IS 606: Statistics and Probability for Data Analytics
Hands-On Laboratory Series
The Poisson Distribution: Fundamentals

Overview

This exercise is designed to give you practice in working with the Poisson distribution.

Prerequisites

You should know the basic concepts of the Poisson distribution, including the essential characteristics, the core formulas that go with the distribution, and how to apply the distribution to answer basic questions.

Materials

This lab exercise is entirely self-contained.

Instructions

This lab exercise is to be completed step by step according to the instructions given. If you are struggling with a particular step, then our recommendation is that you look to the solution ***for only that step*** for help. Once you have sorted out the details of the step in question, proceed to the next task.

1. For most of this lab exercise, we will be interested in a Poisson distribution with parameter $\lambda=2.2$. For review purposes, describe the basics of a Poisson distribution.

   The basics of the Poisson distribution are described in the Dobrow text beginning on page 96. In a nutshell, the distribution models counts of outcomes where there are no prior constraints on the possible numbers of outcomes. We can think of it as breaking a particular time interval into many tiny independent intervals of time in which a single event may or may not occur. (For example, we could break one hour into 3,600 seconds, or even 3,600,000 milliseconds.)

   We choose these small intervals so that it is highly unlikely that two events will occur during a single interval, and then we assume that the probability of an event occurring in a single interval is fixed and independent of other intervals. In practice, however, we prefer to think of the Poisson distribution as having no fixed maximum number of events.

   Together with the binomial distribution, the Poisson distribution is one of the most useful distributions available for modeling real-world phenomena.

2. The Poisson distribution has outcomes that range from 0 to infinity (though the probabilities approach 0 very quickly as X increases far beyond the parameter value). It is therefore impossible to calculate the entire probability distribution in table form. Let's get the first several values, though. Calculate the probability distribution for X ranging from 0 to 8 using the formula:

$$P(X = k) = \frac{e^{-\lambda}\lambda^k}{k!}$$

Here is the table of probabilities, each rounded to four decimal places:

X	0	1	2	3	4	5	6	7	8
P(X)	0.1108	0.2438	0.2681	0.1966	0.1082	0.0476	0.0174	0.0055	0.0015

Notice that this table gives a rough approximation to a probability distribution, since each probability value is non-negative, the values sum to 0.9995 (not 1), and each possible outcome 0 through 8 is assigned a value. Note that the remaining probability of 0.0005 is the probability of getting more than 8 outcomes. We could instead represent this as follows:

X	0	1	2	3	4	5	6	7	8+
P(X)	0.1108	0.2438	0.2681	0.1966	0.1082	0.0476	0.0174	0.0055	0.0020

3. Using your results from the previous step (and ignoring the possibility of X being greater than 8 for now), calculate the expected value of the distribution using the theoretical formula:

$$E(X) = \sum_{i=1}^{n} x_i p(x_i)$$

Using the values in my first table, I get:

$$\boldsymbol{E(X) = (0)(0.1108) + (1)(0.2438) + (2)(0.2681) + (3)(0.1966)}$$
$$\boldsymbol{+(4)(0.1082) + (5)(0.0476) + (6)(0.0174)}$$
$$\boldsymbol{+(7)(0.0055) + (8)(0.0015) = 2.1955}$$

4. Calculate the expected value of the distribution using the Poisson-specific formula:

$$E(X) = \lambda$$

(Yes, this is so easy I cannot believe we are asking you to do it. How does your answer here compare with your answer from the previous part? What accounts for any differences? Is it just rounding error? Or does excluding outcomes greater than 8 have a significant effect?)

Using the formula, we "calculate" that:

$$\boldsymbol{E(X) = 2.2}$$

Note that our answer from the previous step underestimated this value slightly. This is because we did not account for outcomes greater than 8 in our expected value above. (Really, who has time to do infinite calculations?) Outcomes greater than 8 have a small but definitely noticeable effect. (We might also have slight rounding errors…)

5. Using your distribution from step 2 (and again ignoring outcomes greater than 8), calculate the variance and standard deviation of the distribution using the theoretical formulas:

$$Var(X) = \sum_{i=1}^{n}(x_i - E(X))^2 p(x_i) \qquad and \qquad SD(X) = \sqrt{Var(X)}$$

$$\boldsymbol{Var(X) = (0-2.2)^2(0.1108) + (1-2.2)^2(0.2438) + \cdots + (8-2.2)^2(0.0015)}$$

$$\boldsymbol{Var(X) = 2.17608}$$

$$\boldsymbol{SD(X) = \sqrt{2.17608} = 1.475154}$$

6. Calculate the standard deviation of the distribution using the Poisson-specific formula:

$$SD(X) = \sqrt{\lambda}$$

That's almost as easy as the expected value, right? How does your answer here compare with your answer from the previous part? What accounts for any differences?

Here, we have:

$$\boldsymbol{SD(X) = \sqrt{2.2} = 1.48324}$$

Again, by excluding outcomes greater than 8, we are underestimating standard deviation slightly in our calculations above. These outcomes have a noticeable effect. (We could also introduce rounding errors…)

7. Generate a random sample of 2,000 observations from a Poisson distribution with λ=2.2 using the rpois() function. Be sure to set a random seed with set.seed() and assign the resulting observations to a vector named **poissonsample**.

Here is the code that I will use to generate a random sample:

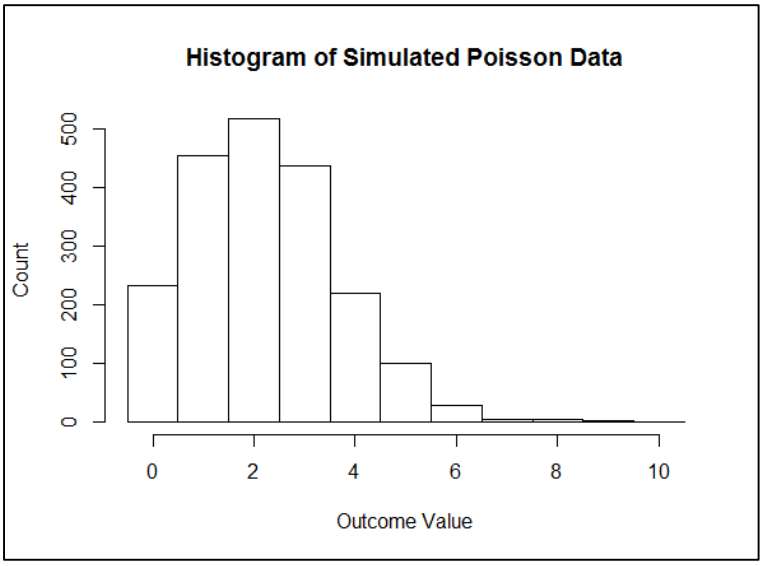
```
set.seed(369)
poissonsample <- rpois(n=2000, lambda=2.2)
```

8. Create a histogram of the simulated data (using breaks=c(-0.5, 0.5, 1.5, 2.5, 3.5, 4.5, 5.5, 6.5, …) so that the breaks extend past your largest observation) and compare it with the probability distribution in step 2. How closely does it match?

Note that for our sample, the maximum value that occurs (found by running the command `max(poissonsample)` at the command line) is 10. I will specify my breaks by generating a sequence instead of just listing the values. Here is the code:

```
hist(poissonsample,
     breaks=seq(from=-0.5, to=10.5, by=1),
     xlab="Outcome Value", ylab="Count",
     main="Histogram of Simulated Poisson Data")
```

The histogram output is given at the top of the next page.



9. Using the sample you drew in step 7 (poissonsample), calculate the simulated mean using the mean() function. How close is it to the theoretical value?

   The code to calculate the mean is simple:

   ```
   mean(poissonsample, na.rm=TRUE)
   ```

   The outputted value is 2.22, which is very close to the theoretical value of 2.2.

10. Using the sample you drew in step 7 (poissonsample), calculate the simulated standard deviation using the sd() function. How close is it to the theoretical value?

    Again, the code is simple:

    ```
    sd(poissonsample, na.rm=TRUE)
    ```

    The outputted value is 1.482801, which is close to the theoretical value of 1.48324.

11. Construct the cumulative distribution function from the probability distribution.

    Here is the table with both the probability distribution P(X) and the cumulative distribution C(X):

X	0	1	2	3	4	5	6	7	8
P(X)	0.1108	0.2438	0.2681	0.1966	0.1082	0.0476	0.0174	0.0055	0.0015
C(X)	0.1108	0.3546	0.6227	0.8193	0.9275	0.9751	0.9925	0.9980	0.9995

12. Obtain the five-number summary from the cumulative distribution you just constructed. (For max, just put ∞.)

    Here is the five-number summary:

    0 – 1 – 2 – 3 – ∞

13. Obtain the same five-number summary from your simulated data using the quantile() function. How closely does it match the theoretical results from the previous step?

    Here is the code:

    ```
    quantile(poissonsample, probs=c(0,0.25,0.5,0.75,1))
    ```

    The output is:

    ```
      0%  25%  50%  75% 100%
       0    1    2    3   10
    ```

    This matches the theoretical results exactly, except of course for the maximum.

Summary

The exercise above walks you through the basics of the Poisson distribution. In an applications lab, we'll apply these and similar concepts to answer real questions and model real scenarios.