

Quiz6

November 16, 2022

1 Rules

- Internet and Books are ALLOWED
- Name your file as following: StudentID_ChineseName/EnglishName_quiz#
- Extension of your file or your file type should be .py .
- If MOSS (Measure of Software Similarity) detects that any two files have more than 50% similarity, both students will get 0 for this quiz (We give it a base code that everyone has, so no need to worry about that)
- If you submit the code within an hour you will get the full score. Otherwise, you will have 24 hours to finish it and you will get 80% of your final score.

2 Accept/Reject Algorithm

2.1 1.Generate a random variable $Y \sim \text{beta}(3.2, 5.6)$ by Accept/Reject algorithm

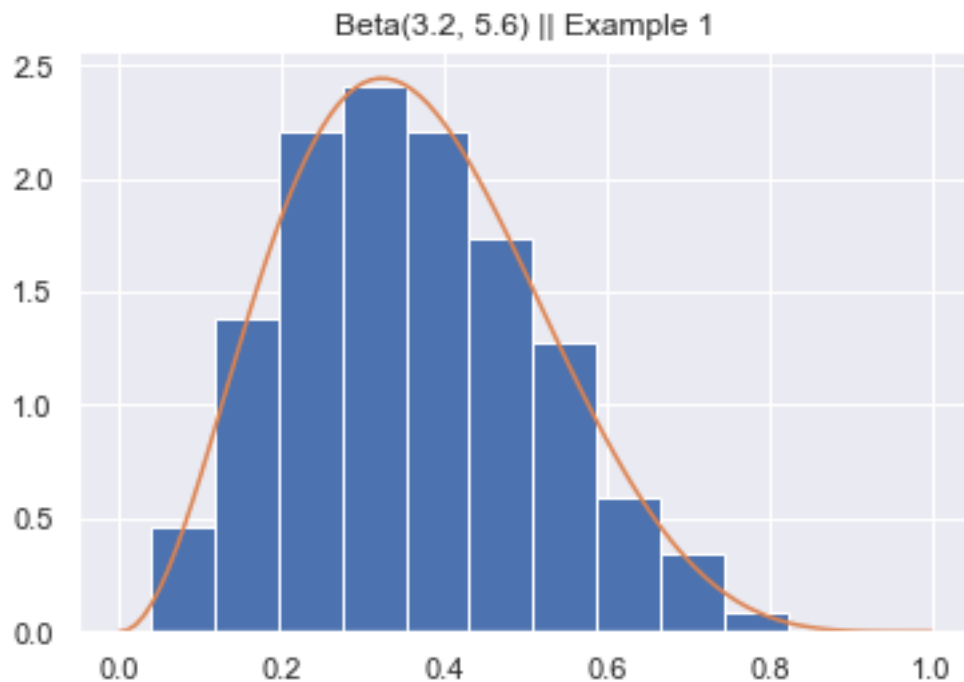
Step1. Generate (U, V) independent uniform(0,1)

Step2. If

$$U < \frac{1}{c}f_Y(V)$$

set $Y = V$; otherwise return to step1. Where $c = \max_y f_Y(y)$

Example plot



2.2 2.Generate a random variable $Y \sim \text{beta}(3.2, 5.6)$ by Accept/Reject algorithm

Step1. Generate $U \sim \text{uniform}(0,1)$, $V \sim \text{beta}(2,6)$

Step2. If

$$U < \frac{1}{M} \frac{f_Y(V)}{f_V(V)}$$

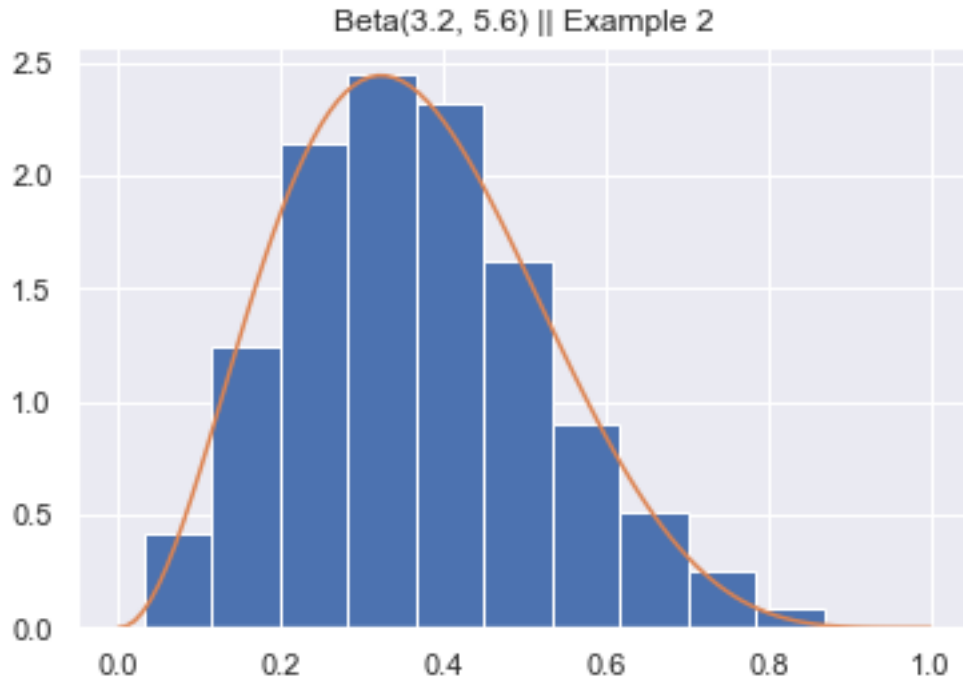
set $Y = V$; otherwise return to step1. Where $M = \sup_y \frac{f_Y(y)}{f_V(y)}$

$$Y = \frac{\sum_{j=1}^a \log(U_j)}{\sum_{j=1}^{a+b} \log(U_j)} \sim \text{beta}(a, b)$$

print M after you calculate it

-Hint you need to find M by yourself

Example plot



3. Some probability (In this section, round all your answers to the 3rd decimal point)

3.1 Probability

The definition of “probability”, [can be surprisingly difficult](#).

To avoid getting stuck before we start, we will use a simple definition for now and refine it later:

A **probability** is a fraction of a finite set.

For example, if we survey 1000 people, and 20 of them are bank tellers, the fraction that work as bank tellers is 0.02 or 2%. If we choose a person from this population at random, the probability that they are a bank teller is 2%. By “at random” I mean that every person in the dataset has the same chance of being chosen.

With this definition and an appropriate dataset, we can compute probabilities by counting. To demonstrate, We’ll use data from the [General Social Survey](#) (GSS).

```
[2]: import pandas as pd

gss = pd.read_csv('gss.csv', index_col=0)
gss.head()
```

```
[2]:      year  age  sex  polviews  partyid  indus10
caseid
1      1974  21.0   1      4.0      2.0    4970.0
2      1974  41.0   1      5.0      0.0    9160.0
5      1974  58.0   2      6.0      1.0    2670.0
```

6	1974	30.0	1	5.0	4.0	6870.0
7	1974	48.0	1	5.0	4.0	7860.0

The `DataFrame` has one row for each person surveyed and one column for each variable I selected.

The columns are

- `caseid`: Respondent id (which is the index of the table).
- `year`: Year when the respondent was surveyed.
- `age`: Respondent's age when surveyed.
- `sex`: Male or female.
- `polviews`: Political views on a range from liberal to conservative.
- `partyid`: Political party affiliation, Democrat, Independent, or Republican.
- `indus10`: Code for the industry the respondent works in.

Let's look at these variables in more detail, starting with `indus10`.

3.2 Fraction of Bankers

The `code` for “Banking and related activities” is **6870**.

Subset the dataset that will only include “Banking and related activities” or simply Bankers

```
[ ]: banker = gss[___] == ___
      banker.head()
```

The result is a Pandas `Series` that contains the Boolean values `True` and `False`.

Now find the total number of bankers.

```
[ ]: banker.---
```

In this dataset, there are (Write your answer here) bankers.

Now you need to compute the *fraction* of bankers

Hint: you need compute the fraction of `True` values in the `Series`:

```
[ ]: banker.---
```

About (Write your answer here. Result should be in percentage)% of the respondents work in banking, so if we choose a random person from the dataset, the probability they are a banker is about (Your answer)%.

3.3 The Probability Function

Complete the code. The function that takes a Boolean series and returns a probability:

```
[9]: def prob(A):
      """Computes the probability of a proposition, A."""
```

```
return A.---
```

So we can compute the fraction of bankers like this:

```
[ ]: prob(banker)
```

Now let's look at another variable in this dataset. The values of the column **sex** are encoded like this:

```
1    Male
2    Female
```

So we can make a Boolean series that is **True** for female respondents and **False** otherwise.

```
[ ]: female = (gss[___] == ___)
```

And use it to compute the fraction of respondents who are women.

```
[ ]: prob(female)
```

3.4 Political Views and Parties

The other variables we'll consider are **polviews**, which describes the political views of the respondents, and **partyid**, which describes their affiliation with a political party.

