Assignment 1: Random Variables

Avneesh Mishra [*]
avneesh.mishra@research.iiit.ac.in

September 20, 2021

Contents

[*] MS2k21 - Reg. No. 2021701032

List of Figures

1 PMF: Finite and Infinite range

Every **PMF** (Probability Mass Function) has to obey the following properties

$$P_X(x_k) \geq 0 \, \forall \, x_k \in R_X = \{x_1, x_2, \cdots\} \tag{1.1}$$

$$\sum_{x_i \in R_X} P_X(x_i) = 1 \tag{1.2}$$

Where R_X is the range of values that X, a discrete random variable, can take. Assumptions for this question are as follows

- If R_X has finite elements, the PMF has **finite** range, else if R_X has *countably infinite*[1] elements, the PMF has **infinite** range.

1.1 Finite Range: Bernouli Distribution

A PMF $P_X(x_k) = P(X = x_k)$, with finite range is assumed to have a finite number of elements that the discrete random variable X can take. A simple example is the *Bernouli distribution* where $R_X = \{0, 1\}$.

$$P_X(x;p) = \begin{cases} p & \text{if } x \text{ is } 1 \\ 1-p & \text{if } x \text{ is } 0 \end{cases} \tag{1.3}$$

Where $0 \leq p \leq 1$, which also means that $1 \geq 1 - p \geq 0$. This proves 1.1: as the function P_X can only yield p or $1 - p$, both being in range $[0, 1]$. To prove 1.2, we can do

$$\sum_{v \in \{0,1\}} P_X(v) = P(X = 0) + P(X = 1) = (1 - p) + p = 1 \tag{1.4}$$

Hence, a finite range Probability Mass Function is achieved using Bernouli's distribution. Example is a coin toss (with $p = 0.5$).

1.2 Infinite Range: Exponentially decaying distribution

Consider an experiment involving multiple coin tosses. We are interested in calculating the probability of all the outcomes being `HEAD`. The discrete random variable X is the trial number with consecutive heads (for example 1 trial with 1 head, 2 trial with 2 heads, and so on). The range of values for X is the countably infinite set of *natural numbers*, that is $R_X = \mathbb{N}$. Basically, $P(X = i)$ is the probability that we do the coin toss i times and we get heads as the outcome for every trial.

It is clear that $P(X = 1) = 0.5$: probability of head in a single coin toss and $P(X = 2) = 0.5^2 = 0.25$: probability of two coin tosses giving heads. This can be extended to any number of coin tosses, giving the probability function

$$P(X = k) = P_X(k) = (0.5)^k \tag{1.5}$$

[1]Elements are in one-to-one correspondence with natural numbers

Since every power of 0.5 is positive, property 1.1 holds good. The proof of property 1.2 can be done by the sum of infinite geometric progression, the general form is given below

$$S = \sum_{i=0}^{n-1} a\, r^i = \frac{a}{1-r} \tag{1.6}$$

Consider $a = r = 0.5$, the sum of all probabilities can be calculated as following

$$\sum_{k \in R_X = \mathbb{N}} P_X(k) = \sum_{k=1}^{\infty} 0.5^k = \sum_{k=1}^{\infty} 0.5 \times 0.5^{k-1} = \frac{0.5}{1-0.5} = 1 \tag{1.7}$$

This proves that P_X is a probability distribution. Its value exponentially decreases (decays) to 0.

2 Variance of Uniform Density Function

The covariance of a function is defined by

$$Cov(X,Y) = E\left[(X - E(X))\,(Y - E(Y))\right] = E[XY] - E[X]E[Y] \tag{2.1}$$

The variance is given by

$$Var(X) = Cov(X,X) = E[X^2] - (E[X])^2 \tag{2.2}$$

The uniform distribution is (given that $b > a$)

$$U(x;a,b) = \begin{cases} \frac{1}{b-a} & x \in [a,b] \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

The expected value, $E[X]$, for the uniform distribution can be calculated as follows

$$E_U[X] = \int_{-\infty}^{\infty} xU(x)\mathrm{d}x = \frac{1}{b-a}\int_a^b x\mathrm{d}x = \frac{1}{b-a}\left[\frac{x^2}{2}\right]_a^b = \frac{b+a}{2} \tag{2.4}$$

The value of $(E[X^2])$ can be calculated as follows

$$E[X^2] = \int_{-\infty}^{+\infty} x^2 U(x)\mathrm{d}x = \frac{1}{b-a}\int_a^b x^2\mathrm{d}x = \frac{1}{b-a}\left[\frac{x^3}{3}\right]_a^b = \frac{b^3 - a^3}{3(b-a)} = \frac{a^2+ab+b^2}{3} \tag{2.5}$$

Using equations 2.4 and 2.5 to calculate variance according to 2.2 gives

$$\begin{gathered} \mathrm{Var}_U(X) = E[X^2] - (E[X])^2 = \left(\tfrac{a^2+ab+b^2}{3}\right) - \left(\tfrac{(b+a)^2}{4}\right) \\ \Rightarrow \mathrm{Var}(X) = \tfrac{4a^2+4ab+4b^2-3a^2-6ab-3b^2}{12} = \tfrac{a^2-2ab+b^2}{12} = \tfrac{(b-a)^2}{12} \end{gathered} \tag{2.6}$$

Therefore, equation 2.6 proves that

$$\mathrm{Var}(X) = \frac{(b-a)^2}{12} \tag{2.7}$$

3 Same mean and variance for different PDFs

Consider the uniform distribution U (same as 2.3)

$$U(x;a,b) = \begin{cases} \frac{1}{b-a} & \text{if } x \in [a,b] \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

Consider the normal distribution N given by

$$N(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{3.2}$$

As observable, the normal distribution is explicitly parameterized by mean μ and variance σ^2, whereas the uniform distribution has mean $\mu_U = \frac{a+b}{2}$ and variance $\sigma_U^2 = \frac{(b-a)^2}{12}$ (as proven by 2.6).

