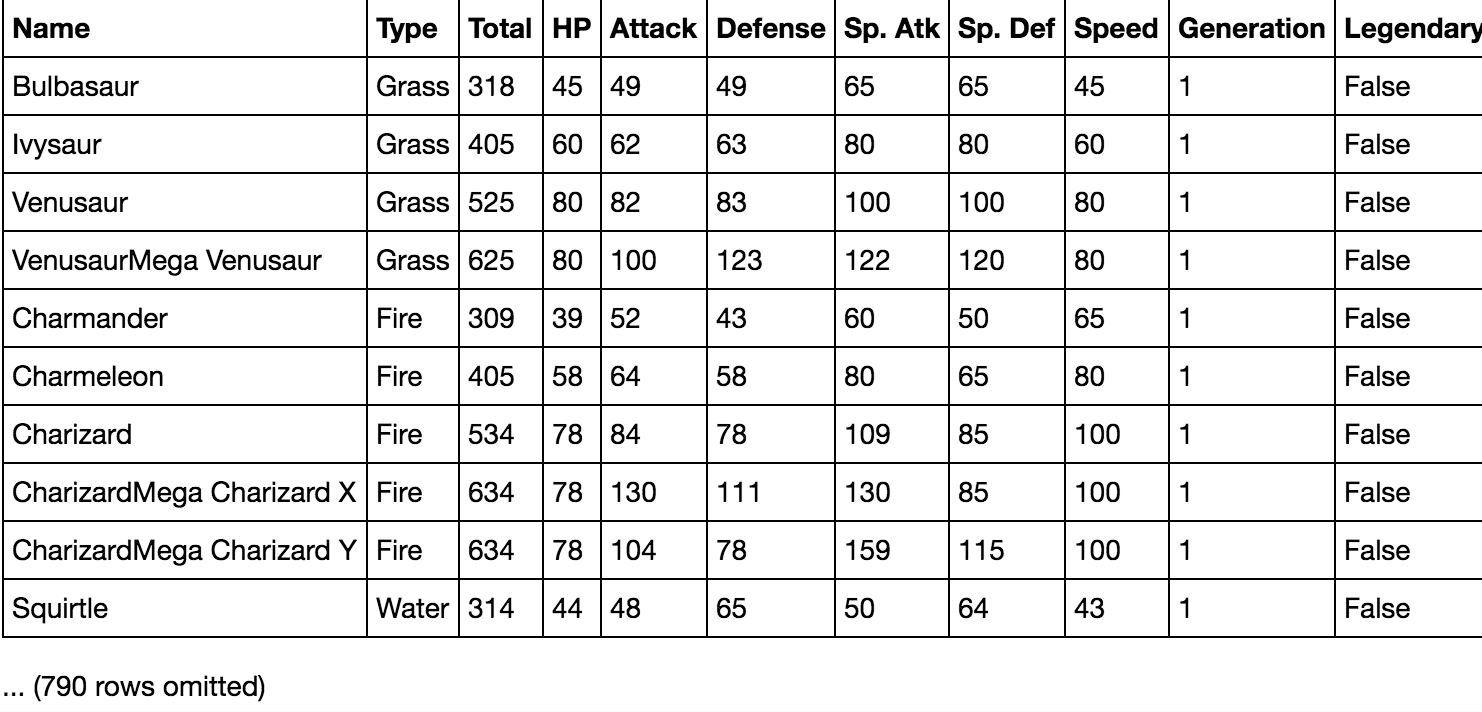
**Data 8 Spring 2020**

**Discussion: Midterm Review**

**Tables**

You are given the following table called pokemon. For the following questions, fill in the blanks.



1. Find the name of the pokemon of type Water that has the highest HP.

water\_pokemon = pokemon.\_\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

water\_pokemon.\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_).column("Name").item(0)

water\_pokemon = pokemon.where("Type", are.equal\_to("Water"))

water\_pokemon.sort("HP",descending=True) .column("Name").item(0)

2. Find the proportion of pokemon of type Fire in the dataset whose Speed is strictly less than 100.

fire\_pokemon = pokemon.\_\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

fire\_pokemon.\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_).\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

fire\_pokemon = pokemon.where("Type", "Fire")

fire\_pokemon.where("Speed", are.below(100)).num\_rows/fire\_pokemon.num\_rows

3. Return a table containing Type and Generation that is sorted in decreasing order by the average HP for each pair of Type and Generation.

