

## Lab 6

### Quiz: Sampling data and analyzing distributions: Bernoulli, Binomial

Arin Chauhan

S24CSEU1777

Level: 3rd Semester

Total Questions: 7

Duration: 25 minutes

#### Section A: Multiple Choice Questions

1. If a coin is tossed 10 times, and the probability of heads is 0.5, what is the expected number of heads: a) 2.5  
b) 5  
c) 7.5  
d) 10

Answer:  $n \times p = 10 \times 0.5 = 5$ . b) 5

2. Which of the following is true about the relationship between Bernoulli and Binomial distributions?  
a) Bernoulli is a special case of Binomial when  $n > 1$ .  
b) Binomial is a special case of Bernoulli when  $p = 0.5$ .  
c) Bernoulli is a special case of Binomial when  $n = 1$ .  
d) they are completely unrelated.

Answer: c) Bernoulli is a special case of Binomial when  $n = 1$ .

3. In a binomial distribution with  $n = 8$  and  $p = 0.3$ , what is the variance?  
a) 1.68  
b) 2.4  
c) 0.7  
d) 1.2

Answer:  $n = 8, p = 0.3 \Rightarrow np(1-p) = 8 \times 0.3 \times 0.7 = 1.68$

a) 1.68

---

#### Section B: Short Answer Questions

4. Write the probability mass function (PMF) of a Bernoulli distribution and Binomial distribution.

Answer:

Bernoulli:  $P(X=k) = p^k(1-p)^{1-k}$  for  $k \in \{0,1\}$

Binomial:  $P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$  for  $k=0,1,2,\dots,n$

**5. Explain the significance of the parameter  $p$  in Bernoulli and Binomial distributions.**

**Answer:**  $p$  represents the probability of "success" in a single trial.

In Bernoulli,  $p$  is the probability that the random variable is 1 (success).

In Binomial,  $p$  is the probability of success in each of the  $n$  independent trials.

---

### Section C: Python Coding

**6. Generate 10 random samples from a Bernoulli distribution with  $p=0.6$  and print them.**

**Answer:**

```
import numpy as np
```

```
bernoulli_samples = np.random.binomial(1, 0.6, 10)
print("Bernoulli samples:", bernoulli_samples)
```

---

**7. Generate a large sample (10000 points) from a Binomial distribution with  $n=20$ ,  $p=0.3$ . Calculate and print the sample mean and variance.**

**Answer:**

```
import numpy as np
binomial_samples = np.random.binomial(20, 0.3, 10000)
mean = np.mean(binomial_samples)
variance = np.var(binomial_samples)
print("Binomial sample mean:", mean)
print("Binomial sample variance:", variance)
```

## Task-2

### Lab (Week -6)

Sampling data and analyzing distributions: Bernoulli, Binomial (Lab Exercise)

Question 1:

**A factory produces light bulbs, and the probability that a bulb is defective is 0.1. If a quality inspector checks 20 bulbs, find using Python:**

- The probability that exactly 3 bulbs are defective.**
- The probability that at most 2 bulbs are defective.**

### Answer:

```
#Question 1 from scipy.stats import binom,
nbinom # Question 1: Light bulb defective
probabilities p_defective = 0.1 n_bulbs = 20

# a) Probability exactly 3 bulbs defective prob_3_defective =
binom.pmf(3, n_bulbs, p_defective) print(f"a) Probability
exactly 3 defective: {prob_3_defective:.4f}")

# b) Probability at most 2 bulbs defective prob_at_most_2_defective =
binom.cdf(2, n_bulbs, p_defective) print(f"b) Probability at most 2
defective: {prob_at_most_2_defective:.4f}")
```

### Question 2:

**A family with 5 children can be modeled as a Binomial distribution where each child is equally likely to be a boy or a girl (assume probability of boy = probability of girl = 0.5). Out of 800 families with 5 children each, answer the following using Python:**

- a. How many families would be expected to have exactly 3 boys?**
- b. How many families would be expected to have exactly 5 girls?**
- c. How many families would be expected to have either 2 or 3 boys?**
- d. How many families would be expected to have at least 2 girls?**

**Write Python code to compute the answers using the Binomial probability formula. Also, verify the results using Answer:**

```
#Question 2
from scipy.stats import binom

n_children = 5 p_boy
= 0.5
n_families = 800

# a) Expected number of families with exactly 3 boys prob_3_boys
= binom.pmf(3, n_children, p_boy) expected_3_boys = prob_3_boys
* n_families print(f"a) Expected families with exactly 3 boys:
{expected_3_boys:.0f}") # b) Expected families with exactly 5 girls (0
boys) prob_5_girls = binom.pmf(0, n_children, p_boy) expected_5_girls
```

```
= prob_5_girls * n_families print(f"b) Expected families with exactly 5
girls: {expected_5_girls:.0f}")
```

```
# c) Expected families with either 2 or 3 boys prob_2_boys = binom.pmf(2,
n_children, p_boy) expected_2_or_3_boys = (prob_2_boys + prob_3_boys)
* n_families print(f"c) Expected families with 2 or 3 boys:
{expected_2_or_3_boys:.0f}")
```

```
# d) Expected families with at least 2 girls (girls >= 2 means boys <= 3)
prob_at_least_2_girls = binom.cdf(3, n_children, p_boy)
expected_at_least_2_girls = prob_at_least_2_girls * n_families print(f"d)
Expected families with at least 2 girls: {expected_at_least_2_girls:.0f}")
```

### Question 3:

**You are playing a game where the probability of winning (success) in each trial is  $p = 0.3$ .**

- Write a Python program to calculate the probability that there will be exactly 5 failures before achieving 4 successes. (Hint: Use Negative Binomial distribution formula)**
- Write a Python program to calculate the probability of getting exactly 4 wins in 9 trials. (Hint: Use Binomial distribution formula)**
- Write a Python program to calculate the probability that the 4th win occurs exactly on the 9th trial. (Hint: Use Negative Binomial pmf for  $r=4$  and  $k=9-r=5$  failures)**

Answer: #Question

3

```
from scipy.stats import nbinom, binom
```

```
p = 0.3
r = 4 # number of
successes k = 5 # number
of failures
n = 9 # total trials
```

```
# a) Probability exactly 5 failures before 4 successes (Negative Binomial PMF)
prob_neg_binomial_a = nbinom.pmf(k, r, p)
print(f"a) Probability 5 failures before 4 successes: {prob_neg_binomial_a:.4f}")
```

```
# b) Probability exactly 4 wins in 9 trials (Binomial PMF)
prob_binomial_b = binom.pmf(4, n, p)
print(f"b) Probability exactly 4 wins in 9 trials: {prob_binomial_b:.4f}")
```

```
# c) Probability 4th win occurs exactly on 9th trial (Negative Binomial PMF)
prob_neg_binomial_c = nbinom.pmf(k, r, p)
print(f"c) Probability 4th win on 9th trial: {prob_neg_binomial_c:.4f}")
```