The values of **polviews** are on a seven-point scale:

```
1    Extremely liberal
2    Liberal
3    Slightly liberal
4    Moderate
5    Slightly conservative
6    Conservative
7    Extremely conservative
```

We'll define **liberal** to be **True** for anyone whose response is "Extremely liberal", "Liberal", or "Slightly liberal".

```
[ ]: liberal = (gss[___] <= 3)
```

Here's the fraction of respondents who are liberal by this definition.

```
[ ]: prob(liberal)
```

If we choose a random person in this dataset, the probability they are liberal is about (Write your answer here)%.

The values of **partyid** are encoded like this:

```
0    Strong democrat
1    Not strong democrat
2    Independent, near democrat
```

```
3 Independent
4 Independent, near republican
5 Not strong republican
6 Strong republican
7 Other party
```

We'll define `democrat` to include respondents who chose "Strong democrat" or "Not strong democrat":

```
[ ]: democrat = (gss[___] == 1)
```

And here's the fraction of respondents who are Democrats, by this definition.

```
[ ]: prob(democrat)
```

3.5 Conjunction

Now that we have a definition of probability and a function that computes it, let's move on to conjunction.

"Conjunction" is another name for the logical `and` operation. If you have two [propositions](#), A and B, the conjunction A `and` B is `True` if both A and B are `True`, and `False` otherwise.

```
[ ]: prob(banker)
```

And the probability that they are a Democrat:

```
[ ]: prob(democrat)
```

Now compute the probability that a respondent is a banker *and* a Democrat:

```
[ ]: prob(___)
```

Remember conjunction is commutative; that is, A `and` B should be the same as B `and` A.

3.6 Conditional Probability

Conditional probability is a probability that depends on a condition, but that might not be the most helpful definition. Here are some examples:

- What is the probability that a respondent is a Democrat, given that they are liberal?
- What is the probability that a respondent is female, given that they are a banker?
- What is the probability that a respondent is liberal, given that they are female?

3.6.1 Let's start with the first one, which we can interpret like this: "Of all the respondents who are liberal, what fraction are Democrats?"

We can compute this probability in two steps:

1. Select all respondents who are liberal. (You already have this)

2. Compute the fraction of the selected respondents who are Democrats.

To select liberal respondents, we can use the bracket operator, `[]`, like this:

```
[ ]: selected = ___
```

`selected` contains the values of `democrat` for liberal respondents, so `prob(selected)` is the fraction of liberals who are Democrats:

```
[ ]: prob(selected)
```

How many liberals are Democrats? Write your answer here in percentage.

Let's try the second example,

"What is the probability that a respondent is female, given that they are a banker?" ## We can interpret that to mean, "Of all respondents who are bankers, what fraction are female?"

Again, we'll use the bracket operator to select only the bankers and `prob` to compute the fraction that are female.

```
[ ]: selected = ___  
     prob(selected)
```

About (Write your answer here)% of the bankers in this dataset are female.

Let's wrap this computation in a function.

We'll define `conditional` to take two Boolean series, `proposition` and `given`, and compute the conditional probability of `proposition` conditioned on `given`:

```
[ ]: def conditional(proposition, given):  
     """Probability of A conditioned on given."""  
     return prob(___)
```

We can use `conditional` to compute the probability that a respondent is liberal given that they are female.

```
[ ]: conditional(___, given=___)
```

About (Write your answer here)% of female respondents are liberal.

We include the keyword, `given`, along with the parameter, `female`, to make this expression more readable. However, it is not necessary

3.7 Conditional Probability Is Not Commutative

We have seen that conjunction is commutative; that is, `prob(A and B)` is always equal to `prob(B and A)`.

But conditional probability is *not* commutative; that is, `conditional(A, B)` is not the same as `conditional(B, A)`.

That should be clear if we look at an example.

Compute the probability a respondent is female, given that they are banker.

```
[ ]: conditional(___, given=___)
```

The result shows that the majority of bankers are female.

Compute the probability that a respondent is a banker, given that they are female:

```
[ ]: conditional(___, given=___)
```

Only about (Write your answer here)% of female respondents are bankers.

I hope this example makes it clear that conditional probability is not commutative, and maybe it was already clear to you. Nevertheless, it is a common error to confuse `conditional(A, B)` and `conditional(B, A)`.

3.8 Condition and Conjunction

We can combine conditional probability and conjunction.

Compute the probability a respondent is female, given that they are a liberal Democrat.

```
[ ]: conditional(___, given=___)
```

About (Write your answer here)% of liberal Democrats are female.

3.8.1 Now compute the probability they are a liberal female, given that they are a banker:

```
[ ]: conditional(___, given=___)
```

About (Write your answer here)% of bankers are liberal women.

4 Congratulations. You are done with your quiz and now you know more about probability and how to compute in using Python.