Course Project: Tantalus

MATH/EE 310/354 - Introduction to Probability and Statistics

Spring 2021

Dr. Abdul Samad, Dr. Shahid Shaikh

Learning domain level: COG 3-4, **CLOs tested:** 1-5

Instructions:

- The project may be be completed in **groups of up to 3. All members MUST belong to the same section**.
- This project contributes **15**% towards the overall grade for the course.
- Each question carries **equal weightage**
- Marks will be awarded for **correctness** and **explanation**.
- The deadline for submission is the end of the 14^{th} **Week, Sunday,** 25^{th} **April**
- **Viva** will ensue for each group in the week following submission
- Marks in each question are subject to the viva.
- Don't plagiarize code. You may discuss approaches with your peers or use online resources to your benefit but **make sure to acknowledge** them.
- The language to attempt the project is restricted to **Python**

The Project:

- A report in PDF form, typed in LaTeX, needs to be submitted for this project
- You may create helper functions or reuse code as you see fit
- State any assumptions you make or any unstated simplifications you perform and state why you made the assumption and why it is reasonable. Don't make unreasonable assumptions.
- Make sure to discuss approach taken in each part of each question. In some cases you may do this simply by starting off with "As the question stated ... we have done so and so using the following values / for the following number of iterations or experiments"
- Include plots wherever required and show working if needed

- Make clear and well thought-out plots. Think from the reader's perspective whether they will be able to understand the diagram or observe the stated trend or the trend you point to you in your explanation
- Feel free to reach out in case of confusion, typos, to discuss approach or a cup of tea[1]

Submission:

3 Items: 2 Zips and 1 PDF

- All code files in one zip folder.
- TeX file along with all its dependencies (images etc.) compressed into a different zip, that can be directly uploaded and compiled.
- Compiled PDF.
- Name the zip files as **student1ID-student2ID-student3ID-code.zip** and **student1ID-student2ID-student3ID-tex.zip** and the PDF in a similar manner.
- Only one person from a group needs to submit

Late Penalties:

2 to 5 hours	10%
5 to 10 hours	25%
10 to 24 hours	50%
>24 hours	100%

[1]Before Ramadan. If we are fasting, we will not break our fast, just because you want to discuss something ~~but food compensation later on in the day is preferred~~

Purpose

The purpose of this project is to familiarize students with the realization of probabilistic methods with simulations and coding.
The **first** question introduces the idea of calculating expected value through multiple simulations. Furthermore, students should come to realize that the expected value found through this way may lie close or far from the theoretical value. Thus, it is useful to calculate the expected value several times and plot it to experience the often normal distribution that expected values follow.
The aim of the **second** question is to teach students to pick values from an arbitrary distribution. Since, the only true random number generator is computers is uniform, students must learn to simulate different distributions by mapping from the uniform distribution. They need to be aware of the both the computational and mathematical side of it.
The **third** question is aimed at addressing natural biases when modelling a scenario. Students must come understand what assumptions they can or cannot make, and what assumptions are at work without their knowledge.
The **fourth** question takes this concept a step further by showing that simplify specify the scenario to modelled without specifying the method is also not enough, as different methods to model a single scenario can result in different distributions.
Question **five** aims to help students understand and apply what they have learned to understand overarching concepts, in this case hypothesis testing, a concept independent of the underlying choice of distribution. It also introduces them to more practical probability and teeters on the edge of statistics.

1 Random Walk 20 points

Random walk– a random walk is a mathematical object, known as a stochastic or random process, that describes a path that consists of a succession of random steps on some mathematical space such as the integers. Mathematical modelling of the movement of animals, micro-organisms and cells is of great relevance in the fields of biology, ecology and medicine.

1.1 8 points

A very simple random walk can be simulated by imagining an object starting at $x = 0$ on a number line. The object is going to take an n number of steps of one unit each, where each step can be right or left with probabilities p and $1 - p$ respectively.

