона; геометрическому закону распределения; гипергеометрическому закону распределения. Для каждой ДСВ составить закон распределения, найти математическое ожидание, дисперсию, среднее квадратическое отклонение и моду, отобразить многоугольник распределения.

1. Биномиальный закон распределения Условие задачи: Сеть кофеен проводит акцию: в каждой десятой чашке кофе клиент находит под крышкой купон на бесплатный десерт. За день постоянный клиент покупает 6 чашек кофе. Случайная величина  $X$  — количество найденных им купонов за день. Для каждой ДСВ составить закон распределения, найти математическое ожидание, дисперсию, среднее квадратическое отклонение и моду, отобразить многоугольник распределения.

Ввод [4]:

```
import numpy as np
from scipy.stats import binom, poisson, geom, hypergeom
import matplotlib.pyplot as plt
```

```

Ввод [5]: n, p = 6, 0.1
X_binom = np.arange(0, n + 1)
probabilities_binom = binom.pmf(X_binom, n, p)

# Закон распределения
print("Закон распределения:")
for x, prob in zip(X_binom, probabilities_binom):
    print(f"P(X = {x}) = {prob:.4f}")

# Числовые характеристики
binomial_dist = binom(n, p)
print(f'Мат. ожидание = {binomial_dist.mean():.4f}')
print(f'Дисперсия = {binomial_dist.var():.4f}')
print(f'Среднее квадратическое отклонение = {binomial_dist.std():.4f}')

max_prob = probabilities_binom.max()
mode = [int(x) for x in X_binom if round(probabilities_binom[x], 6) == round(max_prob)]
print(f'Мода = {mode}')


# Многоугольник распределения
plt.figure(figsize=(10, 6))
plt.stem(X_binom, probabilities_binom, basefmt=' ')
plt.title('Биномиальное распределение\nn=6, p=0.1')
plt.xlabel('Количество купонов (X)')
plt.ylabel('Вероятность P(X)')
plt.grid(True, alpha=0.3)
plt.show()

```

Закон распределения:

$P(X = 0) = 0.5314$

$P(X = 1) = 0.3543$

$P(X = 2) = 0.0984$

$P(X = 3) = 0.0146$

$P(X = 4) = 0.0012$

$P(X = 5) = 0.0001$

$P(X = 6) = 0.0000$

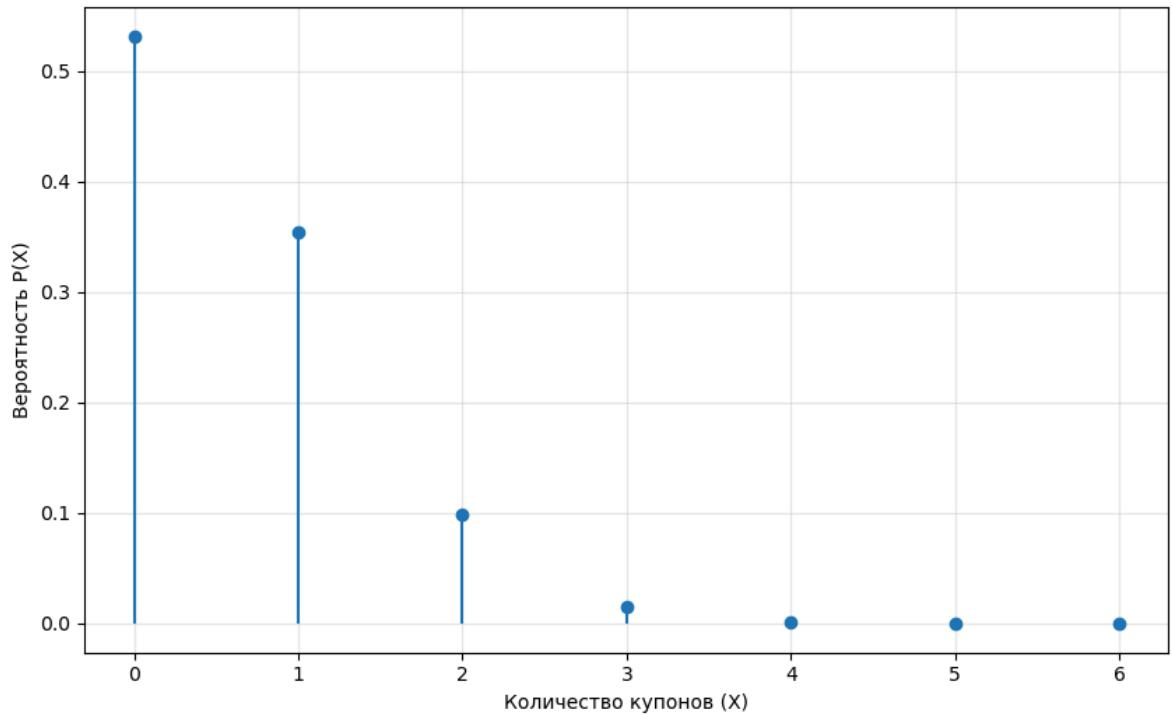
Мат. ожидание = 0.6000

Дисперсия = 0.5400

Среднее квадратическое отклонение = 0.7348

Мода = [0]

Биномиальное распределение  
 $n=6, p=0.1$



2. Закон распределения Пуассона Условие задачи: Колл-центр крупного банка в среднем принимает 4 звонка в минуту. Случайная величина  $Y$  — число звонков, поступивших в колл-центр за любую случайно выбранную минуту в течение дня. Для каждой ДСВ составить закон распределения, найти математическое ожидание, дисперсию, среднее квадратическое отклонение и моду, отобразить многоугольник распределения.

```

Ввод [6]: lambda_ = 4
Y_poisson = np.arange(0, 10) # Берем достаточно большой диапазон
probabilities_poisson = poisson.pmf(Y_poisson, lambda_)

# Закон распределения
print("Закон распределения:")
for y, prob in zip(Y_poisson, probabilities_poisson):
    print(f"P(Y = {y}) = {prob:.4f}")

# Числовые характеристики
poisson_dist = poisson(lambda_)
print(f'Мат. ожидание = {poisson_dist.mean():.4f}')
print(f'Дисперсия = {poisson_dist.var():.4f}')
print(f'Среднее квадратическое отклонение = {poisson_dist.std():.4f}')

max_prob = probabilities_poisson.max()
mode = [int(y) for y in Y_poisson if round(probabilities_poisson[y], 6) == round(max_prob, 6)]
print(f'Мода = {mode}')

# Многоугольник распределения
plt.figure(figsize=(10, 6))
plt.stem(Y_poisson[:12], probabilities_poisson[:12], basefmt=' ')
plt.title('Распределение Пуассона\nλ=4')
plt.xlabel('Количество звонков (Y)')
plt.ylabel('Вероятность P(Y)')
plt.grid(True, alpha=0.3)
plt.show()

```

Закон распределения:

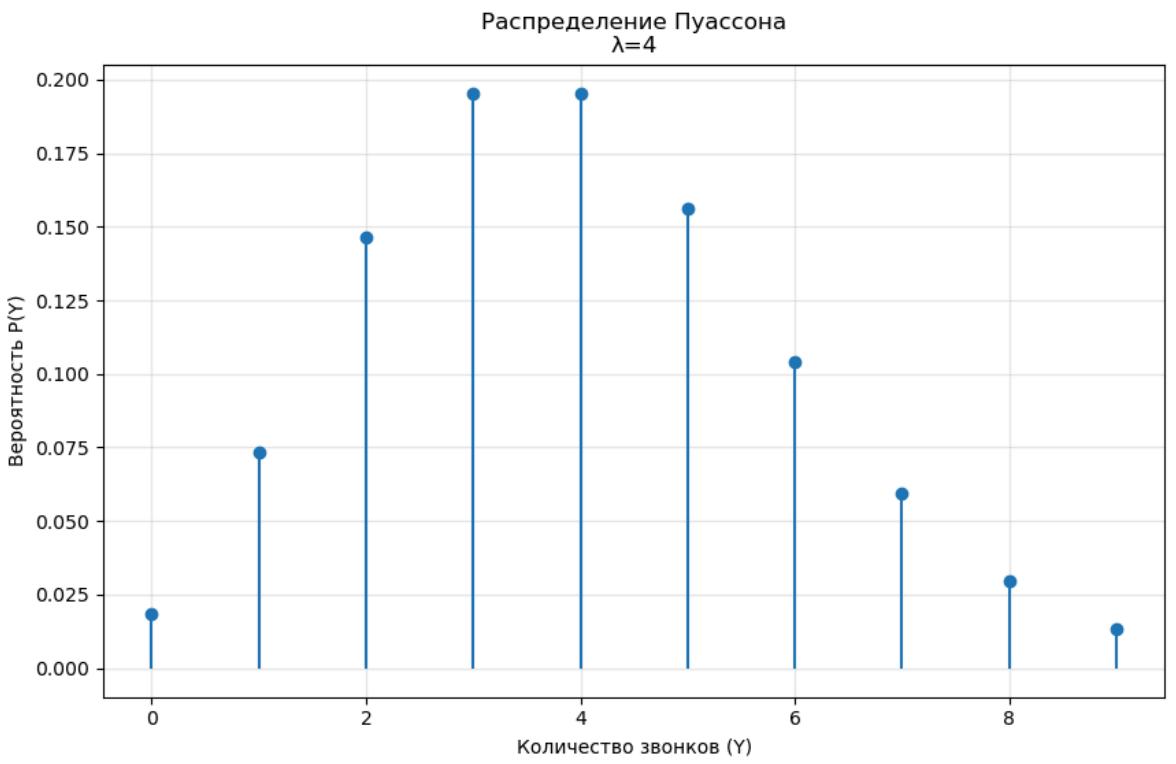
$P(Y = 0) = 0.0183$   
 $P(Y = 1) = 0.0733$   
 $P(Y = 2) = 0.1465$   
 $P(Y = 3) = 0.1954$   
 $P(Y = 4) = 0.1954$   
 $P(Y = 5) = 0.1563$   
 $P(Y = 6) = 0.1042$   
 $P(Y = 7) = 0.0595$   
 $P(Y = 8) = 0.0298$   
 $P(Y = 9) = 0.0132$