d = pokemon.\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

d.sort("HP mean",\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_).\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_)

d = pokemon.group(["Type", "Generation"], np.mean)

d.sort("HP mean", descending=True).select("Type", "Generation")

4. Find the largest difference of average HP between consecutive generations of Pokemon

generation = pokemon.\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_)\

.sort("Generation",descending=False)

\_\_\_\_\_\_\_\_(np.diff(\_\_\_\_\_\_\_\_\_\_\_\_\_\_.\_\_\_\_\_\_\_\_\_("HP mean")))

generation = pokemon.group("Generation",np.mean) \ .sort("Generation", descending=False)

np.max(np.diff(generation.column("HP mean"))) #

5. Return an array that contains ratios of legendary to non-legendary pokemons for each generation.

t = pokemon.\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

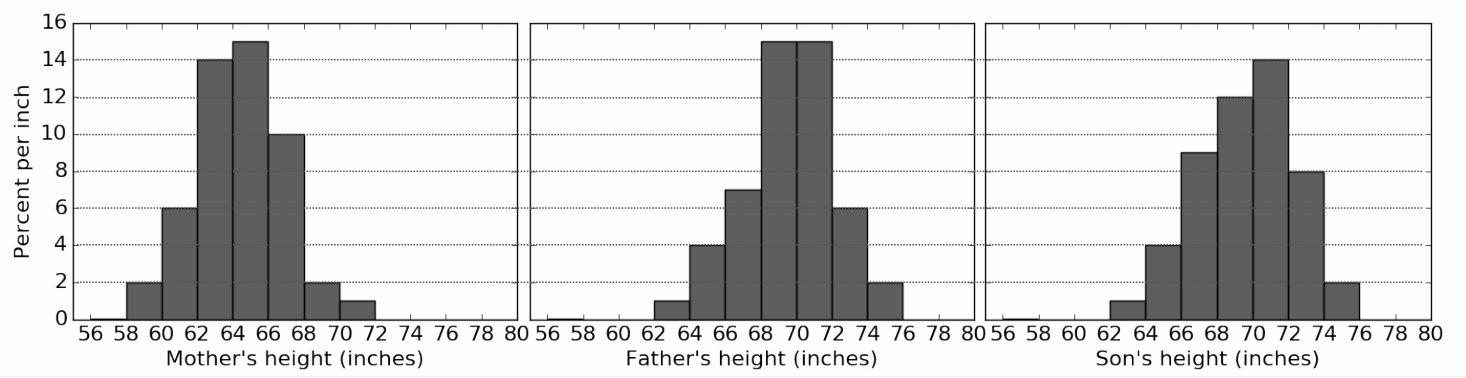
ratio = t.\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_)/t.\_\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_)

t = pokemon.pivot("Legendary", "Generation")

ratio = t.column("True")/t.column("False")

**Histograms**

Galton measured the heights of the members of **200 families** that each included 1 mother, 1 father, and some varying number of adult sons. The three histograms of heights below depict the distributions for all mothers, fathers, and adult sons. All bars are 2 inches wide. All bar heights are integers. The heights of all people in the data set are included in the histograms.



1. Calculate each quantity described below or write *Unknown* if there is not enough information above to express the quantity as a single number (not a range). Show your work!

a. The **percentage** of mothers that are at least 58 inches but less than 62 inches tall.

(58-62): 2 inches \* (6 + 2) percent/inch = 16 percent

b. The **percentage** of fathers that are at least 62 inches but less than 65 inches tall.

Unknown: We can’t tell how heights are distributed within a bin.

c. The **number** of sons that are at least 72 inches tall.

Unknown: The total number of sons is unknown, so the size of any subset is unknown.

d. The **number** of mothers that are at less than 70 inches tall.

(100 percent - (2 inches \* 1 percent/inch)) \* 200 mothers = 196 mothers

2. If the father’s histogram were redrawn, replacing the three bins from 68-70, 70-72 and 72-to-74 with one bin from 68-to-74, what would be the height of its bar? If it’s impossible to tell, write *Unknown*.

The bin contains 15 \* 2 + 15 \* 2 + 6 \* 2 = 72 percent, and the width is 6 inches, so the height is 12 percent/inch.

3. The percentage of sons that are taller than all of the fathers is between 0% and 4%. Fill in the blanks in the previous sentence with the smallest range that can be determined from the histograms, then explain your answer below.