We can select any a and b for a uniform distribution (as long as $b > a$), calculate the mean and variance, and then create a corresponding normal distribution. This is shown in Figure 1. The

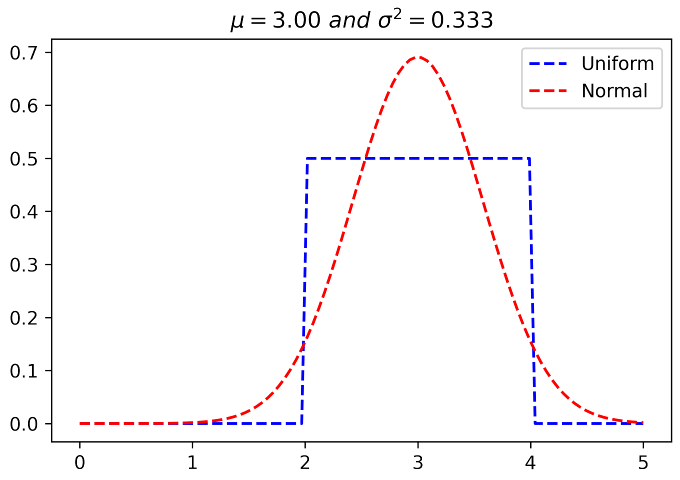


Figure 1: Normal and Uniform distribution having the same mean and variance
The blue line is for a uniform distribution with $a = 2$ and $b = 4$. The red line is for a normal distribution with $\mu = 3$ and $\sigma^2 = \frac{1}{3}$

code responsible for this figure is described in appendix A.1.

4 Simplified variance for discrete random variables

Reiterating the equation which has to be proven

$$Var(X) = Cov(X,X) = E\left[(X - E[X])^2\right] = E[X^2] - (E[X])^2 \tag{2.2 revisited}$$

Starting from the left side of 2.2 revisited, we get

$$Var(X) = Cov(X, X) = E\left[(X - E[X])^2\right] \tag{4.1}$$

The expectation for a function of a discrete random variable X (whose possible values are given by R_X), is given by

$$E_{x\sim X}\left[f(X)\right] = \sum_{x\in R_X} f(x)P(X = x) \tag{4.2}$$

Simplifying the equation 4.1 using the following known conditions

1. The value of $E[X]$ is the mean of the discrete random variable given by
$$E[X] = \sum_{x\in R_X} xP_X(x) = \mu \tag{4.3}$$
2. μ is a constant for a given probability distribution P_X
3. $\sum_{x\in R_X} P_X(x) = 1$ because P_X is a probability mass function

we get

$$\begin{aligned}
E\left[(X - E[X])^2\right] &= E\left[(X-\mu)^2\right] = \sum_{x\in R_X} (x-\mu)^2 P_X(x) = \sum_{x\in R_X}\left(x^2 + \mu^2 - 2x\mu\right) P_X(x) \\
&= \sum_{x\in R_X} x^2 P_X(x) + \mu^2 \sum_{x\in R_X} P_X(x) - 2\mu \sum_{x\in R_X} xP_X(x) \\
&= E_{x\sim X}[X^2] + \mu^2\cdot 1 - 2\mu\cdot\mu = E[X^2] + \mu^2 - 2\mu^2 = E[X^2] - \mu^2 \\
&= E[X^2] - (E[X])^2
\end{aligned} \tag{4.4}$$

The equation 4.4 proves 2.2 revisited, we therefore obtain

$$Var(X) = E[X^2] - (E[X])^2 \tag{4.5}$$

5 Mean and Variance of continuous normal PDF

The normal probability density function for a continuous random variable X is given by

$$N(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{3.2 revisited}$$

Note that the expectation of a function of a continuous random variable is given by

$$E_{x\sim X}[f(X)] = \int_x f(x)p(x)\,\mathrm{d}x \tag{5.1}$$

Where $p(x)$ is the probability density function of X (in this case, it's described by $N(x;\mu,\sigma^2)$ in 3.2 revisited). We shall also use a commonly known identity

$$\int_{-\infty}^{\infty} e^{-x^2}\mathrm{d}x = \sqrt{\pi} \tag{5.2}$$

This is proven as A.2.6 in Appendix A.2. Another known identity we will use is

$$\int_{-\infty}^{\infty} x^2 e^{-x^2}\,\mathrm{d}x = \frac{\sqrt{\pi}}{2} \tag{5.3}$$

This is proven in Appendix A.3. We can proceed to prove the mean of $N(x;\mu,\sigma^2)$ first.

5.1 Proving Mean of Normal PDF

The mean value of a PDF is given as $E(X)$, for the normal PDF, we get

$$\begin{aligned} E_{X\sim N}[X] &= \int_{-\infty}^{\infty} x\, N(x;\mu,\sigma^2)\mathrm{d}x = \int_{-\infty}^{\infty} \frac{x}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\mathrm{d}x \\ &= \int_{-\infty}^{\infty} \frac{x}{\sqrt{2\pi\sigma^2}}\, e^{-\left(\frac{(x-\mu)}{\sigma\sqrt{2}}\right)^2}\mathrm{d}x \end{aligned} \tag{5.4}$$

We can solve 5.4 by substituting a function of x for v (use the equations below)

$$\begin{aligned} \frac{x-\mu}{\sigma\sqrt{2}} = v \Rightarrow x &= \left(\sigma\sqrt{2}\right) v + \mu \\ \Rightarrow \mathrm{d}x &= \sigma\sqrt{2}\,\mathrm{d}v \\ x \to (-\infty,\infty) &\Rightarrow v \to (-\infty,\infty) \end{aligned} \tag{5.5}$$

Substituting this in 5.4, we get

$$\begin{aligned} E_{X\sim N}[X] &= \int_{-\infty}^{\infty} \frac{x}{\sqrt{2\pi\sigma^2}}\, e^{-\left(\frac{(x-\mu)}{\sigma\sqrt{2}}\right)^2}\mathrm{d}x = \frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{\infty}\left(\sigma\sqrt{2}\,v+\mu\right)e^{-v^2}\sigma\sqrt{2}\,\mathrm{d}v \\ &= \frac{1}{\sqrt{\pi}}\left[\sigma\sqrt{2}\int_{-\infty}^{\infty} v e^{-v^2}\mathrm{d}v + \mu\int_{-\infty}^{\infty} e^{-v^2}\mathrm{d}v\right] = \frac{\mu}{\sqrt{\pi}}\int_{-\infty}^{\infty} e^{-v^2}\mathrm{d}v \\ &= \frac{\mu}{\sqrt{\pi}}\sqrt{\pi} = \mu \end{aligned} \tag{5.6}$$

Note that $\int_{-\infty}^{\infty} v e^{-v^2}\mathrm{d}v = 0$ as $f_1(v) = v e^{-v^2}$ is an odd function ($f_1(-v) = -f_1(v)$), so the integral over $(-\infty,\infty)$ is 0 (all values cancel out). The equation 5.6 proves that indeed for a normal distribution, the mean is μ, that is

$$E_{X\sim N}[X] = \mu \tag{5.7}$$

We can now prove the variance for a normal distribution

5.2 Proving Variance of Normal PDF

The variance of a Normal Probability Density Function for a continuous random variable is given by (consider $E[X]=\mu$ as proven in subsection 5.1)

$$Var(X) = E\left[(X - E[X])^2\right] = E_N\left[(X-\mu)^2\right] = \int_{-\infty}^{\infty}(x-\mu)^2\, N(x;\mu,\sigma^2)dx \tag{5.8}$$

To solve this, we make the same assumption as in subsection 5.1.

$$\begin{aligned}\frac{x-\mu}{\sigma\sqrt{2}} = v \Rightarrow x = \left(\sigma\sqrt{2}\right)v + \mu \Rightarrow (x-\mu)^2 = 2\sigma^2 v^2 \\ \to \mathrm{d}x = \sigma\sqrt{2}\,\mathrm{d}v \\ x \to (-\infty,\infty) \Rightarrow v \to (-\infty,\infty)\end{aligned} \tag{5.9}$$

Now, solving for variance becomes

$$\begin{aligned}Var(X) &= \int_{-\infty}^{\infty} \frac{(x-\mu)^2}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{(x-\mu)}{\sigma\sqrt{2}}\right)^2} \mathrm{d}x = \int_{-\infty}^{\infty} \frac{2\sigma^2 v^2}{\sqrt{2\pi\sigma^2}} e^{-v^2} \sigma\sqrt{2}\,\mathrm{d}v \\ &= \frac{2\sigma^2}{\sqrt{\pi}} \int_{-\infty}^{\infty} v^2 e^{-v^2}\,\mathrm{d}v = \frac{2\sigma^2}{\sqrt{\pi}} \frac{\sqrt{\pi}}{2} = \sigma^2\end{aligned} \tag{5.10}$$

The result $\int_{-\infty}^{\infty} v^2 e^{-v^2}\,\mathrm{d}v$ is proved in Appendix A.3. The above result proves that for a normal distribution, the variance is σ^2, that is

$$Var(X) = E_N\left[(X - E_N[X])^2\right] = \sigma^2 \tag{5.11}$$

6 Using inverse of CDF to generate different PDFs

All results are described in Figure 2.

6.1 Normal Distribution

The normal distribution's Probability Density Function (PDF) is given by

$$N(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{6.1}$$

The Cumulative Density Function (CDF) for the above equation is given by

$$C_N(t;\mu,\sigma^2) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{t-\mu}{\sqrt{2\sigma^2}}\right)\right) \tag{6.2}$$

Where erf is the *error function*, given by

$$\mathrm{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-x^2}\,\mathrm{d}x \tag{6.3}$$

The CDF equation $y = C_N(x;\mu,\sigma^2)$ can be inverted (get x for given y) as follows

$$\begin{aligned}y = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x-\mu}{\sqrt{2\sigma^2}}\right)\right) \Rightarrow \mathrm{erf}\left(\frac{x-\mu}{\sqrt{2\sigma^2}}\right) = 2y - 1 \\ \Rightarrow \frac{x-\mu}{\sqrt{2\sigma^2}} = \mathrm{erfinv}\,(2y-1) \Rightarrow x = \mu + \sqrt{2\sigma^2}\,\mathrm{erfinv}\,(2y-1)\end{aligned} \tag{6.4}$$

Note that erfinv is the *inverse of the error function.*

Generating PDF through this is explored in Figure 2a with the corresponding code in A.4.1.

6.2 Rayleigh Distribution

The Rayleigh distribution's Probability Density Function is given by

$$R(x;\sigma^2) = \frac{x}{\sigma^2}\exp\left(\frac{-x^2}{2\sigma^2}\right) \quad x \geq 0 \tag{6.5}$$

The Cumulative Density Function for the above equation is given by

$$C_R(t;\sigma^2) = 1 - \exp\left(\frac{-t^2}{2\sigma^2}\right) \quad t \geq 0 \tag{6.6}$$

The CDF equation $y = C_R(x;\sigma^2)$ can be inverted as follows

$$\begin{aligned} y &= 1 - \exp\left(\frac{-x^2}{2\sigma^2}\right) \Rightarrow (1-y) = \exp\left(\frac{-x^2}{2\sigma^2}\right) \\ &\Rightarrow \frac{x^2}{2\sigma^2} = -\ln(1-y) \Rightarrow x = \sigma\sqrt{-2\ln(1-y)} \\ &\Rightarrow x = \sqrt{2\sigma^2\,\ln\left(\frac{1}{1-y}\right)} \end{aligned} \tag{6.7}$$

Generating PDF through this is explored in Figure 2b with the corresponding code in A.4.2.

6.3 Exponential Distribution

The Exponential distribution's Probability Density Function is given by

$$E(x;\lambda) = \lambda e^{-\lambda x} \quad x \geq 0 \tag{6.8}$$

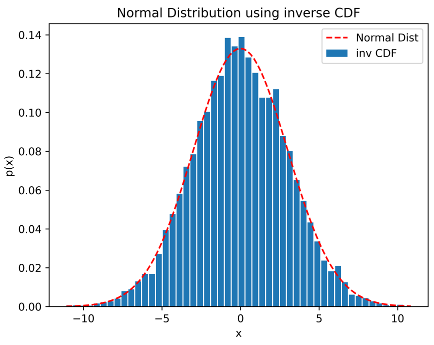
The Cumulative Density Function for the above equation is given by

$$C_E(t;\lambda) = 1 - e^{-\lambda t} \tag{6.9}$$

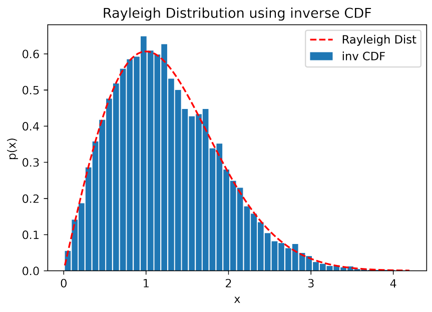
The CDF equation $y = C_E(x;\lambda)$ can be inverted as follows

$$\begin{aligned} y &= 1 - e^{-\lambda x} \Rightarrow e^{-\lambda x} = 1 - y \Rightarrow x = \frac{-1}{\lambda}\ln(1-y) \\ &\Rightarrow x = \frac{1}{\lambda}\ln\left(\frac{1}{1-y}\right) \end{aligned} \tag{6.10}$$

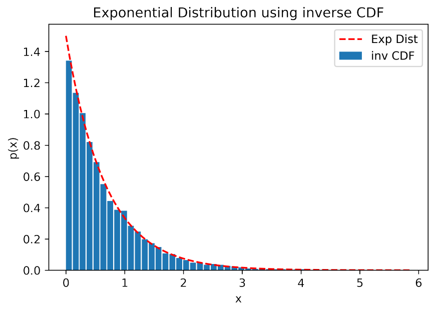
Generating PDF through this is explored in Figure 2c with the corresponding code in A.4.3.



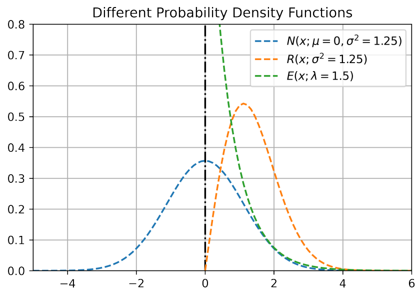
(a) $N(x; \mu = 0, \sigma^2 = 9)$



(b) $R(x; \sigma^2 = 1)$



(c) $E(x; \lambda = 1.5)$



(d) N, R and E PDFs

Figure 2: Using the inverse of CDFs to generate a PDF

Figure 2a consists of the *normal distribution* generated through inverse CDF in 50 bins (as blue bars) and the actual normal PDF through function (as red dotted line). Code for this is in A.4.1. Figure 2b consists of the *rayleigh distribution* generated through inverse of CDF (as blue bars) and the actual rayleigh PDF through function (as red dotted line). Code for this is in A.4.2. Figure 2c consists of the *exponential distribution* generated through inverse of CDF (as blue bars) and the actual exponential PDF through function (as red dotted line). Code for this is in A.4.3. Figure 2d consists of all three PDFs (with different parameters) in one plot, just for comparison. The *Normal PDF* shown in blue line has $\mu = 0$ and $\sigma^2 = 1.25$, the *Rayleigh PDF* shown in orange line has $\sigma^2 = 1.25$, and the *Exponential PDF* shown in green line has $\lambda = 1.5$. The code responsible for this is mentioned in Appendix A.4.4

7 Experimenting sum of samples from Uniform Distribution

The result is demonstrated in Figure 3 and code can be found in Appendix A.5.

