ENGINEERING TRIPOS PART IIA

EIETL MODULE EXPERIMENT 3F3

RANDOM VARIABLES and RANDOM NUMBER GENERATION
Short Report

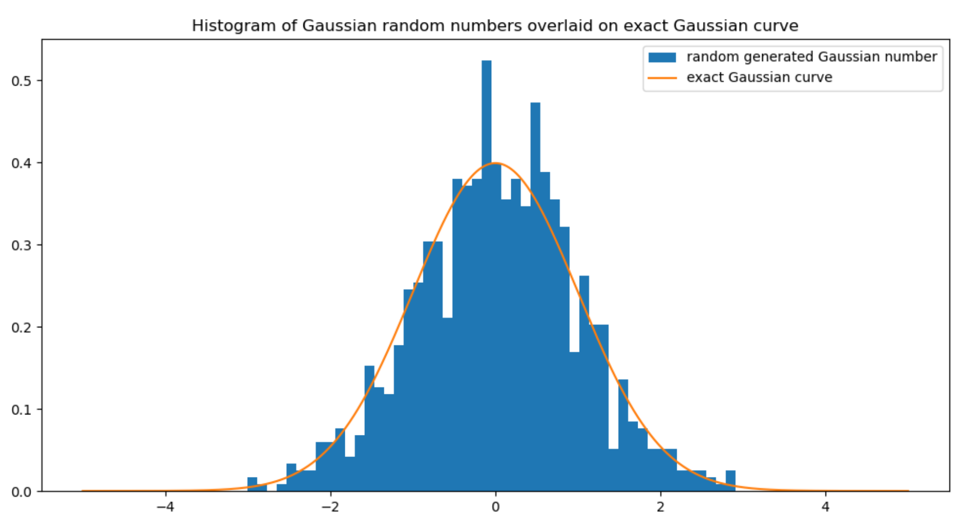
Name: Yongqing Jiang

College: Peterhouse

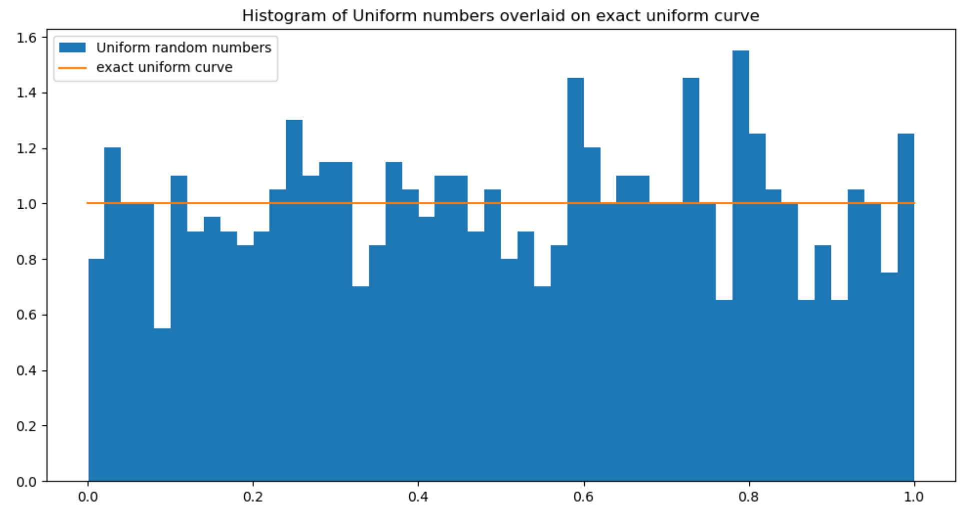
crsid: yj375

1. **UNIFORM AND NORMAL RANDOM VARIABLES.**

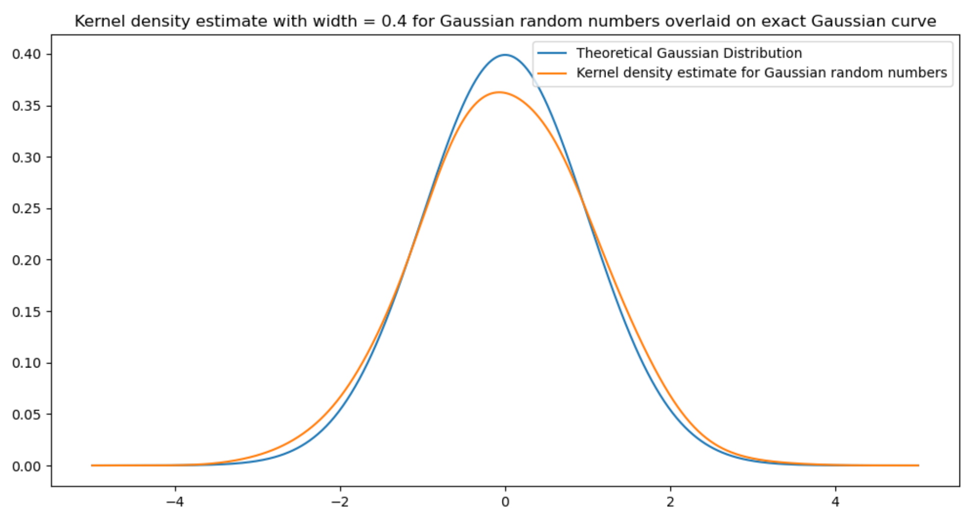
   Histogram of Gaussian random numbers overlaid on exact Gaussian curve (scaled):



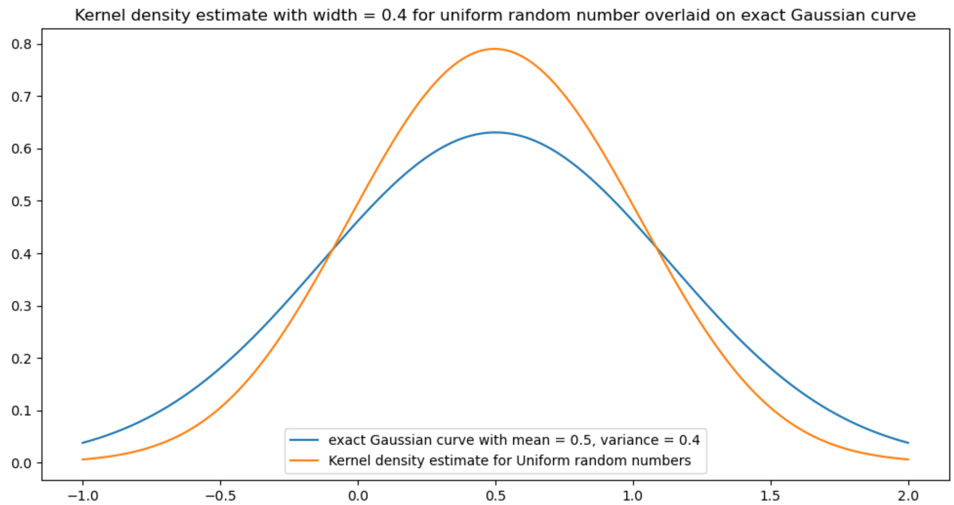
Histogram of Uniform random numbers overlaid on exact Uniform curve (scaled):



Kernel density estimate for Gaussian random numbers overlaid on exact Gaussian curve:



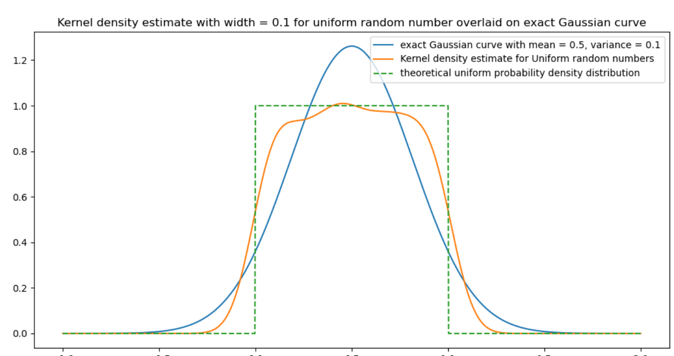
Kernel density estimate for Uniform random numbers overlaid on exact Gaussian curve:



Comment on the advantages and disadvantages of the kernel density method compared with the histogram method for estimation of a probability density from random samples:

According to the plots, kernel smoothing function works well with Gaussian random numbers. It is very close to the exact Gaussian curve. However, it does not perform well when it comes to uniform random numbers. We can see that the smoothing function of uniform random numbers is close to a Gaussian curve. This is due to large value of width applied in the kernel smoothing function. A large value of width underfits the random numbers and makes the smoothing function similar to the kernel function itself, which is a standard Gaussian distribution in this context.

If the width is set to smaller values, for example, 0.1 as shown below. The smoothing function provides a good approximation to the expected probability density function, which is a constant from 0 to 1. But it overfits the data, the density curve is no longer a constant value as expected. It can be seen that fine-tuning on the width is much harder for uniform random numbers.



To sum up, kernel density method works well with Gaussian random numbers, and becomes harder to fine-tune the parameter to fit uniform random numbers.

Theoretical mean and standard deviation calculation for uniform density as a function of N:

Suppose a set of N independent random samples are taken from a uniform distribution:

$$x^{(i)} \sim p_U(x) = \frac{1}{N}, i = 1, ..., N$$

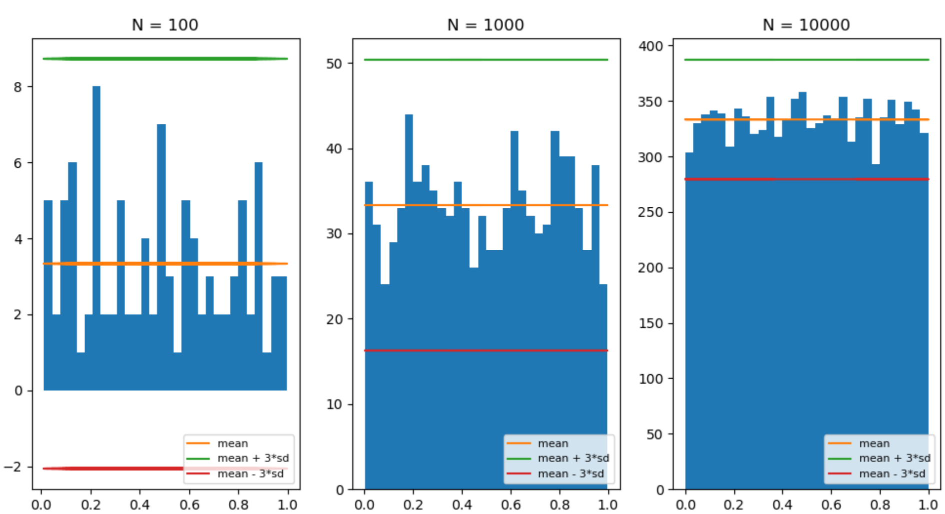
The probability that a sample $x^{(j)}$ lies within a particular bin of the histogram is:

$$p_j = \int_{c_j-\delta/2}^{c_j+\delta/2} p(x)dx = \frac{\delta}{N}$$

in which δ represents the size of a bin.
Thus, the mean of the count data in bin j is $Np_j = \delta$. The variance is $Np_j(1 - p_j) = \delta(1 - \frac{\delta}{N})$

Explain behaviour as N becomes large: Assuming a constant bin size, it is clear that the mean is not affected by N, and the variance decreases as N increases.

Plot of histograms for $N = 100$, $N = 1000$ and $N = 10000$ with theoretical mean and ± 3 standard deviation lines:



Are your histogram results consistent with the multinomial distribution theory?
For the case that N = 1000 and 10000, the histogram results are consistent with the multinomial distribution theory. Considering the counts of each bin as a random variable, its distribution approaches to Gaussian distribution as $N \to \infty$ according to Central Limit Theorem.
For a Gaussian distribution, 99% of random variable lies within the interval $(\mu - 3\sigma, \mu + 3\sigma)$. From the plots that N = 1000 and N = 10000, we can see that

it is true.
However, the plot of N = 100 gives a negative value of $\mu - 3\sigma$. Since $\sigma^2 = \delta(1 - \frac{\delta}{N})$, a small N gives a large σ. This implies that histogram estimatio is not suitable for a set of random numbers with a small size.

2. **FUNCTIONS OF RANDOM VARIABLES**

For normally distributed $\mathcal{N}(x|0,1)$ random variables, take $y = f(x) = ax + b$. Calculate $p(y)$ using the Jacobian formula:

The Jacobian formula provided by lab handout:

$$p(y) = \left.\frac{p(x)}{|dy/dx|}\right|_{x=f^{-1}(y)}$$

$y = f(x) = ax + b \Rightarrow x = f^{-1}(y) = \frac{y-b}{a}$, and $dy/dx = a$. With the standard Gaussian distribution:

$$p(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

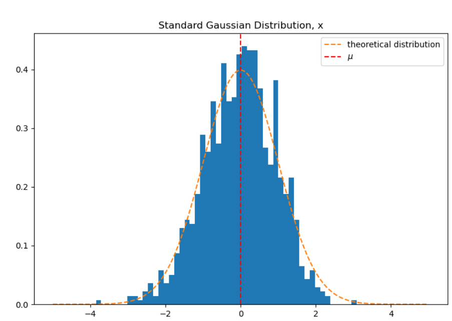
$$p(y) = \frac{1}{|a|} \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{(\frac{y-b}{a})^2}{2}} = \frac{1}{\sqrt{2\pi a^2}} e^{-\frac{(y-b)^2}{2a^2}}$$

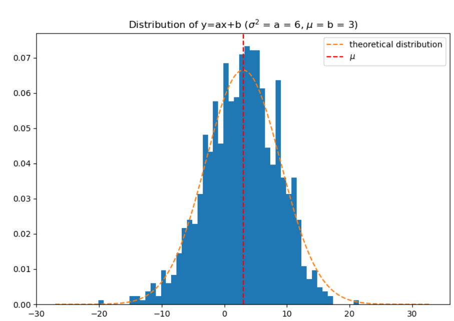
Explain how this is linked to the general normal density with non-zero mean and non-unity variance:
For a general normal distribution $\mathcal{N}(x|\mu, \sigma^2)$, the probability density function is:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

We can see that the transformed random variable is $\mathcal{N}(y|b, a^2)$.

Verify this formula by transforming a large collection of random samples $x^{(i)}$ to give $y^{(i)} = f(x^{(i)})$, histogramming the resulting y samples, and overlaying a plot of your formula calculated using the Jacobian:





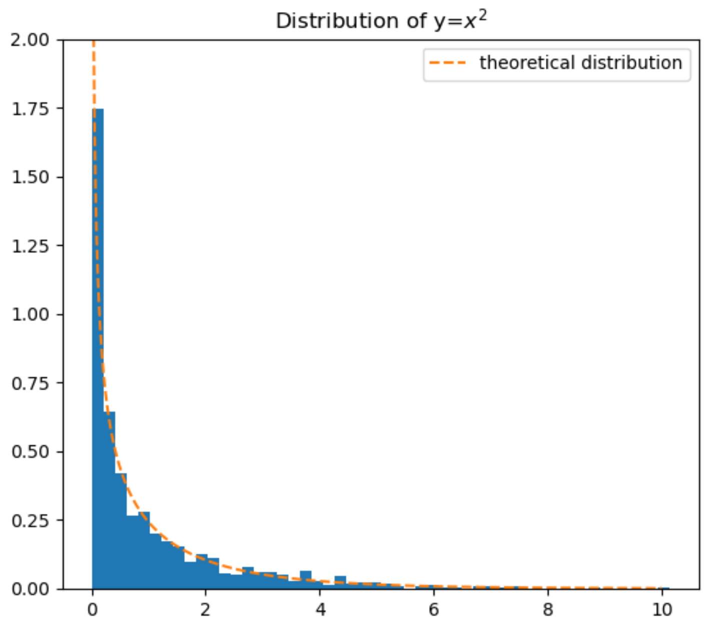
We can see that the histogram of transformed random samples with a = 6 and b = 3 fits well with the probability density function with variance 6 and mean 3.