The tallest father is between 74 and 76 inches. The proportion of sons above 76 inches is 0. The proportion of sons above 74 inches is 2 percent/inch \* 2 inches = 4 percent.

**Probability**

1. A fair coin is tossed five times. Two possible sequences of results are HTHTH and HTHHH. Which sequence of results is more likely? Explain your answer and calculate the probability that each sequence appears.

They are equally likely since the coin is fair. By the multiplication rule, the probability that either of the two sequences appears is (1/2)\*\*5.

2. Consider a biased coin such that the probability of getting heads is ⅕ and the probability of getting tails is ⅘ . The coin is tossed 3 times. What is the probability that you get exactly 2 heads?

Here we need to enumerate all the outcomes that fall into this event and count their probabilities. The outcomes are HHT, HTH, THH. There is a total of 3 events. The probability for each of them is (⅕)^2 \* (⅘)^1. So the final result then will be 3 \* (⅕)^2 \* (⅘)^1.

3. Once again, we toss the same coin 3 times. What is the probability I get no heads?

(⅘)^3

4. Again, we toss the same coin 3 times. What is the probability I get at least 1 heads?

*Hint: There are two ways of calculating this probability. One is significantly easier to calculate than the other.*

1 - (⅘)^3

**Simulation and Hypothesis Testing**

Achilles the turtle sits on the number line. Achilles loves long random walks that last a total of 100 times steps. At each time step, Achilles moves based on the following scheme: He flips a coin and moves one step to the right if the coin comes up heads or one step to the left if the coin comes up tails.



1. Assuming that Achilles’ coin is fair, write a function called one\_walk that simulates one random walk of 100 time steps and returns how far from the origin Achilles ends up at the end of his walk. You may assume that Achilles always starts from the origin.

def one\_walk():

dist = 0

for i in np.arange(100):

move = np.random.choice(make\_array(1, -1))

dist = dist + move

return abs(dist)

OR

return abs(sum(np.random.choice(make\_array(1,-1), 100)))

2. Assuming that Achilles’ coin is fair, we would like to simulate what would happen if Achilles took 10000 different random walks. Complete the simulation below and keep track of how far Achilles ends up from the origin in each of his walks in an array called distances. The histogram shown below is an example of a histogram plotted from distances.

distances = make\_array()

for i in np.arange(10000):

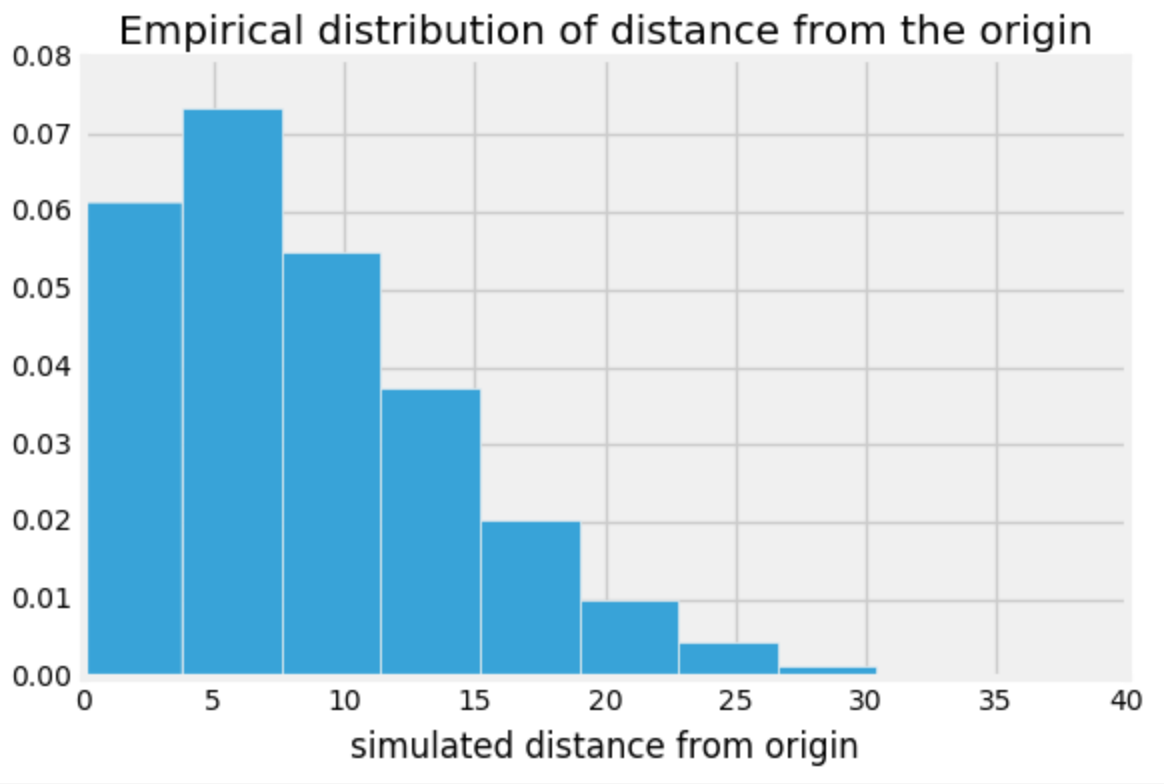
new\_distance = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

