

# Homework Stats 1

Week 1

6733172621 Pathadon Phengpinij

*Collaborators.* ChatGPT (for L<sup>A</sup>T<sub>E</sub>X styling and grammar checking)

---

## 1 Sampling

Sampling is a process that is very important for writing simulations. In this section, you will try to sample from some common distributions. You may implement them yourself or use the provided distribution from `scipy.stats`.

```
1  from scipy.stats import norm, bernoulli, binom, uniform, geom,
2      expon
3
4  # Sample from Uniform(a, b)
5  def sample_uniform(sample_size, a, b):
6      # [YOUR CODE HERE]
7      dist = uniform(a, b)
8      return dist.rvs(sample_size)
9
10 def sample_normal(sample_size, mu, sigma):
11     # [YOUR CODE HERE]
12     dist = norm(mu, sigma)
13     return dist.rvs(sample_size)
14
15 def sample_bernoulli(sample_size, p):
16     # [YOUR CODE HERE]
17     dist = bernoulli(p)
18     return dist.rvs(sample_size)
19
20 def sample_binomial(sample_size, n, p):
21     # [YOUR CODE HERE]
22     dist = binom(n, p)
23     return dist.rvs(sample_size)
24
25 def sample_geometric(sample_size, p):
26     # [YOUR CODE HERE]
27     dist = geom(p)
28     return dist.rvs(sample_size)
29
30 def sample_exponential(sample_size, l):
31     # [YOUR CODE HERE]
32     dist = expon(l)
33     return dist.rvs(sample_size)
```

**TO SUBMIT**

Hamtaro and his friends are collecting sunflower seeds. The bigger the sunflower, the more seeds they can find! The probability of finding a sunflower of a certain height  $x$  (in cm, from 0 to 10) increases with its height, following the probability density function  $f(x) = \frac{x}{50}$ . Write a function `sample_increasing(sample_size)` to simulate the heights of the sunflowers the Ham-Hams find.

**Solution.** First, we need to find the cumulative distribution function (CDF) from the given probability density function (PDF)  $f(x) = \frac{x}{50}$ . To find the CDF, we integrate the PDF from 0 to  $x$ :

$$F(x) = \int_0^x f(t) dt = \int_0^x \frac{t}{50} dt = \left[ \frac{t^2}{100} \right]_0^x = \frac{x^2}{100} \text{ for } 0 \leq x \leq 10.$$

Next, we need to find the inverse of the CDF,  $F^{-1}(y)$ , to use the inverse transform sampling method. Setting  $y = F(x)$ , we have:

$$y = \frac{x^2}{100} \implies x^2 = 100y \implies x = 10\sqrt{y}.$$

Thus, the inverse CDF is  $F^{-1}(y) = 10\sqrt{y}$ .

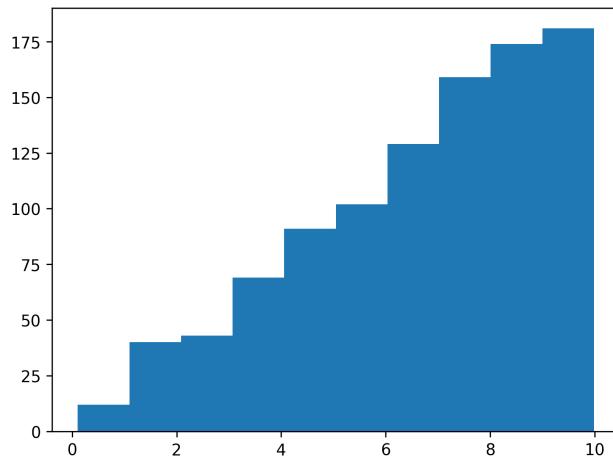
Now, we can implement the function `sample_increasing(sample_size)` to generate random samples from the distribution.

```

1 import numpy as np
2
3 np.random.seed(1) # For reproducibility
4
5 # sample from pdf f(x)=x/50, 0<=x<=10
6 def sample_increasing(sample_size):
7     # [YOUR CODE HERE]
8     u = np.random.rand(sample_size)
9     dist = 10 * np.sqrt(u)
10    return dist

```

We can plot the histogram of our samples. If the sample functions are implemented correctly, the histogram should look like our distribution.