Мат. ожидание = 4.0000

Дисперсия = 4.0000

Среднее квадратическое отклонение = 2.0000

Мода = [3, 4]



3. Геометрический закон распределения Условие задачи: Спортсмен выполняет броски баскетбольного мяча по корзине до первого попадания. Вероятность попадания при каждом броске постоянна и равна 0.3. Случайная величина  $Z$  — количество бросков, которое требуется спортсмену до первого попадания (включая успешный бросок). Для каждой ДСВ составить закон распределения, найти математическое ожидание, дисперсию, среднее квадратическое отклонение и моду, отобразить многоугольник распределения.

Ввод [8]:

```
p_geom = 0.3
Z_geom = np.arange(1, 11)
probabilities_geom = geom.pmf(Z_geom, p_geom)

# Закон распределения
print("Закон распределения (первые 10 значений):")
for z, prob in zip(Z_geom, probabilities_geom):
    print(f"P(Z = {z}) = {prob:.4f}")

# Числовые характеристики
geometric_dist = geom(p_geom)
print(f'Мат. ожидание = {geometric_dist.mean():.4f}')
print(f'Дисперсия = {geometric_dist.var():.4f}')
print(f'Среднее квадратическое отклонение = {geometric_dist.std():.4f}')

max_prob = probabilities_geom.max()
mode = [int(z) for z in Z_geom if round(probabilities_geom[z-1], 6) == round(max_prob, 6)]
print(f'Мода = {mode}')

# Многоугольник распределения
plt.figure(figsize=(10, 6))
plt.stem(Z_geom, probabilities_geom, basefmt=' ')
plt.title('Геометрическое распределение\nnp=0.3')
plt.xlabel('Количество бросков до попадания (Z)')
plt.ylabel('Вероятность P(Z)')
plt.grid(True, alpha=0.3)
plt.show()
```

Закон распределения (первые 10 значений):

P(Z = 1) = 0.3000  
P(Z = 2) = 0.2100  
P(Z = 3) = 0.1470  
P(Z = 4) = 0.1029  
P(Z = 5) = 0.0720  
P(Z = 6) = 0.0504  
P(Z = 7) = 0.0353  
P(Z = 8) = 0.0247  
P(Z = 9) = 0.0173  
P(Z = 10) = 0.0121

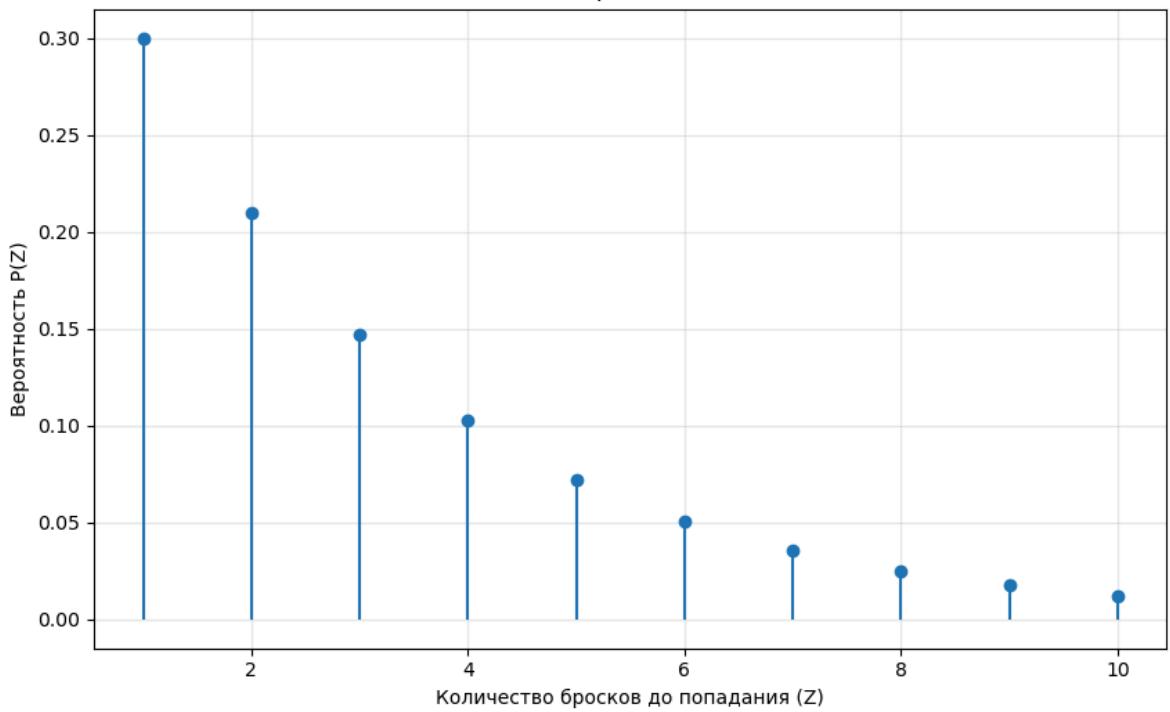
Мат. ожидание = 3.3333

Дисперсия = 7.7778

Среднее квадратическое отклонение = 2.7889

Мода = [1]

Геометрическое распределение  
 $p=0.3$



4. Гипергеометрический закон распределения Условие задачи: В отделе из 15 сотрудников случайным образом выбирают 5 человек для представления компании на выставке. Известно, что в отделе 6 человек владеют английским языком на уровне носителя. Случайная величина K — количество сотрудников, владеющих английским на уровне носителя, среди отобранных пяти. Для каждой ДСВ составить закон распределения, найти математическое ожидание, дисперсию, среднее квадратическое отклонение и моду, отобразить многоугольник распределения.

Ввод [12]:

```
N, M, n = 15, 6, 5
K_hypergeom = np.arange(0, min(M, n) + 1)
probabilities_hypergeom = hypergeom.pmf(K_hypergeom, N, M, n)

# Закон распределения
print("Закон распределения:")
for k, prob in zip(K_hypergeom, probabilities_hypergeom):
    print(f"P(K = {k}) = {prob:.4f}")

# Числовые характеристики
hypergeom_dist = hypergeom(N, M, n)
print(f'Мат. ожидание = {hypergeom_dist.mean():.4f}')
print(f'Дисперсия = {hypergeom_dist.var():.4f}')
print(f'Среднее квадратическое отклонение = {hypergeom_dist.std():.4f}')

max_prob = probabilities_hypergeom.max()
mode = [int(k) for k in K_hypergeom if round(probabilities_hypergeom[k], 6) ==
print(f'Мода = {mode}')

# Многоугольник распределения
plt.figure(figsize=(8, 4))
plt.plot(K_hypergeom, probabilities_hypergeom, '-bo', ms=8, label='hypergeom pmf')
plt.xlabel('Количество носителей языка (K)')
plt.ylabel('Вероятность P(K)')
plt.title('Гипергеометрическое распределение')
plt.grid(True, alpha=0.3)
plt.legend()
plt.show()
```

Закон распределения:

P(K = 0) = 0.0420

P(K = 1) = 0.2517

P(K = 2) = 0.4196

P(K = 3) = 0.2398

P(K = 4) = 0.0450

P(K = 5) = 0.0020

Мат. ожидание = 2.0000

Дисперсия = 0.8571

Среднее квадратическое отклонение = 0.9258

Мода = [2]

### Гипергеометрическое распределение