distances = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

for i in np.arange(10000):

new\_distance = one\_walk()

distances = np.append(distances, new\_distance)



3. Achilles goes for a walk and claims that at the end of his walk, he ended up 30 steps away from the origin. You notice this is strange, so you want to run a hypothesis test to test whether or not Achilles used a fair coin. Fill in the blanks below for the null and alternative hypotheses and test statistic.

*Hint: When considering your alternative hypothesis, note that we do not really care about whether the coin is biased towards heads or towards tails.*

**Null Hypothesis:**

The coin Achilles uses his walk is fair. The fact that he ended up so far away from the origin is merely due to chance.

**Alternative Hypothesis:**

Achilles coin was unfair.

**Test Statistic:**

Absolute difference from the origin at the end of a walk. Can be thought of as how far he walked in either direction.

4. Write the code to calculate the p-value given the test statistic listed above and using a 5% p-value cut-off. Then, describe the different conclusions that you would arrive at depending on the p-value.

*Hint: We simulated an array in part(b) of test statistics under the null hypothesis. Try to use the* distances *array.*

p\_value = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

p\_value = np.count\_nonzero(distances >= 30) / len(distances)

If the p value is less than the 5% cutoff, we would consider this to be evidence against the null hypothesis while if it is above 5% we would say that there is not strong evidence against the null.

**True/False**

Respond with true or false to the following questions. If your answer is false, explain why.

1. In the U.S. in 2000, there were 2.4 million deaths from all causes, compared to 1.9 million in 1970, which represents a 25% increase. The data shows that the public’s health got worse over the period 1970-2000.

False, because the population also got bigger between 1970 and 2000. It would be more appropriate to look at the total number of deaths compared to the total population at each year. In fact, the U.S. population in 1970 was 203 million, while in 2000 it was 281 million.

2. A company is interested in knowing whether women are paid less than men in their organization. They share *all* their salary data with you. An A/B test is the best way to examine the hypothesis that all employees in the company are paid equally.

False, there is no room for statistical inference here. We have access to the whole population so the answer can simply be retrieved by directly looking at the data. No need for an A/B test here.

3. Consider a randomized control trial where participants are randomly split into treatment and control groups. There will be no systematic differences between the treatment and control groups if the process is followed correctly.

False, randomization can give rise to significantly different treatment and control groups merely by chance, meaning there is still the possibility for systematic differences between the treatment and control groups.

4. A researcher considers the following scheme for splitting a people into control and treatment groups. People are arranged in a line and for each person, a fair, six-sided die is rolled. If the die comes up to be a 1 or a 2, the person is allocated to the treatment group. If the die comes up to be a 3, 4, 5 or 6 then the person is allocated to the control group. This is a randomized control experiment.

True, because the participants were randomly assigned to each group through the roll of a die. This makes it a randomized controlled experiment!

5. You are conducting a hypothesis test to check whether a coin is fair. After you calculate your observed test statistic, you see that its p-value is below the 5% cutoff. At this point, you can claim with certainty that the null hypothesis can not be true.

False, remember the definition of a p-value: A p-value expresses the probability, under the null Hypothesis, that you observe a value for your test statistic that is at least as extreme as your observed test statistic in the direction of the alternative. Assuming that this probability is non 0 then we can not claim that the null can never be true. It could be the case that we simply got an unusual sample from our null.

6. You roll a fair die a large number of times. While you are doing that, you observe the frequencies with which each face appears and you make the following statement: As I increase the number of times I roll the die, the probability histogram of the observed frequencies converges to the empirical histogram.

False, the statement should be: As I increase the number of times I roll the die, the **empirical histogram** of the observed frequencies converges to the **probability histogram** of a fair die.

**Bonus Question:** What is the Law of Averages? Can you see why it allows us to run large scale simulations instead of trying to find exactly what the probability distribution of a test statistic is?

The Law of Averages states that if I repeat a chance experiment many many times under identical conditions, then, in the long run, the proportion of times an event occurs gets closer to the theoretical probability of that event. Hence, the Law of Averages allows us to say that, given enough trials, we can approximate the probability distribution of a test statistic with the empirical distribution. The reason this is extremely useful because a lot of times, identifying the exact probability distribution of a test statistic is quite hard.

**Multiple Choice**

1. Gary is playing with a coin and he wants to test whether his coin is fair. His experiment is to toss the coin 100 times. He chooses the following Null Hypothesis:

**Null Hypothesis:** The coin is fair and any deviation observed is due to chance.

For each of the alternative hypotheses listed below, determine whether or not the test statistic is valid.

a. **Alternative Hypothesis:** The coin is biased towards heads.

**Test Statistic:** # of heads

Correct

b. **Alternative Hypothesis:** The coin is not fair.

**Test Statistic:** # of heads

Incorrect

c. **Alternative Hypothesis:** The coin is not fair.

**Test Statistic:** |# of heads - expected # of heads|

Correct

d. **Alternative Hypothesis:** The coin is biased towards heads.

**Test Statistic:** |# of heads - expected # of heads|

Incorrect

e. **Alternative Hypothesis:** The coin is not fair.

**Test Statistic:** ½ - proportion of heads

Incorrect

2. It is now generally accepted that cigarette smoking causes heart disease, lung cancer, and many other diseases. However, in the 1950s, this idea was controversial. The statistician and geneticist R. A. Fisher advanced the “constitutional hypothesis”, which claims there is some genetic factor that predisposes individuals to smoke as well as to die from diseases.

Suppose that Fisher was correct and there is a gene that predisposes individuals towards smoking as well as getting lung cancer. In the context of this experiment, how would you characterize this gene?

A. treatment

B. outcome

C. confounding factor

D. placebo

**Fun with Functions**

1. Write a function called compute\_pvalue that given an empirical distribution in the form of an array and the observed value of your test statistic, calculates the p-value for that test statistic. You may assume that large values of your test statistic provide evidence against the null hypothesis.

def compute\_pvalue(empirical\_dist, observed\_ts):

return np.count\_nonzero(empirical\_dist >= observed\_ts)/len(empirical\_dist)

2. Now write a function called is\_significant that takes in an empirical distribution, the observed test statistic and a p-value cutoff, returns True if the p-value of the observed test statistic is statistically significant based on the cutoff provided and False otherwise.

*Hint: Use the function you defined in Question 1!*

def is\_significant(empirical\_dist, observed\_ts, cutoff):

observed\_pval = compute\_pvalue(empirical\_dist, observed\_ts)

return observed\_pval <= cutoff

3. Write a function called is\_prime that takes in a number n and returns True if the number is prime and False otherwise. Remember that a number is prime if it is only divisible by itself and 1. In general, we do not consider 1 to be a prime number.

*Hint: The % operator is your friend.*

def is\_prime(n):

if n==1:

return False

for i in np.arange(2, n):

if n%i==0:

return False

return True

**More Hypothesis Testing**

Chloe is a big fan of Trader Joes’ frozen mac n cheese, but she noticed that the cheese used in it varies from box to box. A Trader Joe’s employee provides her with some data about the 4 different cheeses used and the probability of them being used in each box:

|  |  |
| --- | --- |
| **Cheese** | **Probability** |
| Velveeta | 0.05 |
| Gruyère | 0.55 |
| Sharp Cheddar | 0.25 |
| Monterey Jack | 0.15 |

Chloe is suspicious about this distribution. After all, Velveeta is much cheaper to use than Gruyère, and she has also never bought a box that uses Gruyère. Chloe decides to buy many boxes throughout the next month and tracks the type of cheese used in each box. She uses this to conduct a hypothesis test.

1. Write the correct null hypothesis for this experiment

* Null Hypothesis: The types of cheese in the frozen Mac n Cheese boxes are distributed according to the probability distribution provided by the employee.
* Alternative Hypothesis: The types of cheese in the frozen mac n cheese boxes are not distributed according to the probability distribution provided by the employee.

observed\_proportions = make\_array(0.2, 0.3, 0.45, 0.05)

employee\_proportions = make\_array(0.05, 0.55, 0.25, 0.15)

The array observed\_proportions contains the proportions of cheese that Chloe observed in 20 boxes of Mac n Cheese.

