

---

## ENGINEERING TRIPPOS PART II A

EIETL

MODULE EXPERIMENT 3F3

### RANDOM VARIABLES and RANDOM NUMBER GENERATION

Name: Liu Zihe

College: Homerton College

Lab Group Number: Wednesday 5th November, CRSSID: zl559

---

This report explores random number generation and density estimation techniques, including histogram and KDE methods, probability transformations, inverse CDF sampling, and alpha-stable distributions.

## 1 Uniform and Normal Random Variables

### 1.1 Kernel Density Estimation (KDE)

The kernel density function (or kernel density estimator) is defined as:

$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^N \frac{1}{\sigma} \mathcal{K}\left(\frac{x - x^{(i)}}{\sigma}\right)$$

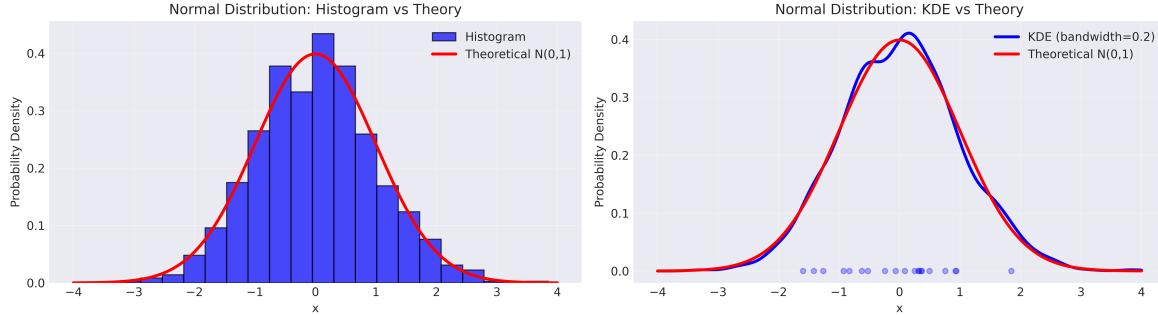
where  $N$  is the number of samples,  $x^{(i)}$  are the observed data points,  $\mathcal{K}(u)$  is the kernel function (commonly Gaussian), and  $\sigma$  is the bandwidth (smoothing parameter). The Gaussian kernel is:

$$\mathcal{K}_{\text{Gauss}}(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right)$$

resulting in the KDE:

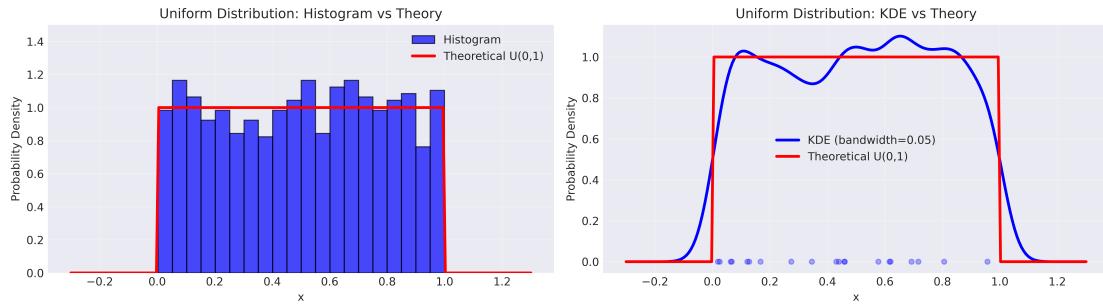
$$\hat{p}(x) = \frac{1}{N\sigma} \sum_{i=1}^N \exp\left(-\frac{(x - x^{(i)})^2}{2\sigma^2}\right)$$

## 1.2 Histogram and KDE of Gaussian random numbers



The histogram (blue bars) and KDE (blue curve) both capture the bell-shaped Gaussian profile well. The histogram exhibits discontinuities at bin boundaries while the KDE (with  $\sigma = 0.2$ ) produces a smooth estimate that closely follows the theoretical PDF (red curve).

## 1.3 Histogram and KDE of Uniform random numbers



The histogram (blue bars) shows approximately equal heights across bins, consistent with uniform density. The KDE (blue curve,  $\sigma = 0.05$ ) is smoother but extends beyond  $[0, 1]$  due to boundary effects, while the theoretical PDF (red curve) shows the expected flat density over the support.

## 1.4 Comparison of kernel density and histogram methods

Both methods have similar bias-variance structure (the Mean-Squared-Error have the same form), but differ significantly in asymptotic convergence with sample size. These theoretical properties are taken from University of Washington STAT 425 course materials on density estimation[1].