The experiment is to generate multiple random numbers from a *Uniform Distribution* (0 to 1), sum them, store the result in a buffer and then do the same again. The resultant buffer has to be viewed as a normalized histogram.

The resultant distribution achieved is called an *Irwin-Hall distribution* which is defined as the sum of n independent and identically distributed $U(0,1)$ random variables.

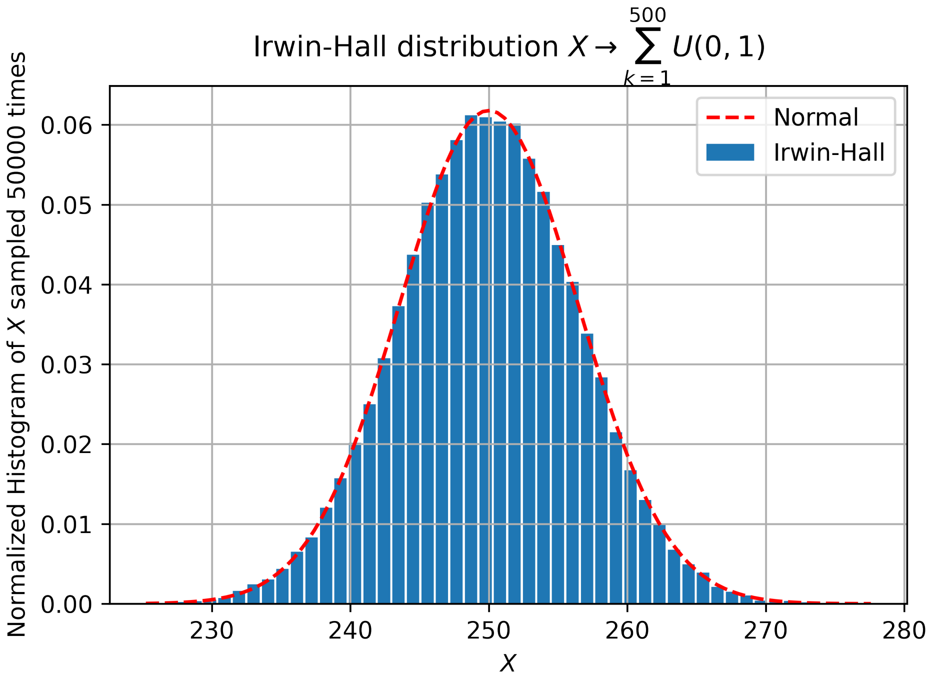


Figure 3: Irwin-Hall Distribution
This figure is obtained by visualizing the histogram of multiple experiments. In each experiment, 500 random numbers were generated from $U(0,1)$ (uniform distribution) and added, the result being returned. The histogram is in blue, the red line is a *normal distribution* with $\mu = \frac{N}{2} = 250$ and $\sigma^2 = \frac{N}{12} = 41.\bar{6}$. The code responsible for this figure can be found in Appendix A A.5.

Irwin-Hall Distribution

The Irwin-Hall distribution is defined as the distribution of a continuous random variable X, where

$$X = \sum_{k=1}^{n} U(k;0,1) \tag{7.1}$$

Basically, X is the sum of n independent and identically distributed uniform distributions (spanning 0 to 1). The probability density function is given by

$$f_X(x;n) = \frac{1}{2\,(n-1)!}\sum_{k=0}^{n}(-1)^k \binom{n}{k} (x-k)^{n-1}\,\text{sgn}(x-k) \tag{7.2}$$

Where sgn is the sign function defined by

$$\text{sgn}(x-k) = \begin{cases} -1 & x < k \\ 0 & x = k \\ +1 & x > k \end{cases} \tag{7.3}$$

This is **different** from the normal distribution, but it can be *approximated* to one by using $\mu = \frac{n}{2}$ and $\sigma^2 = \frac{n}{12}$ where n is the number of times the summation is done. Such an approximated fit is shown in Figure 3.

A Appendix

A.1 Code for P3

```python
# %% Import everything
import numpy as np
from matplotlib import pyplot as plt

# %% Declare functions

# Generate uniform distribution function
def gen_uni_dist(a, b):
    """
    Generates a callable uniform distribution for a continuous random
    variable. Pass the 'a' and 'b' values ('b' > 'a').
    """
    if (a > b):
        return gen_uni_dist(b, a)
    cv = (1/(b-a))
    ud_func = lambda x: cv if a <= x and x <= b else 0
    return ud_func

# Generate normal distribution function
def gen_norm_dist(mu, si2):
    """
    Generates a callable normal distribution for a continuous random
    variable. Pass the 'mu' (mean) and 'si2' (variance = square of the
    standard deviation)
    """
    nd_func = lambda x: 1/( np.sqrt(2*np.pi*si2) ) * \
        np.exp( -(x-mu)**2/(2*si2) )
    return nd_func

# %% Plot everything
if __name__ == "__main__":
    # Configurations
    a, b = 2, 4
    mu, si2 = (a+b)/2, ((b-a)**2)/12
    xlim = [0, 5]
    # Actual code to generate data to be plotted
    xvals = np.linspace(xlim[0], xlim[1], 100)
    uni_dist = gen_uni_dist(a, b)
    uni_vals = np.array([uni_dist(xv) for xv in xvals])
    norm_vals = gen_norm_dist(mu, si2)(xvals)
    # Plot everything
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.plot(xvals, uni_vals, 'b--', label='Uniform')
    ax.plot(xvals, norm_vals, 'r--', label='Normal')
    ax.legend()
    ax.set_title(fr"$\mu={mu:.2f}\,\,and\,\,\sigma^2={si2:.3f}$")
    fig.savefig("plot_p3.png", dpi=600)
    plt.show()

# %%
```

