

## EXERCISE SET 1

### 1. Exercises

**Exercise 1** (Working with probability distributions and modeling): The first two problems are inspired by those in section 2 of [2]. You should look there for more practice.

(a) Suppose that

$$Y \sim \text{Bernoulli}(q)$$

and let  $Z = 1/(1 + Y) + Y$ . What is the probability function of  $Z$ ?

- (b) Suppose a coin is flipped. If the coin is heads, we write down 0. If the coin is tails, we roll a dice and write down the number. Write down the probability distribution for  $Y$ , the number that we write down.
- (c) For the previous problem, conditioned on the dice rolling a 4, what is the probability we write down 0? Conditioned on the coin being tails, what is the probability the dice rolls a 3?
- (d) Consider the geometric distribution discussed in lecture. What are 3 examples of variables in the real world for which this might be a good model and what are some limitations of these models.

**Exercise 2** (Working with nested for loops): Consider the following code:

```
> for i in range(5):
>   for j in range(i+1):
>     print(i,end='')
>   print("")
```

prints out

```
> 0
> 11
> 222
> 3333
> 44444
```

Modify this code to print

```
> 0
> 01
> 012
> 0123
> 01234
> 012345
```

**Exercise 3** (Washington post data): Below I load some data on homicide victims in US from the washington post. Don't worry about how I process it, all you need to work with is the DataFrame "data" on the very last line.

```
> data = pd.read_csv("https://raw.githubusercontent.com/washingtonpost
> /data-homicides/master/homicide-data.csv",encoding = "ISO-8859-1")
> data["victim_age"] = pd.to_numeric(data["victim_age"],errors="coerce")
```

- (a) For each age  $a = 1, \dots, 100$  determine the number of victims  $n(a)$  with an age  $< a$ . You can ignore the effects of those entries with missing ages. Make a plot of  $n(a)$  vs.  $a$ .
- (b) What do you notice about the plot. Does it have the expected behavior?
- (c) Break the data up into white and non-white victims. Then, for each group make the plot from part (a). Comment on what you find.

**Exercise 4** (Getting a sequence of wins): Let  $J$  denote a random variable representing the number of times a fair coin is flipped before two heads appear in a row. As we saw in class, the following code generates simulations of  $J$ :

```

> def flip_until_two():
>     num_heads = 0
>     total_flips = 0
>     while num_heads < 2:
>         y = np.random.choice([0,1])
>         if y == 0:
>             num_heads = 0
>         else:
>             num_heads = num_heads + 1
>         total_flips = total_flips + 1
>     return total_flips

```

- By changing the code above, write a function `flipuntiltwo()` that rolls a dice until we get  $n$  ones in a row. You should change the variable names accordingly. We will call this random variable `FlipTwo( $n$ )`.
- Make a DataFrame where each column represents a value of  $n$  from 1 to 6 and each row is a simulation from the model `FlipTwo( $n$ )`. There should be 100 rows.
- Make a plot of the maximum and minimum values of  $J$  as a function of  $n$  on the same plot. You might notice one of these increases much faster than the other.

**Exercise 5** (Two gene model): Consider the variant of the model discussed in class:

$$\mathbb{P}(Y_A, Y_B) = \begin{cases} 1/3 & \text{if } Y_A = 0 \text{ and } Y_B = 0 \\ 1/3 & \text{if } Y_A = 0 \text{ and } Y_B = 1 \\ 1/6 & \text{if } Y_A = 1 \text{ and } Y_B = 0 \\ 1/6 & \text{if } Y_A = 1 \text{ and } Y_B = 1 \end{cases}$$

- What are the marginal distributions of  $Y_A$  and  $Y_B$ ?
- Are  $Y_A$  and  $Y_B$  independent?
- Confirm your answer with simulations.

**Exercise 6** (Verifying variance formula for Bernoulli variable): Verify the formula for the variance

$$\text{Var}(Y) = q(1 - q)$$

Remember, you can do this you can use the fact that pointwise arithmetic between numpy arrays can be performed directly on the ways, e.g.

```
> q_range*q_range
```

makes a list where every element is the corresponding element of `q_range` squared. You should experiment to ensure you are using enough samples.

**Exercise 7** (Working with Washington Post Data): This is a continuation of Exercise 3. Consider the quantities

$$P(\text{age} < z)$$

$$P(\text{age} < z | \text{white})$$

$$P(\text{age} < z | \text{not white}).$$

- Explain what each of these is related to the plot you made in Exercise 3.
- Make plots of them and comment on the difference between the plot in Exercise 3. Do you think age and race are independent based on these plots?
- Using the data, approximate,

$$P(\text{white} | 10 < \text{age} < 60)$$

Hint: One way to do this is to use Bayes' rule

**Exercise 8** (Covid modeling – **ungraded**): Suppose we are interested in modeling how likely we are to contract covid after a night out. Imagine that you interact with  $N$  people. Let  $Y_i$  represent whether or not the  $i$ th person you interacted with has covid and  $T_i$  represent whether or not you contract covid from the interaction with the  $i$ th person.

Our model is as follows:

$$Y_i \sim \text{Bernoulli}(1/10)$$

$$T_i | (Y_i = 1) \sim \text{Bernoulli}(1/2)$$

$$T_i | (Y_i = 0) \sim \text{Bernoulli}(0)$$

- (a) What is the distribution  $T_i$  NOT conditioned on  $Y_i$ . That is, what is the marginal distribution of  $T_i$ ?
- (b) Fill in the question marks in the following function so that it simulates whether or not you got covid from the night out; that is, so it returns 1 if you got covid and 0 if you didn't.

```
> def sim_covid(n):
>     got_covid = 0
>     for k in range(n):
>         got_covid_interaction = ???
>         if got_covid_interaction == 1:
>             got_covid = 1
>     return got_covid
```

- (c) Confirm with Monte Carlo simulations simulations that the probability of getting covid from the entire night out is

$$(1) \quad P(\text{get covid}) = 1 - \left(1 - \frac{1}{20}\right)^n$$

You should make a plot of this probability vs.  $n$ , similar to what we did for the Bernoulli distribution in the class notebook.

#### References

- [1] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, et al. *An introduction to statistical learning (python version)*, volume 112. Springer, 2013.
- [2] John Tabak. *Probability and statistics: The science of uncertainty*. Infobase Publishing, 2014.