

How can the Multinomial Distribution be used in Python?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How can the Multinomial Distribution be used in Python?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165333>

The Multinomial Distribution is a statistical concept that describes the probability of observing a specific combination of outcomes in a series of independent, categorical trials. This distribution can be used in Python by utilizing the built-in functions and packages such as NumPy and SciPy. These tools allow users to calculate the probability mass function and cumulative distribution function of the Multinomial Distribution, as well as generate random samples from the distribution. This can be useful in various applications, such as modeling the outcomes of multiple independent events, predicting the occurrence of different outcomes in a given scenario, and analyzing categorical data. Overall, the Multinomial Distribution in Python provides a powerful tool for understanding and analyzing data with categorical outcomes.

Use the Multinomial Distribution in Python

The describes the probability of obtaining a specific number of counts for k different outcomes, when each outcome has a fixed probability of occurring.

If a X follows a multinomial distribution, then the probability that outcome 1 occurs exactly x_1 times, outcome 2 occurs exactly x_2 times, etc. can be found by the following formula:

$$\text{Probability} = n! * (p_1^{x_1} * p_2^{x_2} * \dots * p_k^{x_k}) / (x_1! * x_2! * \dots * x_k!)$$

where:

n : total number of events
 x_1 : number of times outcome 1 occurs
 p_1 : probability that outcome 1 occurs in a given trial

The following examples show how to use the function in Python to answer different probability questions regarding the multinomial distribution.

Example 1

In a three-way election for mayor, candidate A receives 10% of the votes, candidate B receives 40% of the votes, and candidate C receives 50% of the votes.

If we select a random sample of 10 voters, what is the probability that 2 voted for candidate A, 4 voted for candidate B, and 4 voted for candidate C?

We can use the following code in Python to answer this question:

```
from scipy.stats import multinomial
```

```
#calculate multinomial probability
```

```
multinomial.pmf(x=, n=10, p=)
```

```
0.050400000000000001
```

The probability that exactly 2 people voted for A, 4 voted for B, and 4 voted for C is 0.0504.

Example 2

Suppose an urn contains 6 yellow marbles, 2 red marbles, and 2 pink marbles.

If we randomly select 4 balls from the urn, with replacement, what is the probability that all 4 balls are yellow?

We can use the following code in Python to answer this question:

```
from scipy.stats import multinomial
```

```
#calculate multinomial probability  
multinomial.pmf(x=, n=4, p=)
```

```
0.12959999999999999
```

Example 3

Suppose two students play chess against each other. The probability that student A wins a given game is 0.5, the probability that student B wins a given game is 0.3, and the probability that they tie in a given game is 0.2.

If they play 10 games, what is the probability that player

A wins 4 times, player B wins 5 times, and they tie 1 time?

We can use the following code in Python to answer this question:

```
from scipy.stats import multinomial
```

```
#calculate multinomial probability
```

```
multinomial.pmf(x=, n=10, p=)
```

```
0.03827249999999997
```

The probability that player A wins 4 times, player B wins 5 times, and they tie 1 time is about 0.038.

Additional Resources

The following tutorials provide additional information about the multinomial distribution: