Arden University

Master of Data Science

Mathematics for Data Science

Understanding Conditional Probability and Its Application

Author:
STU230944

Tutor:
Mohammad Amin
Mohammadi Banadaki

November 20, 2024

Word Count: 1108

Headers: 11

Introduction

Conditional probability is a fundamental concept in statistics that quantifies the likelihood of an event occurring, given that another related event has already occurred. The term $P(A \cap B)$ represents the probability that both events A and B occur simultaneously. It is often referred to as the **joint probability** of A and B, while $P(B)$ serves as the **normalizing factor**, ensuring that the conditional probability $P(A|B)$ is expressed relative to the occurrence of event B.

The mathematical representation of conditional probability is:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

This analysis applies conditional probability to evaluate how age influences the likelihood of individuals making a purchase. Two scenarios are analysed:

- **Scenario 1: Weighted Probabilities:** Older individuals are more likely to purchase, with purchasing probabilities increasing with age.
- **Scenario 2: Uniform Probabilities:** All age groups have the same likelihood of making a purchase, removing age as a factor.

The main objective is to critically evaluate the impact of age on purchasing behavior under these scenarios.

Task Interpretation

The task is to generate a data set of 20,000 random purchases for five age groups (25, 35, 45, 55 and 65), where purchase will depend on age. The older the consumer, the greater is the probability of purchasing the item. The study calculates the probability of a person buying an item, given that person is aged 35 years. It repeats the calculation, removing the weight of age, and comparing the findings.

Approach and Implementation

Weighted Probabilities

In this case, purchase probabilities are weighted by age, showing a dependency between age and the likelihood of making a purchase. Older individuals have a higher probability of purchasing, while younger individuals are less likely to do so. For instance, individuals aged 25 have only a 10% chance of making a purchase, while those aged 65 have a 90% chance. This setup real-world scenarios where age often influences purchasing behavior, possibly due to factors like disposable income or consumer preferences.

The code starts with defining the age groups and their respective weights. The age groups are between 25 and 65 in steps of 5 years, and each group is assigned a weight representing the purchase probability. The next step is to create 20,000 random ages with NumPy using these age groups. Purchase status for each person is determined by comparing a random number, ranging from 0 to 1, to the purchase weight corresponding

to that person's age group. If the random number is less than the weight, that person makes a purchase (1); otherwise, not (0). The data is then stored in a Pandas DataFrame.

A sample of the generated data includes age and purchase status variables. The results reflects the probability nature of purchases while maintaining the age-based weighting. For example, people in the 25–30 age group have a considerably lower rate of purchase than those in the 60–65 age group, which demonstrates the increasing probability of purchases associated with increasing age.

```
import numpy as np
import pandas as pd

# Step 1: Create the dataset with 20,000 data points
np.random.seed(0)  # For reproducibility

# Define age groups and weights
age_groups = list(range(25, 66, 5))  # [25, 30, 35, ..., 65]
purchase_weights = [0.1, 0.15, 0.25, 0.35, 0.45, 0.55, 0.7, 0.85,
    0.9]

# Generate random data
ages = np.random.choice(age_groups, size=20000)
purchases = (np.random.rand(20000) <
             np.array([purchase_weights[age_groups.index(age)] for
                 age in ages])).astype(int)

# Create DataFrame
data = pd.DataFrame({'Age': ages, 'Purchase': purchases})

# Display sample data
print(data.head())
```

Sample Data with Weighted Purchases

```
   Age  Purchase
0   50         1
1   25         0
2   40         1
3   40         1
4   60         1
```

Uniform Probabilities

Now, we have removed the effect of age on the probability of purchase. In this case, everyone, despite of age, is assigned a 50% equal probability of making a purchase. This way, it removes any dependence on age and serves as a control to analyse the probabilities without any external influences. Starting the implementation, a new column is added to

the previously generated DataFrame. This column, `Purchase_No_Weight`, is a Bernoulli trial where every individual has an independent 50% chance of purchasing something. This is expressed by the binomial function in NumPy, returning a `1` for a successful trial

(purchase), and 0 if there was no purchase. This scenario makes certain that no imbalance in age will affect the outcome, since the probabilities are equal in all age groups.

A sample of the dataset now includes three columns: age, purchase status under weighted probabilities, and purchase status without weighted probabilities. Results: Purchase rates are roughly equal across every age group, confirming the elimination of age dependency in this scenario. This sets up a baseline comparison for introducing or removing outside factors such as age to purchasing behavior.

```
# Step 2: Assign uniform probabilities (50% chance for all ages)
data['Purchase_No_Weight'] = np.random.binomial(1, 0.5, size
    =20000)

# Display sample data
print(data.head())
```

Sample Data with Uniform Purchases

	Age	Purchase	Purchase_No_Weight
0	50	1	1
1	25	0	0
2	40	1	1
3	40	1	0
4	60	1	1

Calculating Conditional Probabilities with Weighted Probabilities

This following code calculates the conditional probability P(Purchase—Age) for each age group according to the weighted probability. This loop goes through row by row for each age in the predefined age groups, filtering the data to include only rows that had an Age equal to the current age group being looked at with `data[data['Age'] == age]`, effectively extracting the data that corresponds to that age group.

On the filtered data, the conditional probability is simply calculated using the mean over the Purchase column. Since the purchases are encoded as binary values-1 for purchase and 0 for no purchase. the mean here is the share of people in that age group who purchased it, which is the conditional probability.

