Practical Homework 3

ECE399 - Probability Models and Inference for Engineering

May 10, 2019

In this homework, you'll have a glance into continuous random variables and will have a brief introductory understanding of some of the popular continuous distributions. The generator functions suggested here are all from the package random. However you may use any other package available.

1 Part 1: A Glimpse of the Continuous Random Variables

1.1 Plotting the Distribution of Discretized Continuous Random Variables

In this questio, you will have a glance into some of the famous continuous random variables and their -approximated- distributions. We are going to approximate the continuous random variables with discrete ones, and then plot the distributions.

A) Uniform distribution: The simplest continous distribution is *uniform* or *rectangular*, where in a closed interval $[a, b]$ all the real numbers are equally probable, and the numbers outside $[a, b]$ have a zero probability to appear. In python, from the package random the function uniform generates continuous uniform numbers. The function takes two input arguments: The first one is the lower limit of the interval (denoted by a), and the second one is the upper limit b. Then, every time called, the function will return a single random number uniformly distributed between a and b.
a) Generate 100,000 random uniformly distributed samples between $a = 0$ and $b = 1$ using function uniform.
b) Quantize the interval [0,1] into 100 equally-spaced intervals, and map the samples to each corresponding interval. One way of doing this is to replace each sample inside each interval with the center of the interval. For example, if you have observed 0.6382, then it lies within your 64th interval $[0.63, .64]$ and you approximate it by 0.635. This will approximate the continuous random variable with a discrete one.
c) Plot the pmf of the discrete random variable defined above. How does it look like?

B) Gaussian distribution: Gaussian or Normal distribution appears in many natural phenomena and is often used in the natural and social sciences to represent real-valued random variables whose distributions are not known. In Python, the function gauss from random generates samples from the Gaussian distribution. The function takes two input arguments: The first one is the average of the distribution denoted by μ, and the second one is the *standard deviation* of the Gaussian distribution denoted by σ (Don't worry if you don't know what that means, you'll learn later throughout the course).
a) Generate 100,000 samples setting $\mu = 0$ and $\sigma = 1$.
b) Similar to the previous part, divide the interval $[-5, 5]$ into proper number of equal intervals (for example 100), and quantize the samples like the previous part. (Discard all the samples outside of $[-5, 5]$.)
c) Plot the pmf of the discretized random variable. Compare the pmf to $\alpha \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(x-\mu)^2}{2\sigma^2})$. ($\alpha$ is a scaling parameter properly set by you.)

C) Exponential distribution: The exponential distribution is one of the well-known distributions used to describe the time interval between events in some natural phenomena modelings (For example, the time between two consecutive lightnings on a stromy day). In Python, the function expovariate from random generates samples from the exponential distribution. The function takes the *rate* parameter denoted by λ as the input.

a) Similar to previous parts, generate 100,000 samples setting $\lambda = 1$.
b) Again, quantize the interval $[0, 5]$ into the proper number of intervals such as 100. Discard any sample outside $[0, 5]$.
c) Plot the pmf of the discretized random variable. Compare it to $\alpha\lambda \exp(-\lambda x)$. ($\alpha$ is again a scaling parameter properly set by you.)

1.2 Sum of the Independent Random Variables

Consider a set of m random variables $X_1, X_2, \ldots, X_m$. Let's define the random variable Y as the summation of all X_i variables:

$$Y = \sum_{i=1}^{m} X_i \tag{1}$$

A) For the case $m = 10$ and X_is being independent uniform variables in the interval $[-0.5, 0.5]$, generate 100,000 samples of Y. Use the discretization technique from the previous section for the $[-5, 5]$ interval and plot the pmf of Y.

B) Now increase m to 100 with X_is being uniform variables in the interval $[-0.15, 0.15]$, repeat the previous part.

C) Increase m to 1000 with X_is being uniform variables in the interval $[-0.05, 0.05]$, repeat the previous part. What does the distribution of Y look like as you increase m? Compare to the distributions introduced in question 1.1.

D) Repeat the experiments A, B, and C, this time with X_is being independent exponential variables, with rate equal to λ. For $m = 10$ set $\lambda = 3$, for $m = 100$ set $\lambda = 10$, and for $m = 1000$ set $\lambda = 30$. Choose the discretization interval properly so almost all of the Y samples are accounted for (It won't be necessarily equal to $[-5, 5]$ you saw before). Compare to the distributions from question 1.1.

2 Part 2: Experimenting with Continuous Random Variables

2.1 Height-Weight-Temperature Dataset

In this question, we will look at an artificial height-weight-temperature dataset, in which we have measured the height and weight of $100,000$ different people, as well as the temperature of the room in which the measurement is carried out. Let's denote the height with the random variable X, weight with Y and the temperature of the room with Z.

A) (Short answer) Do you think X and Y are independent or not? How about X and Z? How about Y and Z?

B) Download and read the dataset file HeightWeightTemperatureData.csv containing the measurements for $100,000$ people as mentioned above. The first column represents the ID of each case, the second column represents height in centimeters, the third column keeps the weight of each person in pounds, and the third column is the temperature of the experiment room in Fahrenheits.

C) With the quantization technique you learned in section 1.1 (also called *binning*), plot the distribution of X, Y, and Z. To which one of the distributions you saw in section 1.1 do they look similar?

D) Estimate the expected value and the variance of X, Y, Z. (You may use numpy.mean and numpy.var functions)

E) Estimate the expected value of XY. Does the equation $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$ hold? Repeat this question for XZ and YZ. Are the results conforming to your answer to part A?

F) Estimate the variance of $X + Y$. Does the equation $\text{Var}[X + Y] = \text{Var}[X] + \text{Var}[Y]$ hold? Repeat this question for $X + Z$ and $Y + Z$. Are the results conforming to your answer to part A?

2.2 Communication Channel

In this question we will look at a simplified model of a communication channel. In general, a communication channel or simply channel refers either to a physical transmission medium such as a wire, or to a logical connection over a multiplexed medium such as a radio channel in telecommunications and computer networking. A channel is used to convey an information signal, for example a digital bit stream, from one or several *senders (or transmitters)* to one or several *receivers.*

Transmitter: Our very simple model consists of a single transmitter that creates random pulses -1 and 1 with equal probability. Let's denote the ranodm variable corresponding to the pulses by X.

A) What distribution does X have? Generate $100,000$ samples of it.

B) Estimate the mean and the variance of X. Compare to the theory values.

Channel: The samples of X generated above go through the channel and are transformed into another random variable denoted by Y. Let us here consider a very simple model for the channel called *Additive white Gaussian Noise* (AWGN) which means each samlpe of X is polluted by a Gaussian noise term, i.e. we have:

$$Y = X + N \tag{2}$$

Where N is a Gaussian random variable independent of X. Let's assume the mean of the noise is 0 and the variance of it is equal to 1.

C) Simulate the AWGN channel mentioned above, and generate samples of Y corresponding to the samples of X from part A.

D) Plot the distribution of Y samples. What does it look like?

Receiver: The polluted samples Y are then received by the receiver. The receiver's goal is to recover the original X samples with the minimum error possible by looking at the Y samples received. The receiver has no knowledge of the noise terms N. Assume the receiver in our model works simply like this: for each sample of Y, if it's greater than zero, then the recovered value of the X will be 1, otherwise it will be equal to -1.

E) Simulate the receiver decribed above, and apply it to your samples of Y from part C. What fraction of the X samples are recovered wrongly (in other words, what is the probability of error)?

F) Now let's increase the power of the transmitter: This time let's assume the transmitter generates -5 and 5 randomly. Repeat parts B, C, D and E. How does the probability of error changes with increasing the transmitter's power?