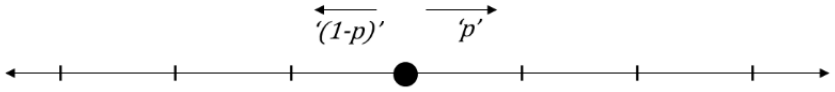

Figure 1: One Dimensional Discrete Random Walk

Intuitively one can judge that if $p = 0.5$, then the object will return to its starting position often, or at least close to it. We are going to confirm this using simulation.

Implement a function that takes in two parameters n and p. It should take, n, number of steps left or right, as dictated by the probabilities and note the final position of the object. This should be repeated several items, which we will call iterations, and the expected value of the objects final position should be calculated using these iterations. The expected value will not come out to be zero every time even if appropriate probabilities are passed. Thus, the procedure/experiment must be repeated several times, to obtain several expected values.
Plot the histogram of these expected values to observe if they follow some trend. Tune the bin width, bin count, number of iterations, number of experiments as you see fit to make the trend obvious. Mention them in your report. Repeat this with a different value of p.

You may work out the expected final position for arbitrary n and p for a bonus. Some good approaches for this are the Bernoulli random variable and the Binomial Random variable. Include full working in your report if you do.

1.2 4 points

Now we are going to simulate the same random walk with an added constraint. Namely, that going into the negative part of the number line is not allowed. Therefore, at any time the object finds itself at $x = 0$, the only option it has is to move in the positive direction. Implement a function that takes in parameter, n and p, and does exactly what the above the function does. Plot a histogram of expected values and discuss your results.

1.3 8 points

Removing the constraint of the previous question, we will now note the number of steps it takes for two objects initialized at different points, and moving randomly to meet. Implement a function that

takes in 4 parameters, the start position of object 1, the start position of object 2, the probability of object 1 to move right and the probability of object 2 to move right. It conducts a large number of experiments each with several iterations, to collect a number of expected values for the number of steps it takes for the two objects to meet. It should plot the expected values in a histogram. You may assume that the initial distance between the objects is an even number.

2 Simulating Distributions — 20 points

Look at the following algebra and examine the accompanying code.

Let X follow a uniform distribution between 0 and 1. The probability that X is less than some number, x, is $P(X < x) = x$. Suppose we want Y to follow a random distribution for which we do not have any in built functions. Let Y follow the distribution $f_Y(y) = e^{-y}$ for $y \geq 0$. The following trick is used to derive the relation between X and Y.

$$P(Y < y) = P(X < x)$$
$$\int_0^y e^{-y}\, dy = x$$
$$1 - e^{-y} = x$$
$$y = -\ln(1 - x)$$

```
y = []

for i in range(100000):
    x = np.random.random()
    y.append(- np.log(1-x))

bins = 20
binWidth = (max(y) - min(y)) / bins
plt.hist(y, bins=bins, weights=np.ones(len(y))/(len(y)*binWidth))
values = np.linspace(min(y), max(y), 50)
plt.plot(values, np.exp(-values))
plt.show()
```

2.1 — 4 points

Does the code accomplish simulating the distribution? Which distribution does it follow? Try running the code with different number of bins. Attach plots and discuss your results.

2.2 — 8 points

Examine the following section of code and mathematically deduce what distribution Y follows (Try working the above trick in reverse starting with the last statement). Show all required working.

```
y = []
for i in range(100000):
    x = np.random.random()
    y.append(1 / (1-x))
y.sort()
ind = (np.array(y) > 30).tolist().index(1)
y = y[:ind]
bins = 100
binWidth = (max(y) - min(y)) / bins
plt.hist(y, bins=bins, weights=np.ones(len(y))/(len(y)*binWidth))
values = np.linspace(min(y), max(y), 50)
plt.plot(values, (1 / values ** 2))
plt.show()
```

Why are the lines 5-7 important. What does removing them do?