The results are displaying the age group with the corresponding conditional probability under the weighted purchase scenario. This makes it easier to understand how the likelihood of making a purchase changes with age, providing valuable insights based on the predefined weights.

```
# Calculate conditional probability with weighted probabilities
print("\nConditional Probabilities with Weighted Probabilities:")
for age in age_groups:
    # Filter data for the specific age group
    age_data = data[data['Age'] == age]
    # Calculate P(Purchase | Age=age) with weights
    p_given_age_weighted = age_data['Purchase'].mean()
```

```
    # Print the result
    print(f"Age {age}: P(Purchase | Age={age}) with weight = {
        p_given_age_weighted:.4f}")
```

```
Conditional Probabilities with Weighted Probabilities:
Age 25: P(Purchase | Age=25) with weight = 0.0954
Age 30: P(Purchase | Age=30) with weight = 0.1511
Age 35: P(Purchase | Age=35) with weight = 0.2363
Age 40: P(Purchase | Age=40) with weight = 0.3525
Age 45: P(Purchase | Age=45) with weight = 0.4566
Age 50: P(Purchase | Age=50) with weight = 0.5573
Age 55: P(Purchase | Age=55) with weight = 0.7072
Age 60: P(Purchase | Age=60) with weight = 0.8425
Age 65: P(Purchase | Age=65) with weight = 0.9004
```

Calculating Conditional Probabilities without Weighted Probabilities

This section of the code calculates the conditional probability $P(\text{Purchase}|\text{Age})$ for each age group without weight probability case, where the likelihood of making a purchase is independent of age. The loop goes through each age in the predefined `age_groups` and filters the dataset to include only rows where the `Age` matches the current age group. This filtering is done using `data[data['Age'] == age]`, effectively isolating the data for the specific age group.

Once the data is filtered, the conditional probability is calculated as the mean of the `Purchase_No_Weight` column. In this column, purchases are assigned randomly with a uniform 50% probability, ensuring that the results remain same across all age groups and eliminating any dependency on age. The results are showing age and the corresponding conditional probability. This approach provides a neutral baseline for analysing purchasing behavior without the influence of age.

```
# Calculate conditional probability without weighted
    probabilities
print("\nConditional Probabilities with Uniform Probabilities:")
for age in age_groups:
    # Filter data for the specific age group
    age_data_no_weight = data[data['Age'] == age]
    # Calculate P(Purchase | Age=age) without weights
    p_given_age_no_weight = age_data_no_weight['
        Purchase_No_Weight'].mean()
    # Print the result
    print(f"Age {age}: P(Purchase | Age={age}) without weight = {
        p_given_age_no_weight:.4f}")
```

```
Conditional Probabilities with Uniform Probabilities:
Age 25: P(Purchase | Age=25) without weight = 0.5029
Age 30: P(Purchase | Age=30) without weight = 0.4973
Age 35: P(Purchase | Age=35) without weight = 0.5018
```

```
Age 40: P(Purchase | Age=40) without weight = 0.4985
Age 45: P(Purchase | Age=45) without weight = 0.4963
Age 50: P(Purchase | Age=50) without weight = 0.5051
Age 55: P(Purchase | Age=55) without weight = 0.4972
Age 60: P(Purchase | Age=60) without weight = 0.5046
Age 65: P(Purchase | Age=65) without weight = 0.4998
```

Results and Analysis

The conditional probability $P(\text{Purchase}|\text{Age})$ was calculated for both scenarios.

Scenario 1: Weighted Probabilities

Purchasing likelihood increases significantly with age:

- **Age 25:** $P(\text{Purchase} \mid \text{Age} = 25) = 0.0954$
- **Age 35:** $P(\text{Purchase} \mid \text{Age} = 35) = 0.2363$
- **Age 65:** $P(\text{Purchase} \mid \text{Age} = 65) = 0.9004$

This highlights a strong correlation between age and purchasing probability, aligning with real-world patterns where older individuals often have higher purchasing power.

Scenario 2: Uniform Probabilities

Purchasing likelihood remains constant across all age groups:

- **Age 25:** $P(\text{Purchase} \mid \text{Age} = 25) = 0.5029$
- **Age 35:** $P(\text{Purchase} \mid \text{Age} = 35) = 0.5018$
- **Age 65:** $P(\text{Purchase} \mid \text{Age} = 65) = 0.4998$

This scenario removes the influence of age, ensuring equal chances of purchasing for all age groups.

Conclusion

This analysis demonstrates the role of conditional probability in studying demographic factors such as age. Weighted probabilities provide a realistic view of purchasing behavior, reflecting real-world patterns, while uniform probabilities offer unbiased insights. Both approaches are valuable depending on the context weighted probabilities for targeted strategies and uniform probabilities for generalized predictions. Future studies could extend this approach to include other factors like income, gender, or regional differences.

References

Hurwitz, J. S., Nugent, A. & Halper, F. (2020), *Big Data For Dummies*, 2nd edn, Wiley.

Lynch, S. (2020), *Python for Scientific Computing and Artificial Intelligence*, Springer.

McKinney, W. (2017), *Python for Data Analysis*, 2nd edn, O'Reilly Media, Inc.

Mueller, J. P. (2017), *Beginning Programming with Python For Dummies*, 2nd edn, Wiley.

Sharda, R., Delen, D. & Turban, E. (2021), *Systems for Analytics, Data Science, & Artificial Intelligence: Systems for Decision Support*, 11th edn, Pearson.

McKinney (2017) Hurwitz et al. (2020) Mueller (2017) Sharda et al. (2021) Lynch (2020)