2. Chloe wants to use the mean as a test statistic, but Katherine suggests that she uses the TVD (total variation distance) instead. Which test statistic should Chloe use in this case? Briefly justify your answer. Then write a line of code to assign the observed value of the test statistic to observed\_stat.

Katherine is correct, we should use the total variation distance, because she’s comparing two categorical distributions (the observed distribution and the one provided by Trader Joes)

observed\_stat = sum(abs(observed\_proportions - employee\_proportions)) / 2

3. Define the function one\_simulated\_test\_stat to simulate a random sample according to the null hypothesis and return the test statistic for that sample.

def one\_simulated\_test\_stat():

sample\_prop = sample\_proportions(20, employee\_proportions)

return sum(abs(employee\_proportions - sample\_prop)) / 2

4. Chloe simulates the test statistic 10,000 times and stores the results in an array called simulated\_stats. The observed value of the test statistic is stored in observed\_stat. Complete the code below so that it evaluates to the p-value of the test:

np.count\_nonzero(simulated\_stats >= observed\_statistic) / 10000

5. Given that the computed p-value is 0.0825, which of the following are true? Select all that may apply.

1. Using an 8% p-value cutoff, the null hypothesis should be rejected
2. Using a 10% p-value cutoff, the null hypothesis should be rejected.
3. There is an 8.25% chance that the null hypothesis is true
4. There is an 8.25% chance that the alternative hypothesis is true

**A/B Testing**

Alvin loves the lemon bars served at Crossroads and Cafe 3 dinner. However, they are highly popular among students and often run out. Alvin would like to know if a different number of students go to Cafe 3 or Crossroads (for reasons other than chance), so he can attend the less-populated dining hall and grab a lemon bar without worry.

1. Alvin will use A/B testing to figure out if there is a difference in the dining halls. What null hypothesis, alternative hypothesis and test statistic should Alvin use? Make sure to define how you will simulate the null hypothesis.

Null Hypothesis: There is no difference between the distribution of students in Cafe 3 and Crossroads. The distributions of students are drawn from the same underlying distribution, any difference is drawn from chance. To simulate the distributions being the same, we will use a permutation test.

Alternative Hypothesis: The distribution of students between Cafe 3 and Crossroads are not the same.

Test Statistic: The absolute difference in the mean number of students for Cafe 3 and Crossroads.

2. Alvin collects some data about the distribution of students at Cafe 3 and Crossroads for the semester. He reports the following numbers in the dining\_hall table (only the first three rows are shown). Write a line of code to find the observed value of the test statistic and assign it to observered\_stat.

|  |  |
| --- | --- |
| **Dining Hall** | **Students** |
| Crossroads | 350 |
| Crossroads | 420 |
| Cafe 3 | 280 |

means = dining\_hall.group(“Dining Hall”, np.mean).column(“Students mean”)

observered\_stat = abs(means.item(0) - means.item(1))

3. Complete the function one\_sim\_stat to perform one permutation test and return a value of the test statistic.

def one\_sim\_stat():

shuffled = dining\_hall.sample(with\_replacement=False)

original\_with\_shuffled\_labels = dining\_hall.with\_column(“Dining

Hall”, shuffled.column(“Dining Hall”))

means = original\_with\_shuffled\_labels.group(“Dining Hall”,

np.mean).column(“Students mean”)

return abs(means.item(1) - means.item(0))

4. Generate 10,000 simulated statistics and calculate the empirical p\_value.

simulated\_stats = make\_array()

repetitions = 10,000

for i in np.arange(repetitions):

one\_stat = one\_sim\_stat()

simulated\_stats = np.append(simulate\_stats, one\_stat)

p\_value = np.count\_nonzero(simulated\_stats >= observed\_stat) / repetitions

5. Describe some potential flaws in the data collection process that Alvin employed. What could bias his results?

Alvin could eat meals at different times of the day which would impact the two distributions (Cafe 3 at 5 PM vs Crossroads at 8 PM). The data could represent distributions for different days of the week (ie Cafe 3 on Mon, Tues vs Crossroads on Sun, Sat). Sometimes Alvin could care about getting complete data and counting each person he sees, other times he would rather just eat his dinner and hang out with his friends [convenience sample vs random sample].