

Lab 2 - Probability

February 5, 2020

1 Preamble

As in Lab 1, you will find your personalised Lab exercises on Brightspace, under Assessment → Quizzes → Lab 2. You will find help on how to answer the question in this document.

In this week's lab we will need the following packages. Make sure, to import in before you start.

```
[ ]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Tip 1: You can run a section of code in your Jupyter Notebook by pressing **Shift** and **Return**.

Tip 2: Google is your friend. If you don't know what a particular function does, or don't know what function to use for a particular task, you will typically find the answer on Google.

The deadline for submitting your Quiz attempt and Python file is **Monday February 17th at 12pm**.

2 Dice-rolling

2.1 Two dice

Suppose we roll two dice, one red and one blue, and sum the two numbers that are rolled. The result can be anywhere between two and twelve, but all results are not equally likely. The below code makes a dataframe where each row is a possible outcome. The first column is the number on the red die, the second the number on the blue die and the third is the total. Can you figure out how this code works? You may want to print the dataframe to see what it looks like.

```
[ ]: red = np.repeat(np.arange(1,7), 6)
blue = np.tile(np.arange(1,7), 6)
dice_roll = pd.DataFrame({'Red':red, 'Blue':blue, 'Total':red+blue})
```

We can use the `value_counts` function to determine the unique values in a column and how many times each value appeared. The output is ordered from most occurrences to least occurrences. Can you figure out how to order the output by the values rather than the counts?

```
[ ]: print(dice_roll.Total.value_counts())
```

Suppose we roll the pair of dice 25 times and write down the sum each time. This would be tedious, but would you agree that picking a number at random from the `Total` column is the same thing as rolling a pair of dice?

To choose a random entry in an array, we will need Numpy's `random` module. Functions from this module are called with the prefix `np.random`.

The code below chooses a random entry for the `Total` column. This corresponds to the results of rolling the pair of dice once. Can you figure out how to change the code to simulate rolling the pair of dice 25 times?

```
[ ]: print(np.random.choice(dice_roll.Total))
```

Now consider rolling the dice 5,000 times. Compute the number of occurrences of each result. You will need convert your array of experiment results to a dataframe if you want to use the `value_counts` function. Note that there are many ways to compute the occurrences, and you are free to use any method you like.

```
[ ]: results = # Insert code for creating results of 5,000 dice rolls here #  
results_df = pd.DataFrame({'Results':results})
```

How would you graphically represent the experimental results?

When simulating random events or choosing random numbers each time we run the code we get a different result. To avoid this, and obtain reproducible results, we set a random seed. In Python, the random seed is used as a base to generate a random number. If we don't set a seed value, the current time is used. Hence, each time you run the code you will get a different answer. To set the seed value yourself and create reproducible results use `np.random.seed`.

What value do you get when you run the code below? Is it the same when you run it a second time?

Try changing the seed value (101) to something else. Does the value change?

```
[ ]: np.random.seed(101)  
print(np.random.choice(dice_roll.Total))
```

Now repeat the experiment with 5,000 dice rolls with a seed and observe if the number of times each result occurs changes.

2.2 Four dice

The below code creates a table for the possible outcomes of 4 dice. Can figure out how it works?

```
[ ]: four_dice = np.zeros([pow(6,4),5])  
n = 0  
for i in range(1,7):  
    for j in range(1,7):  
        for k in range(1,7):  
            for l in range(1,7):
```

```

four_dice[n,:] = [i,j,k,l,i+j+k+l]
n +=1
four_dice_df = pd.DataFrame(four_dice,columns=('1','2','3','4','Total'))

```

What are the possible totals? How many ways can you make up each of these totals?

Simulate rolling the 4 dice 10, 100, 1000 and 10000 times. How does the distribution change the more times you run the experiment?

3 Additional non-assessed exercises

3.1 Twenty-One

The objective of card game Twenty-One, also known as Blackjack, is to obtain a set of cards summing to 21, or as close to 21 as possible. If a player's cards sum to more than 21 they are out. In the first deal, each player is given two cards. Then in subsequent deals, the players choose whether or not they want more cards. A good player will calculate the probability of getting a card that increases their hand passed 21 when deciding whether or not to take another card.

Face cards (Jack, Queen and King) are worth 10 and the Ace can be used as either a 1 or an 11. An Ace should be used an 11, unless doing so increases to hand over 21.

1. Write a piece of code to create a table of possible outcomes for the first deal (two cards). Remember that the cards are not returned to the deck after being dealt to you.
2. What is the probability of getting 21 on the first deal?
3. Write a piece of code to create a table of possible outcomes for the second deal (three cards). Assume that you play on regardless of how close you were to 21 after the first deal.
4. What is the probability of going bust (cards sum to more than 21) on the second deal?

3.2 Airline tickets

An airline company runs a flight that has 200 seats. Each passenger who buys a ticket has a probability p of turning up for the flight. These events are independent. 1. The airline sells 200 tickets. What is the expected number of passengers, if $p = 0.9$? 2. How many tickets should the airline sell to ensure that the expected number of passengers is greater than 200, if $p = 0.7$?

Hint: Write a quick program that computes the expected value of passengers that turn up for each the number of tickets sold, then search the number of tickets sold.

[]: