Homework 02

Imola Fodor SM3500474

Probabilistic Machine Learning 2022

Exercise 1.

0.3% of the population has an unknown virus and a test is being developed.

This test gives a false positive 10% of the time and a false negative 5% of the time.

1. Calculate the probability that you are positive to the test.
2. Suppose you are positive to the test. What is the probability that you contracted the disease?

Result

1. The probability that a person is positive to the test is 10.255%, shadowed out in green. The result is gotten by summing the probability of a True Positive and a False Positive (both normalized w.r.t. to Truth) result on a test.

		Truth		
		Infected	Not Infected	Total
Test Result	Positive	95%*0.3 = 0.285 (*True positive*)	10% * 99.7 = 9.97 (*False Positive*)	10.255 *Test Positive*
	Negative	5%*0.3 = 0.015 (*False Negative*)	90% * 99.7 = 89.73 (*True Negative*)	89.745 *Test Negative*
		0.3 *Total Infected*	99.7 *Total Not Infected*	100

2. Let's consider the random variable of *contracting the disease* as A, with possible values Has/~Has, and random variable B, for *test is positive*, with values Yes/No.

$$P(A|B) = \frac{P(A = Has\ , B = Yes)}{P(B = Yes)} = \frac{P(B = Yes|A = Has)P(A = Has)}{\sum_{i\in\{\text{Has},\sim\text{Has}\}} P(A = Yes|B = b_i)P(B = b_i)} =$$

$$= \frac{95*0.3}{95*3+10*99.7} = \frac{28.5}{1025.5} = 0.027 = 2.7\%$$

The probability of having the disease, given that one is positive to the test is 2.7%.

Exercise 2.

Implement the empirical cumulative distribution function $F\_X(x) = \text{cdf}(\text{dist, x, n\_samples})$ taking as inputs a **pyro.distributions** object **dist** , corresponding to the distribution of X , a real value x and the number of samples n_samples .

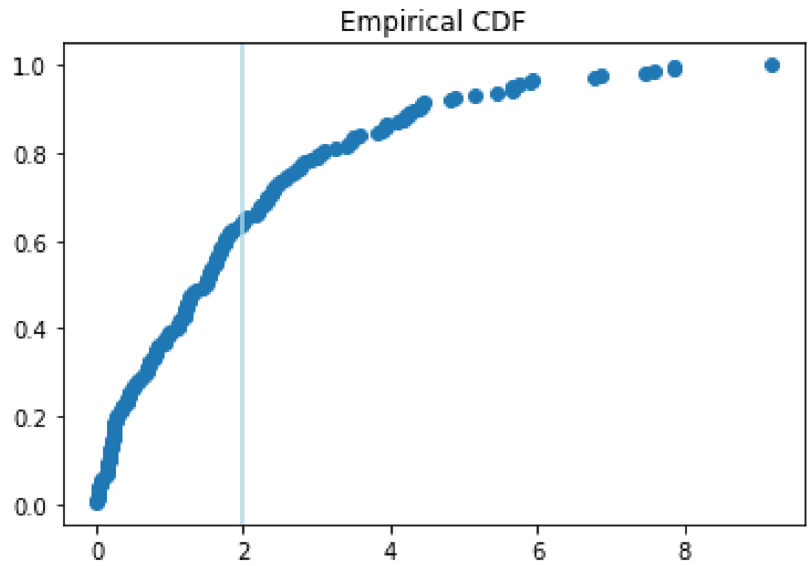
The function $\text{cdf(dist, x, n\_samples)}$ should return the value of at x and also plot the cdf.

Suppose that $X \sim Exp(0.5)$. Using your function, plot and compute $F_X(x = 2)$.