## 2 Maximum Likelihood Estimation

**TO SUBMIT**

**Problem 4.** Dexter tracks the growth of his prized sunflower over three days (day 0, 1, and 2). He believes the sunflower's height at the end of each day  $y_{t+1}$  is its height from the previous day  $y_t$  multiplied by a secret growth factor,  $\alpha$ , plus some random daily noise.

For a two-day period, we can observe the following Markov process:

$$P(y_2, y_1, y_0 | \alpha) = P(y_2 | y_1)P(y_1 | y_0)P(y_0 | \alpha)$$

where  $y_2 \sim \mathcal{N}(\alpha y_1, \sigma^2)$ ,  $y_1 \sim \mathcal{N}(\alpha y_0, \sigma^2)$ ,  $y_0 \sim \mathcal{N}(0, \lambda)$

Find the Maximum Likelihood Estimate (MLE) for the secret growth factor,  $\alpha$ , given the observed heights at the end of each day  $y_2, y_1, y_0$ . In other words, compute for the value of  $\alpha$  that maximizes  $P(y_2, y_1, y_0 | \alpha)$ .

**Solution.** The probability density functions for the normal distributions are given by:

$$\begin{aligned} P(y_2 | y_1) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_2 - \alpha y_1)^2}{2\sigma^2}} \\ P(y_1 | y_0) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_1 - \alpha y_0)^2}{2\sigma^2}} \\ P(y_0 | \alpha) &= \frac{1}{\sqrt{2\pi\lambda}} e^{-\frac{y_0^2}{2\lambda}} \end{aligned}$$

Thus, the joint probability is:

$$\begin{aligned} P(y_2, y_1, y_0 | \alpha) &= P(y_2 | y_1)P(y_1 | y_0)P(y_0 | \alpha) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_2 - \alpha y_1)^2}{2\sigma^2}} \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_1 - \alpha y_0)^2}{2\sigma^2}} \times \frac{1}{\sqrt{2\pi\lambda}} e^{-\frac{y_0^2}{2\lambda}} \end{aligned}$$

Using **log likelihood**, we have:

$$\begin{aligned} \ln(P(y_2 | y_1)) &= \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \left(-\frac{(y_2 - \alpha y_1)^2}{2\sigma^2}\right) \\ \ln(P(y_1 | y_0)) &= \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \left(-\frac{(y_1 - \alpha y_0)^2}{2\sigma^2}\right) \\ \ln(P(y_0 | \alpha)) &= \ln\left(\frac{1}{\sqrt{2\pi\lambda}}\right) + \left(-\frac{y_0^2}{2\lambda}\right) \end{aligned}$$

Therefore, the log-likelihood function is:

$$\begin{aligned} \ln(P(y_2, y_1, y_0 | \alpha)) &= \ln(P(y_2 | y_1)) + \ln(P(y_1 | y_0)) + \ln(P(y_0 | \alpha)) \\ \implies \left[ \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \ln\left(\frac{1}{\sqrt{2\pi\lambda}}\right) \right] - \frac{(y_2 - \alpha y_1)^2 + (y_1 - \alpha y_0)^2}{2\sigma^2} - \frac{y_0^2}{2\lambda} \end{aligned}$$

To find the MLE of  $\alpha$ , we take the derivative of the log-likelihood with respect to  $\alpha$  and set it to zero:

$$\begin{aligned}\frac{d}{d\alpha} \ln(P(y_2, y_1, y_0 | \alpha)) &= -\frac{-2y_1(y_2 - \alpha y_1) - 2y_0(y_1 - \alpha y_0)}{2\sigma^2} \\ &= \frac{y_1(y_2 - \alpha y_1) + y_0(y_1 - \alpha y_0)}{\sigma^2} \\ 0 &= y_1(y_2 - \alpha y_1) + y_0(y_1 - \alpha y_0) \\ \implies y_1(y_2 - y_1\hat{\alpha}) &= -y_0(y_1 - y_0\hat{\alpha}) \\ y_1y_2 - y_1^2\hat{\alpha} &= -y_0y_1 + y_0^2\hat{\alpha} \\ \hat{\alpha}(y_1^2 + y_0^2) &= y_0y_1 + y_1y_2\end{aligned}$$

Therefore:  $\implies \hat{\alpha} = \frac{y_0y_1 + y_1y_2}{y_1^2 + y_0^2}$