Listing 1: Code to generate Figure 1

A.2 Evaluating integral of e^{-x^2} over $(-\infty, \infty)$

Here, we shall prove the result

$$\int_{-\infty}^{\infty} e^{-x^2} \mathrm{d}x = \sqrt{\pi} \tag{A.2.1}$$

To prove A.2.1, consider the integral to be I, that is

$$I = \int_{-\infty}^{\infty} e^{-x^2} \mathrm{d}x \tag{A.2.2}$$

Consider evaluating the integral below (we'll introduce evaluation in multiple dimensions)

$$\begin{aligned} J &= \int_{y=-\infty}^{\infty} \int_{x=-\infty}^{\infty} e^{-x^2-y^2} \mathrm{d}x\, \mathrm{d}y = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-x^2} e^{-y^2} \mathrm{d}x\, \mathrm{d}y \\ &= \int_{-\infty}^{\infty} e^{-y^2} \left(\int_{-\infty}^{\infty} e^{-x^2} \mathrm{d}x \right) \mathrm{d}y = \left(\int_{-\infty}^{\infty} e^{-x^2} \mathrm{d}x \right) \left(\int_{-\infty}^{\infty} e^{-y^2} \mathrm{d}y \right) \\ &= I \cdot I = I^2 \end{aligned} \tag{A.2.3}$$

We will evaluate a value for J, then, using result of $J = I^2$ (from A.2.3), we will solve for I. It is also useful to recall the conversion from cartesian system to polar coordinate system

$$x = r\cos(\theta);\ y = r\sin(\theta);\ \mathrm{d}x\, \mathrm{d}y = r\mathrm{d}\theta \mathrm{d}r;\ r \to (0, \infty);\ \theta \to (0, 2\pi) \tag{A.2.4}$$

Using this, we can compute J as follows

$$\begin{aligned} J &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-x^2} e^{-y^2} \mathrm{d}x\, \mathrm{d}y = \int_{0}^{\infty} \int_{0}^{2\pi} e^{-r^2} r\mathrm{d}\theta \mathrm{d}r \\ &= \int_{0}^{\infty} e^{-r^2} r \left(\int_{0}^{2\pi} \mathrm{d}\theta \right) dr = 2\pi \int_{0}^{\infty} e^{-r^2} r dr = 2\pi \frac{\left[-e^{-r^2} \right]_0^{\infty}}{2} \\ &= 2\pi \frac{[(-0) - (-1)]}{2} = \pi \end{aligned} \tag{A.2.5}$$

Therefore, from A.2.5, we have $J = \pi$. Since $J = I^2$ (from A.2.3), we have $I = \sqrt{\pi}$, that is

$$I = \int_{-\infty}^{\infty} e^{-x^2} \mathrm{d}x = \sqrt{\pi} \tag{A.2.6}$$

A.3 Evaluating integral of $x^2 e^{-x^2}$ over $(-\infty, \infty)$

Here, we shall prove the result

$$\int_{-\infty}^{\infty} x^2 e^{-x^2}\,\mathrm{d}x = \frac{\sqrt{\pi}}{2} \tag{A.3.1}$$

To prove this, let

$$I = \int_{-\infty}^{\infty} x^2 e^{-x^2}\,\mathrm{d}x \tag{A.3.2}$$

To evaluate the integral, we use the identity

$$\int u\,\mathrm{d}v = uv - \int v\,\mathrm{d}u \tag{A.3.3}$$

Let us use the following substitution

$$u = x;\ \mathrm{d}v = xe^{-x^2}\mathrm{d}x \Rightarrow \mathrm{d}u = \mathrm{d}x;\ v = \frac{-e^{-x^2}}{2} \tag{A.3.4}$$

Using A.3.3 to solve for I, we can get

$$I = \int_{-\infty}^{\infty} (x)\left(xe^{-x^2}\right)\,\mathrm{d}x = \left[\frac{-xe^{-x^2}}{2}\right]_{-\infty}^{\infty} + \frac{1}{2}\int_{-\infty}^{\infty} e^{-x^2}\mathrm{d}x = \frac{\sqrt{\pi}}{2} \tag{A.3.5}$$

Note that $-xe^{-x^2} = 0$ for $x = \pm\infty$. This proves the result

$$\int_{-\infty}^{\infty} x^2 e^{-x^2}\,\mathrm{d}x = \frac{\sqrt{\pi}}{2} \tag{A.3.6}$$

A.4 Code for P6

A.4.1 Normal Distribution with $\mu = 0$ and $\sigma = 3.0$

```python
# %% Import everything
import numpy as np
from matplotlib import pyplot as plt
from scipy import special as sps

# %% Define functions

# CDF Inverse of Normal Distribution
def normal_cdfinv(yvals, mu, sig_sq):
    """
    Returns the point-wise inverse CDF for 'yvals', all values in the
    range (0, 1). The mean is 'mu' and variance is 'sig_sq'
    """
    x = mu + np.sqrt(2 * sig_sq) * sps.erfinv(2 * yvals - 1)
    return x

# Normal distribution function
def pdf_normal(xvals, mu, sig_sq):
    """
    Evaluates the value of the Normal Probability Density Function at
    the given 'xvals'. The value 'mu' is mean and variance is 'sig_sq'
    """
    cv = 1/np.sqrt(2*np.pi*sig_sq)
    return cv * np.exp(-(xvals-mu)**2 / (2*sig_sq))

# %% Main code
if __name__ == "__main__":
    # Random number generator
    rng = np.random.default_rng(10)
    mu, sigma = 0, 3    # Mean and standard deviation
    N = 10000
    # Generate 10000 samples in U[0, 1]
    yvals = rng.uniform(0, 1, N)
    xvals = normal_cdfinv(yvals, mu, sigma**2)
    # Histogram
    hvals, hedges = np.histogram(xvals, 50, density=True)
    lhedges = hedges[:-1]   # Left edges
    bar_width = 0.85*(lhedges[1] - lhedges[0])
    # Normal distribution (to cross-verify)
    x_n = np.linspace(hedges[0], hedges[-1], 100)
    norm_vals = pdf_normal(x_n, mu, sigma**2)
    # Plot everything
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.bar(lhedges, hvals, align='edge', width=bar_width, \
        label="inv CDF")
    ax.plot(x_n, norm_vals, 'r--', label="Normal Dist")
    ax.set_title("Normal Distribution using inverse CDF")
    ax.legend()
    ax.set_xlabel("x")
    ax.set_ylabel("p(x)")
    fig.savefig("plot_p6_normal.png", dpi=600)
    plt.show()

```

```python
# %%
```

Listing 2: Code to generate Figure 2a

A.4.2 Rayleigh Distribution with $\sigma = 1.0$

```python
# %% Import everything
import numpy as np
from matplotlib import pyplot as plt

# %% Define functions

# CDF Inverse of Rayleigh Distribution
def rayleigh_cdfinv(yvals, sig_sq):
    """
    Returns the point-wise inverse CDF for 'yvals' (whose all values
    must be in the range (0, 1)). CDF is for Rayleigh distribution.
    The 'sig_sq' is squared of sigma (a parameter for the
    distribution).
    """
    return np.sqrt( 2*sig_sq*np.log(1/(1-yvals)) )

# Rayleigh distribution function
def pdf_rayleigh(xvals, sig_sq):
    """
    Evaluates the value of Rayleigh Probability Density Function at
    given 'xvals'. The value for 'sig_sq' is the squared of sigma (a
    parameter for the distribution).
    """
    return (xvals/sig_sq) * np.exp( -(xvals**2)/(2*sig_sq) )

# %% Main code
if __name__ == "__main__":
    # Random number generator
    rng = np.random.default_rng(10)
    sigma = 1.0
    N = 10000
    # Generate 10,000 samples in U[0, 1]
    yvals = rng.uniform(0, 1, N)
    xvals = rayleigh_cdfinv(yvals, sigma**2)
    # Histogram
    hvals, hedges = np.histogram(xvals, 50, density=True)
    lhedges = hedges[:-1]   # Left edges
    bar_width = 0.85*(lhedges[1] - lhedges[0])
    # Rayleigh distribution (to cross-verify)
    x_n = np.linspace(hedges[0], hedges[-1], 100)
    rayleigh_vals = pdf_rayleigh(x_n, sigma**2)
    # Plot everything
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.bar(lhedges, hvals, align='edge', width=bar_width, \
        label="inv CDF")
    ax.plot(x_n, rayleigh_vals, 'r--', label="Rayleigh Dist")
    ax.set_title("Rayleigh Distribution using inverse CDF")
    ax.legend()
    ax.set_xlabel("x")
    ax.set_ylabel("p(x)")
    fig.savefig("plot_p6_rayleigh.png", dpi=600)
    plt.show()

# %%
```

Listing 3: Code to generate Figure 2b

A.4.3 Exponential Distribution with $\lambda = 1.5$

```
# %% Import everything
import numpy as np
from matplotlib import pyplot as plt

# %% Define functions

# CDF Inverse of Exponential Distribution
def exp_cdfinv(yvals, ld):
    """
    Returns the point-wise inverse CDF for 'yvals' (whose all values
    must be in the range (0, 1)). CDF is for Exponential distribution.
    The 'ld' is the lambda value for the exponential distribution.
    """
    return (1/ld)*np.log( 1/(1-yvals) )

# Exponential distribution function
def pdf_exp(xvals, ld):
    """
    Evaluates the value of Exponential Probability Density Function at
    given 'xvals'. The 'ld' is the lambda value for the exponential
    distribution.
    """
    return ld * np.exp(-ld*xvals)

# %% Main code
if __name__ == "__main__":
    # Random number generator
    rng = np.random.default_rng(10)
    lbda = 1.5  # Lambda value for the distribution
    N = 10000
    # Generate 10,000 samples in U[0, 1]
    yvals = rng.uniform(0, 1, N)
    xvals = exp_cdfinv(yvals, lbda)
    # Histogram
    hvals, hedges = np.histogram(xvals, 50, density=True)
    lhedges = hedges[:-1]   # Left edges
    bar_width = 0.85*(lhedges[1] - lhedges[0])
    # Exponential distribution (to cross-verify)
    x_n = np.linspace(hedges[0], hedges[-1], 100)
    exponential_vals = pdf_exp(x_n, lbda)
    # Plot everything
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.bar(lhedges, hvals, align='edge', width=bar_width, \
        label="inv CDF")
    ax.plot(x_n, exponential_vals, 'r--', label="Exp Dist")
    ax.set_title("Exponential Distribution using inverse CDF")
    ax.legend()
    ax.set_xlabel("x")
    ax.set_ylabel("p(x)")
```

```python
    fig.savefig("plot_p6_exp.png", dpi=600)
    plt.show()

# %%
```