Result

```
import torch
import pyro
pyro.set_rng_seed(1) # for reproducibility

import matplotlib.pyplot as plt
import pyro.distributions as dist

# distribution
exp = dist.Exponential(0.5)


import numpy as np

def ecdf(dist, k, n_samples):
    samples = [pyro.sample("d",dist) for i in range(n_samples)]
    x = np.sort(samples)
    # the y values correspond to the proportion of data points less 
    than each data point
    y = np.arange(1, n_samples+1) / n_samples
    result = np.interp(k, x,y)

    plt.scatter(x=x, y=y);
    plt.axvline(x=2, color='lightblue')

    plt
    print(result)


ecdf(exp, 2, 200)
```

$F_X(x = 2) =$: 0.64



Exercise 3.

Suppose the heights of female students are normally distributed with unknown mean μ and known variance 6^2.

Suppose that μ is in the range [155, 175] with approximately 95% probability and assign to μ a normal $N(160,3^2)$ prior distribution.

1. Using the cdf from the previous exercise, empirically verify that the prior probability that μ ∈ [155, 175] is approximately 95%.
2. Analytically derive the posterior distribution for a set of observations of heights x=[x1, ..., xn] .
3. Plot the posterior distribution corresponding to the data x = [174, 158, 194, 167] together with the prior distribution

Result

1. Herein verifying that the prior probability of the given range is apx. 95%.

```
normal = dist.Normal(160, 3)
ecdf(normal, 155, 200)
ecdf(normal, 175, 200)
0.0680446593502119
1.0
```

$$\mathrm{P}(145 \leq \mu \leq 175) = \mathrm{P}(\mu \leq 175) - \mathrm{P}(\mu \leq 145) \approx 1 - 0.06 \approx 0.95.$$

1. Assuming that $x_i|\mu \approx \mathrm{N}(\mu, \sigma^2)$ i.i.d. and $\mu \approx \mathrm{N}(\mu_0, \sigma_0^2)$, knowing the posterior distribution formula Posterior ∝ Normal Likelihood × Normal Prior, we derive:

<u>*Normal prior*</u> with known prior mean and variance: $p(\mu|\mu_0, \sigma_0^2) = \frac{1}{\sqrt{2\pi\sigma_0}} e^{-\frac{1}{2\sigma_0^2}(\mu-\mu_0)^2}$

<u>*Normal Likelihood*</u> with unknown mean and known posterior variance:

$$p(x_1, x_2, \ldots, x_n|\mu) \propto \frac{1}{\sqrt[n]{2\pi\sigma}} e^{-\frac{1}{2\sigma^2}\sum_1^n (x_i-\mu)^2}$$

Then, for the <u>*posterior distribution*</u>:

$$p(\mu|x_1, x_2, \ldots, x_n) \propto p(x_1, x_2, \ldots, x_n|\mu)p(\mu)$$

$$p(\mu|x) = (2\pi\sigma_{post}^2)^{\frac{1}{2}} e^{-\frac{1}{2*\sigma_{post}}(\mu-\mu_{post})^2}$$

To deal with the posterior for multiple measurements, instead to use the sum, the mean of the sample was used, x':

$$x' = \frac{1}{n}\sum_{i=1}^{n} x_i$$

By further steps, not denoted in this report, for the posterior distribution of the mean, we get:

$$p(\mu|x_1, x_2, \ldots, x_n) \approx \mathrm{N}\left(\frac{\frac{n}{\sigma^2}\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right) + \frac{\mu_0}{\sigma_0^2}}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}, \frac{1}{\frac{1}{\sigma_0^2} + \frac{n}{\sigma^2}}\right) \approx \mathrm{N}\left(\frac{\frac{\sum x}{\sigma^2} + \frac{\mu_0}{\sigma_0^2}}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}, \frac{1}{\frac{1}{\sigma_0^2} + \frac{n}{\sigma^2}}\right)$$