2.3 — 8 points

Implement a function that returns a random variable from the distribution,

$$f_Y(y) = \frac{1}{y^3} \text{ for } y \geq \sqrt{\frac{1}{2}}$$

Use it to produce a histogram and line plot like the above code.
Implement a different function that calculates the expected value using the experiments and iterations approach and plots the set of expected values obtained. You may need to utilize the trick pointed to in the above lines and choose an appropriate cutoff for both of these.

3 Picking a random point correctly — 20 points

3.1 — 5 points

For this question, you have to pick random points in a circle in a uniform manner. The most intuitive approach for this is usually to pick a random number, r, from the uniform distribution between 0 and R, where R is the radius of the circle. Similarly, one can pick the angle θ in a similar manner and generate x, y coordinates from them.

Implement a function that takes in a radius, R, and samples a large number of points in the described manner. The function should generate a scatter plot containing all the sampled points, as well as plotting a circle of the appropriate radius, Find and mention the variation in the x-coordinates as well.

3.2 — 7 points

This, however, does not result in a uniform pick.[2] You may spot this from the plot which should have points concentrated more towards the center rather then points being uniformly spread out across the circle. Change the number points you are plotting if you do not observe this trend. Now instead of generating r and θ values we will generate x and y values uniformly. To generate random points on a circle of radius, R, pick both x and y independently and uniformly from the range $[-R, R]$ to obtain

[2]This was the method given in Assignment 1, Sharp Shooter. If you had done that question, you would have observed that simulation was giving probability 0.4, while your analytical result was 0.368. You can try to implement actual uniform selection and will see that the new simulation's result is close to the theoretical value.

a point. If the distance of this point from the origin is more than R, discard it and generate a new point in its place.

Implement a function that takes in a radius, R, and samples a large number of points in the described manner. The function should generate a scatter plot containing all the sampled points, as well as plotting a circle of the appropriate radius, Find and mention the variation in the x-coordinates as well. Comment on why this found variation is different or same as in the previous part.

3.3 8 points

To get an intuition of why the first approach does not result in a uniform pick imagine a circle of radius 1 embedded in a circle of radius 2 as shown in Fig.2. If points are picked randomly, the probability of the point lying inside the larger circle should be 4 times than the smaller one. Does this hold when the above described method to pick r and θ is used? Explain in report with working.

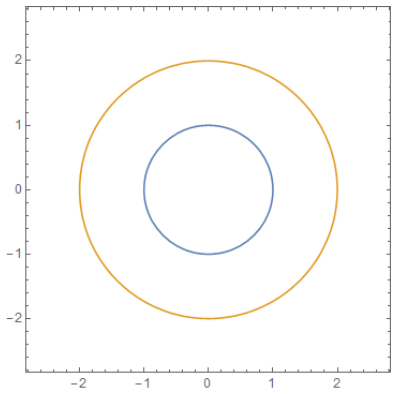

Figure 2: Comparison of area. Circles of radius 1 and 2

In this part, modify one or both of the ways to pick r and θ, such that the points are sampled in a uniform manner and a plot similar to that in part 2 is obtained. Implement a similar function as the above part. The plot generated this time should contain points that are uniformly spread across the circle. Describe how are you picking the random variables and find the variance of the x-coordinates once again and comment on your results.

If you feeling up to it or for a bonus then you may derive the distribution from the following two facts.

The probability of a point to lie inside a circle of radius, $r \leq R$ is proportional to its area. i.e. $P(r \leq R) = k \cdot \pi r^2$. The probability of it lying inside the outermost circle of radius, R, should be 1 i.e. $P(R \leq R) = 1$.

After finding the distribution that r follows, you may then generate the values of r appropriately by mapping from the uniform random distribution as in the previous questions. Show all mathematical working.

4 Saying random is not enough - Approaches effect distributions 20 points

In this question we are going to observe the distribution followed by the length of a random chord picked from a circle of radius r.