Now take $p(x) = \mathcal{N}(x|0,1)$ and $f(x) = x^2$. Calculate $p(y)$ using the Jacobian formula:

Note that $y = f(x) = x^2 \Rightarrow x = f^{-1}(y) = \pm\sqrt{y}$. Thus, $dy/dx = 2x = 2 \pm \sqrt{y}$. The probability density function of the transformed random variable is:

$$\begin{aligned} p(y) &= \frac{p(x)}{|dy/dx|}\bigg|_{x=f^{-1}(y)} + \frac{p(x)}{|dy/dx|}\bigg|_{x=-f^{-1}(y)} \\ &= \frac{1}{2\sqrt{y}} \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{(\sqrt{y})^2}{2}} + \frac{1}{2\sqrt{y}} \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{(-\sqrt{y})^2}{2}} \\ &= \frac{1}{\sqrt{2\pi y}} e^{-\frac{y}{2}} \end{aligned}$$

Verify your result by histogramming of transformed random samples:



3. **INVERSE CDF METHOD**

Calculate the CDF and the inverse CDF for the exponential distribution:

The probability density function of an exponential distribution Y with mean 1 is:

$$f_Y(y) = e^{-y}$$

The corresponding cdf is found by integration:

$$F_Y(y) = \int_0^y f_Y(t)dt = \int_0^y e^{-t}dt = 1 - e^{-y}$$

The inverse of this function is found by: $x = F_Y(y), y = F_Y^{-1}(x)$.

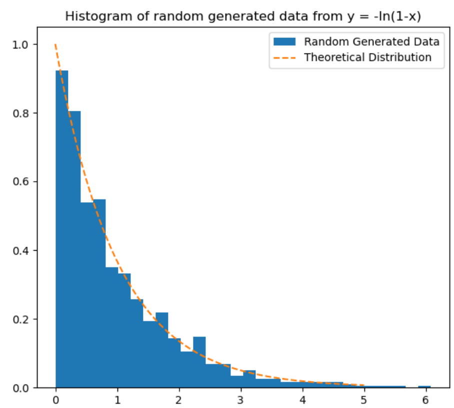
$$F_Y^{-1}(x) = -\ln(1 - x)$$

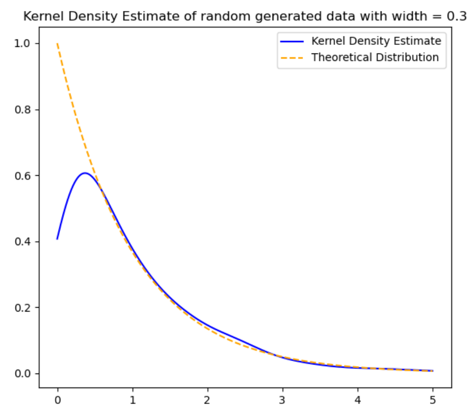
Matlab/Python code for inverse CDF method for generating samples from the exponential distribution:

Listing 1: PYTHON CODE

```
x_data = np.random.uniform(0,1,1000)
y_data = -np.log(1-x_data)

y = np.linspace(0,5,1000)
exp_theoretical = np.e**(-y)
```

Plot histograms/ kernel density estimates and overlay them on the desired exponential density:





4. **SIMULATION FROM A 'NON-STANDARD' DENSITY.**

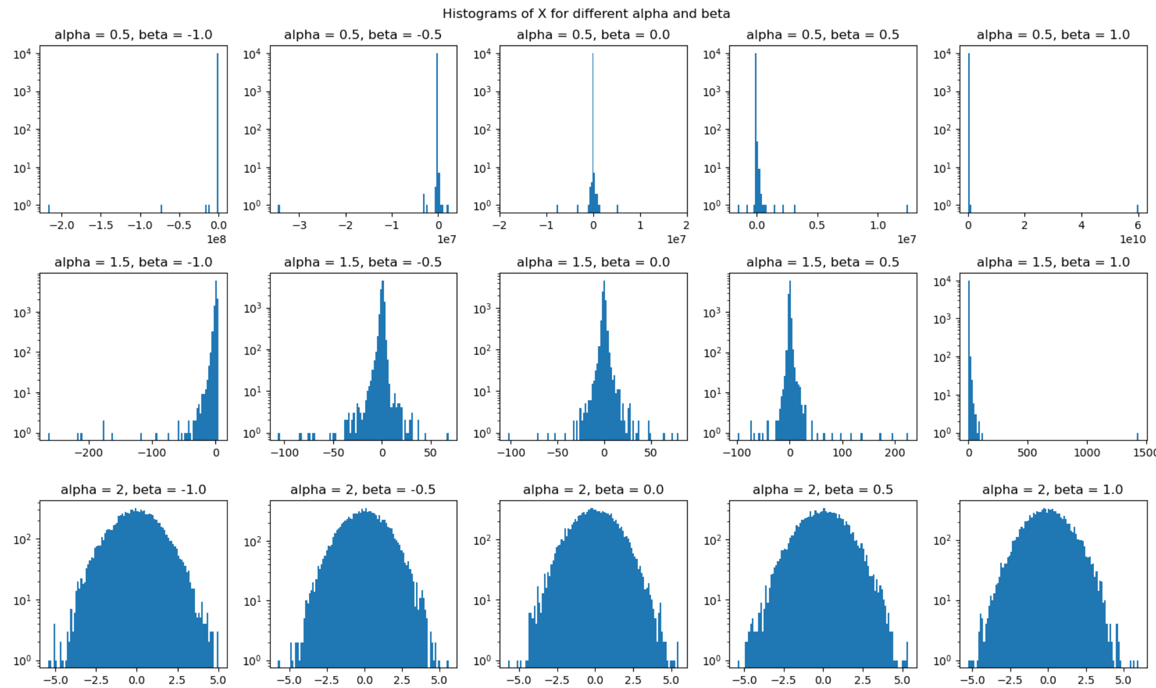
   Matlab/Python code to generate N random numbers drawn from the distribution of X:

Listing 2: PYTHON CODE

```
def generate_x(alpha, beta, N):
    def b(alpha, beta):
        temp1 = beta*np.tan(np.pi*alpha/2)
        temp2 = np.arctan(temp1)
        return temp2/alpha

    def s(alpha, beta):
        temp1 = beta*np.tan(np.pi*alpha/2)
        temp2 = 1 + temp1**2
        temp3 = 1/(2*alpha)
        return temp2**(temp3)

    u = np.random.uniform(-np.pi/2,np.pi/2,N)

    v = -np.log(1-np.random.uniform(0,1,N))

    b = b(alpha, beta)
    s = s(alpha, beta)
```

```
def x(b,s,u,v):
    temp1 = np.sin(alpha*(u + b))
    temp2 = (np.cos(u))**(1/alpha)
    temp3 = np.cos(u - alpha*(u + b))/v
    temp4 = (1-alpha)/alpha
    return s*temp1/temp2*(temp3**temp4)
return x(b,s,u,v)
```

Plot some histogram density estimates with $\alpha = 0.5, 1.5$ and several values of β.



Hence comment on the interpretation of the parameters α and β.
From the figures, we can see that α determines the extent of the data with extreme values. A small value of α gives a distribution that has more extreme values, while a large value of α gives a distribution that is more concentrated around the mean. When $\alpha = 2$, we can see that the distribution has become a standard Gaussian distribution.
The value of β determines the skewness of the distribution. A positive β skews the distributio to the right, and a negative β skews the distribution to the left.

A Link to worked files

The worked python files are uploaded to a repository which can be found at: `https://github.com/OliverJiang2025/3F3_lab.git`