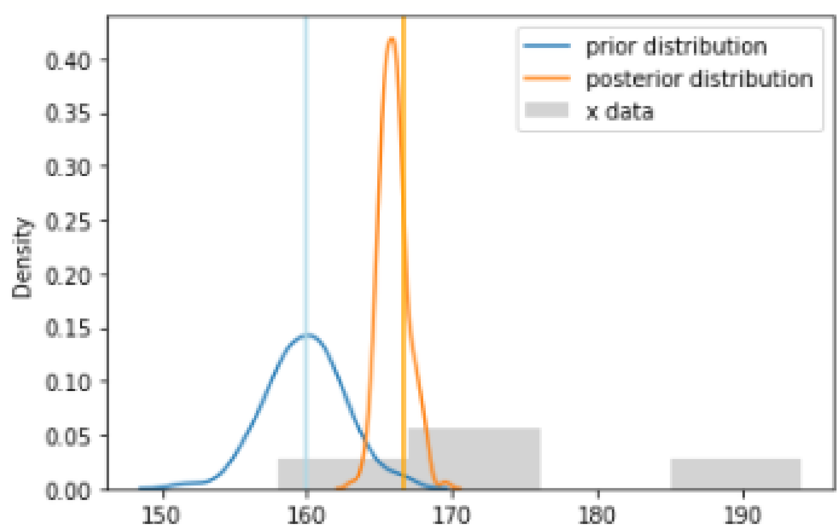
Therefore, using the μ_{post}, shadowed out in the posterior normal distribution, we can check the updated mean's value for additional data; similarly, the σ_{post}:

```
import math

mu_0 = 160
sigma_0 = 3
sigma_post = 6

x = [174, 158, 194, 167]
x_i = 693/4 #sum

mu_post = (160/9 + 693/36)/(1/9 + 4/36)
sigma_post = math.sqrt(1/ (1/9 + 4/36))

print(mu_post)
print(sigma_post)
```

μ_{post} = 166.625
σ_{post} = 2.121

2. Herein the prior and posterior distribution compared with the x data provided. It can be noted that the posterior distribution gives a more precise mean.

```
# distribution
normal_prior = dist.Normal(160, 3)
normal_post = dist.Normal(166, 2.12)

normal_prior_samples = [pyro.sample("normal",normal_prior) for i in range(200)]
normal_post_samples = [pyro.sample("normal",normal_post) for i in range(200)]

#plot
sns.distplot(normal_prior_samples,label='prior distribution', hist = False)
sns.distplot(normal_post_samples, label='posterior distribution', hist = False)
plt.hist(x, bins=4, range=None, density=True, color = "lightgray", label="x dat
a")

plt.legend()
plt.axvline(x=mu_0,
            color='lightblue')
plt.axvline(x=mu_post,
            color='orange')
plt.show()
```



Exercise 4.

Prove that the Beta distribution is a conjugate prior distribution for the Geometric likelihood.

Result

If the posterior distribution $p(\Theta|x)$ belongs to the same family as the prior distribution $p(\Theta)$, then the prior is said to be a conjugate prior for the likelihood function $p(x|\Theta)$.

To prove that the Beta distribution is a conjugate prior distribution for the Geometric likelihood we need to prove that:

$$\text{Posterior} \propto \text{Geometric Likelihood} \times \text{Beta Prior}$$

, where the Geometric Likelihood: $p(x|\Theta) = (1-\ \Theta)^{x-1} * \Theta$

, and Beta Prior $p(x|\alpha, \beta) = constant * x^{\alpha-1}(1-\ x)^{\beta-1}$.

The x of the Beta Prior becomes Θ, because the sample parameter is represented by Θ itself, the constant is not affecting the proportionality, and we can write:

$$\begin{aligned} \text{Posterior} &\propto (1-\ \Theta)^{x-1} * \Theta \times\ constant * \Theta^{\alpha-1}(1-\ \Theta)^{\beta-1} \\ &\propto \Theta^{(\alpha-1)+1}\ * (1-\Theta)^{(\beta-1+x-1)} \\ &\propto\ \Theta^{(\alpha+1)-1} * (1-\Theta)^{(\beta-1+x)-1} \\ &\propto \mathrm{Beta}(\alpha+1, \beta+\mathrm{x}-1) \end{aligned}$$

Indeed, the posterior distribution belongs to the same family as the prior, the Beta distribution, hence we it can be stated that the Beta distribution is a conjugate prior for the Geometric distribution.