The difficulty of the question lies in how to pick a random chord in a circle. For each of the described approaches implement a different function that takes in radius, r and plots a histogram of the length of chords with an appropriate number of bins, with proportion (probability) of values in the bin on y-axis instead of counts. **Include mathematical calculation of chord lengths in all parts.**

4.1 6 points

For the first approach we imagine the circle centred on the origin of the Cartesian plane. The $\theta = 0$ ray/line is defined as starting at the origin and pointing in the direction of increasing x, and theta increasing counter clockwise. We pick two angles θ_1 and θ_2 uniformly between 0 and 2π, and our random chord is the chord between the points of the circle defined by those two angles.

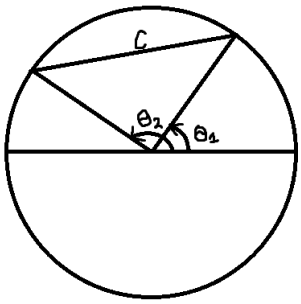

Figure 3: Picking a chord through 2 random angles

4.2 6 points

For the second approach we imagine the circle in a similar manner. Then we pick a random direction, θ, and draw a line from the center of the circle to its boundary such that the angle from the ray $\theta = 0$ to this line, measured counter clockwise is θ. To create a random chord, we pick a point along this line and construct the perpendicular bisector of the line at this point. The perpendicular bisector can be extended to touch the boundary of the circle at either ends to obtain a chord.

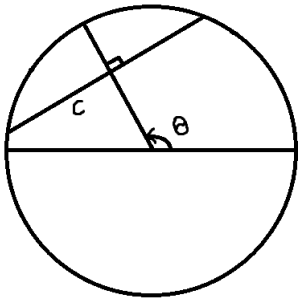

Figure 4: Picking a chord as bisector of some ray

4.3 **6 points**

For the third approach we again visualize the circle as before. This time we pick a random point uniformly from the circle as we did in the previous question. You may use any helper functions you may have developed in the previous part for this. After picking a point we find the chord which will have this point as it midpoint and this will be our random chord. There will be only one such chord.

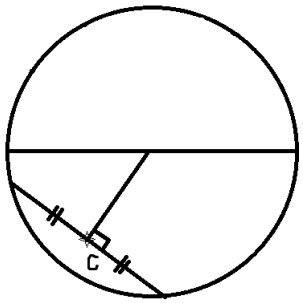

Figure 5: Picking a chord through random point

4.4 **2 points**

You will notice that all of these approaches results in a different distribution. Which of these do you think corresponds most to our goal, which was to find the distribution of the length of a random chord.

5 Hypothesis Testing 20 points

Intuition - If someone hands you a coin, and tells you its fair, you toss it 15 times and get 15 heads, you are going to be skeptical. That is the essence of hypothesis testing, we make a certain assumption, and then sample some data. If the sum of probability of obtaining the observed data or data less or equally likely is less than a certain threshold then we conclude our assumption to be false. The statement 'or data less likely' is a little vague and more importantly problem dependent. Let use look at a concrete example.

Suppose you have a coin which we do not know as fair or not. We assume that the coin is fair. This is known as the null hypothesis. The alternative hypothesis is that the coin is not fair. We then set a certain threshold, and declare that given our assumption if the observed data or data less or equally likely has a total probability less than this threshold we will reject our assumption. Let us set the threshold at 0.05.

We toss the coin 15 times and obtain 15 heads. The probability of this happening given that the coin is fair is $(\frac{1}{2})^{15} \approx 0.00003$. An event that is less or equally like is getting 15 tails, with a probability of 0.00003 as well. The cumulative of these is 0.00006 which is less than our threshold, therefore we reject the null hypothesis.

Suppose instead that we had tossed the coin 10 times and obtained 2 heads, while using the threshold of 0.1. The events equally or less likely are getting 2 or less heads and 2 or less tail, the sum of whose probabilities is 0.109375, which is greater than our threshold. Therefore, we declare that the null hypothesis is valid, and an unlikely but not too unlikely possibility has occurred.