Listing 4: Code to generate Figure 2c

A.4.4 Plotting Normal, Rayleigh and Exponential PDFs

```python
# %% Import everything
import numpy as np
from matplotlib import pyplot as plt
# PDFs in earlier codes
from p6_normal import pdf_normal
from p6_rayleigh import pdf_rayleigh
from p6_exp import pdf_exp

# %% Main code
if __name__ == "__main__":
    xlim = [-5, 6]
    n_m, n_sigsq = 0, 1.25 # Mean and Variance of Normal Distribution
    r_sigsq = 1.25 # Sigma squared of Rayleigh Distribution
    e_lambda = 1.5  # Lambda value of Exponential Distribution
    # X values
    xvals = np.linspace(xlim[0], xlim[1], 100)
    # All distribution values
    normal_vals = pdf_normal(xvals, n_m, n_sigsq)
    xvpos = xvals[xvals >= 0]   # Only positive side of X
    rayleigh_vals = pdf_rayleigh(xvpos, r_sigsq)
    exp_vals = pdf_exp(xvpos, e_lambda)
    # Plot names
    normal_name = fr"$N(x;\mu={n_m}, \sigma^2={n_sigsq})$"
    rayleigh_name = fr"$R(x; \sigma^2={r_sigsq})$"
    exponential_name = fr"$E(x; \lambda = {e_lambda})$"
    # Plot everything
    fig = plt.figure()
    ax = fig.add_subplot()
    ax.axvline(0, c='k', ls='-.')
    ax.plot(xvals, normal_vals, '--', label=normal_name)
    ax.plot(xvpos, rayleigh_vals, '--', label=rayleigh_name)
    ax.plot(xvpos, exp_vals, '--', label=exponential_name)
    ax.set_xlim(xlim)
    ax.set_ylim([0, 0.8])
    ax.set_title("Different Probability Density Functions")
    ax.legend()
    ax.grid()
    fig.savefig("plot_p6_all.png", dpi=600)
    plt.show()

# %%
```

Listing 5: Code to generate Figure 2d

A.5 Code for P7

```python
# %% Import everything
import numpy as np
from matplotlib import pyplot as plt

# %% Define functions

# Run the experiment once
def experiment_trial(rng: np.random.Generator, N = 500):
    """
    Runs the experiment: Generates 'N' random numbers from Uniform
    Probability Density Function (low = 0, high = 1), adds them all up
    and returns the resultant sum. The function requires the 'rng'
    which is the random number generator object (numpy.random)
    """
    # Generate N samples
    samples = rng.uniform(0, 1, N)
    return np.sum(samples)  # Return their sum

# Normal distribution function
def pdf_normal(xvals, mu, sig_sq):
    """
    Evaluates the value of the Normal Probability Density Function at
    the given 'xvals'. The value 'mu' is mean and variance is 'sig_sq'
    """
    cv = 1/np.sqrt(2*np.pi*sig_sq)
    return cv * np.exp(-(xvals-mu)**2 / (2*sig_sq))

# %% Main code
if __name__ == "__main__":
    N = 500 # n value for each experiment (number of samples of U)
    N_iter = 50000   # Number of times the experiment must be run
    # Rnadom Number Generator
    rng = np.random.default_rng(10)
    exp_results = []
    for _ in range(N_iter):
        exp_results.append(experiment_trial(rng, N))
    # Convert results to numpy array
    exp_results = np.array(exp_results, dtype=float)
    # Generate normalized histogram from experiment results
    hvals, hedges = np.histogram(exp_results, 50)
    hvals_n = hvals/np.sum(np.diff(hedges)*hvals)   # Normalize hist.
    ledges = hedges[0:-1]
    # Approximating a normal distribution
    x_n = np.linspace(hedges[0], hedges[-1], 100)
    app_norm_vals = pdf_normal(x_n, N/2, N/12)
    # Plot the resultant histogram
    fig = plt.figure()
    ax = fig.add_subplot()
    bin_width = 0.85 * (ledges[1]-ledges[0])
    ax.bar(ledges, hvals_n, width=bin_width, align="edge", \
        label="Irwin-Hall")
    ax.plot(x_n, app_norm_vals, 'r--', label="Normal")
    ax.set_title("Irwin-Hall distribution " + \
        r"$X \rightarrow \sum^{" + str(N) + r"}_{k=1} U(0,1) $")
    ax.set_xlabel(r"$X$")
    ax.set_ylabel(r"Normalized Histogram of $X$ sampled " \
        + str(N_iter) + r" times")
    ax.grid()
    ax.legend()
    fig.savefig("plot_p7.png", dpi=600)
    plt.show()

# %%
```

Listing 6: Code to generate Figure 3