It may be argued that in the former case as well, the coin could have been fair and it was only that an unlikely possibility had occurred. The argument is valid, and when it comes to simulations, one can rectify this problem by repeating several times to obtain an expected value, and then repeating the entire experiment multiple times, to get a distribution of the expected values as we did in the previous questions. In real life, however, we hardly have such liberties, such as when conducting surveys, and therefore hypothesis testing remains a reliable method. Of course, we could be wrong sometimes to reject the null hypothesis but we would be right most of the time. *That's just how probability works.*

5.1 5 points

Implement a function that simulates the behavior of a fair coin, you may choose return types as you see fit. Implement another function that uses the above function to simulate 10 coin tosses multiple times and finds the expected number of times the null hypothesis is rejected even though it is true. Use the several experiments each having several iterations approach to generate a histogram of expected values. Mathematically and simulation-wise, what is the probability we will reject the null hypothesis even though it is true. Explain both approaches in your report. Use a threshold of 0.05. Reach out if you have confusions but not at the 11^{th} hour.

5.2

You are out fishing. The length of fish in your fishing area follows a normal distribution. You are trying to prove or disprove what someone said to you to about the mean length of the fishes. Unfortunately, you do not have access to the lengths of every fish in the area, which would allow you to calculate the population mean and the population variance. The best you can do is to catch a small sample, find the sample mean and the sample variance, and make some simplifying assumptions. You are provided some code files which can be used in the following way to catch a single fish and measure its length.

```
import fishCImport as f
length = f.fish()
```

5.2.1 9 points

Suppose that the mean length you have been told is 23, and the size of the sample i.e. the number of fish you decide to catch, n, is 30. You simplified your problem by stating that the means of samples follow the normal distribution with mean u_0 which is the population mean, and standard deviation, $\frac{\sigma}{\sqrt{n}}$, where σ is the standard deviation of your sample, and n the size of your sample.

$$S \sim N\left(u_0, \left(\frac{\sigma}{\sqrt{n}}\right)^2\right)$$

You may start off by declaring that the null hypothesis is that the population mean, u_0, is exactly 23. Conduct hypothesis testing several times with a threshold of 0.05. Measure the proportion/expected number of times, the null hypothesis is rejected. Conduct the experiment several times to find several values of this proportion. Plot these as a histogram.

Implement a function that takes in u_0 and n, and conducts a single hypothesis test and returns its result. A single hypothesis test here constitutes catching a sample of 30 fish, finding the sample mean, u, the sample variance, σ, and using the above specified normal distribution with the population mean, u_0, as 23, assumed through the null hypothesis, to find the probability of obtaining the sample mean or a mean with an absolute difference greater or equal to $|u - u_0|$. Mathematically, if $a = |u - u_0|$, then the null hypothesis is rejected if

$$P(|S - u_0| \geq a) < \text{ Threshold}$$

Implement another function that utilizes the above function or otherwise, performs several experiments, each with several hypothesis tests and plots a histogram of the proportion of times the null hypothesis is rejected.

Comment on whether it would have been sufficient to accept or reject the null hypothesis based on a single hypothesis test.

5.2.2 **3 points**

Conduct the same experiments with same u_0 and $n = 70$. Implement a different function for this and generate a similar histogram plot of proportion of times the null hypothesis is rejected.

Comment on what increasing the value of n accomplishes and whether it would have been sufficient to accept or reject the null hypothesis based on a single hypothesis test in this case.

5.2.3 **3 points**

In 5.1 we saw that proportion of times the null hypothesis is rejected despite being true is close to the threshold we choose. In normal distributions it is exactly equal to the threshold. Experimentally or mathematically, determine the least value of n (or close enough) to ensure that the null hypothesis is not wrongly rejected more than 10 percent of the time. You may use a sample standard deviation of 3, if you decide to approach mathematically. If you decide to approach simulation wise you will have to define your own fish function which returns a random variable from the normal distribution $N(23, 3^2)$. Attach a similar histogram plot for your chosen value of n in this approach.