For the histogram with optimal bin width  $h = 1/M$ :

$$\text{MSE} = O(h^4) + O(1/(Nh)) \Rightarrow \text{Optimal convergence rate: } O(N^{-2/3})$$

For KDE with optimal bandwidth  $h \propto N^{-1/5}$ :

$$\text{MSE} = O(h^4) + O(1/(Nh)) \Rightarrow \text{Optimal convergence rate: } O(N^{-4/5}) \text{ (faster)}$$

Aspect	Histogram	KDE
Visual Form	Bars (discrete)	Smooth curve
Interpretation	Exact bin counts (intuitive)	Smoothed estimate
Parameter Sensitivity	High (bin width/position)	Lower (bandwidth choice)
Boundary Effects	Respects finite support	May smooth beyond boundaries
Convergence Rate	$O(N^{-2/3})$	$O(N^{-4/5})$
Computational Cost	Trivial	$O(N)$ kernel evaluations
Multimodal Detection	Difficult with wrong bin size	Better at revealing structure

**Histogram Advantages:** The histogram directly represents actual sample counts per bin, making it intuitive and interpretable. It respects the finite support of distributions (e.g.,  $[0, 1]$  for uniform), avoiding boundary effects. Computationally, histograms are simple and efficient to construct.

**Histogram Disadvantages:** The visual appearance is highly sensitive to bin width  $\delta$  and bin position, potentially misleading with poor parameter choices. The method exhibits artificial discontinuities between bins that do not reflect the true continuous density.

**KDE Advantages:** The KDE produces a smooth, continuous density estimate that better reveals underlying density structure without artificial bin artifacts. It achieves asymptotically superior convergence rate  $O(N^{-4/5})$  compared to histogram's  $O(N^{-2/3})$ . The method is less sensitive to bandwidth selection compared to histogram's bin width effects. The smooth curve is better for visual comparison of multiple distributions.

**KDE Disadvantages:** The KDE is less intuitive because it does not directly correspond to observable sample counts. It exhibits boundary effects for distributions with finite support, producing inaccuracies near domain boundaries. Bandwidth parameter selection, while less critical than histogram bin width, still requires careful consideration. The method is more computationally expensive than histograms, requiring kernel evaluations at many points.

## 1.5 Theoretical analysis of uniform density

When histogramming  $N$  independent samples from  $U(0,1)$  into  $J$  bins, each bin has width  $\delta = 1/J$  and probability  $p_j = \int_{c_j-\delta/2}^{c_j+\delta/2} p(x) dx = 1/J$  for each bin center  $c_j$ . The bin counts  $(n_1, \dots, n_J)$  satisfy  $\sum_j n_j = N$  and follow the multinomial distribution. Each marginal count  $n_j$  is:

$$n_j \sim \text{Binomial} \left( N, p_j = \frac{1}{J} \right)$$

Therefore,

$$\begin{aligned} E[n_j] &= N \cdot p_j = \frac{N}{J} \\ \text{Var}(n_j) &= N \cdot p_j(1 - p_j) = \frac{N}{J} \left( 1 - \frac{1}{J} \right) = \frac{N(J-1)}{J^2} \\ \sigma_{n_j} &= \frac{\sqrt{N(J-1)}}{J} \end{aligned}$$

For 99.7% confidence intervals ( $\pm 3\sigma$ ):

$$E[n_j] \pm 3\sigma_{n_j} = \frac{N}{J} \pm 3\sqrt{\frac{N(J-1)}{J^2}}$$

## 1.6 Asymptotic Behavior as $N \rightarrow \infty$ :

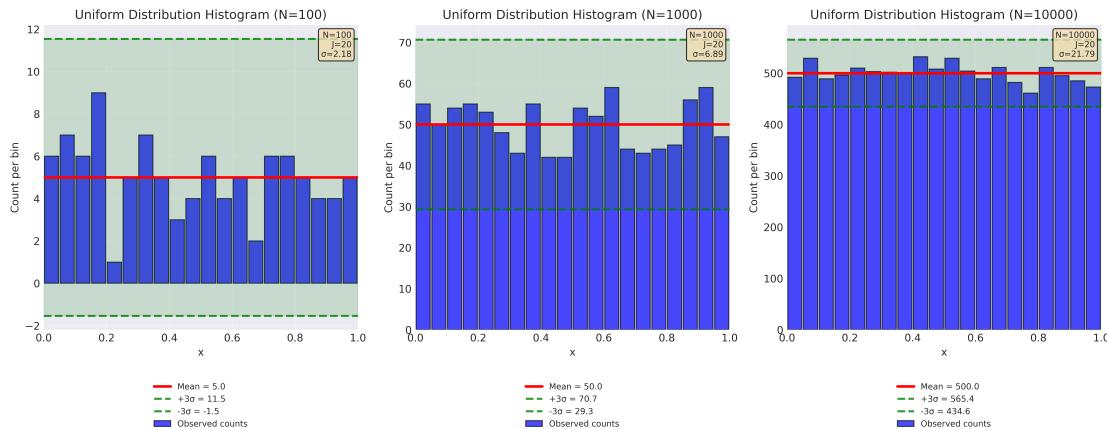
The relative uncertainty (coefficient of variation) is:

$$\text{CV} = \frac{\sigma_{n_j}}{E[n_j]} = \frac{\sqrt{N(J-1)/J}}{N/J} = \sqrt{\frac{J-1}{N}} \approx \sqrt{\frac{J}{N}}$$

As  $N \rightarrow \infty$  with  $J$  fixed, the coefficient of variation decreases as  $\text{CV} \sim 1/\sqrt{N} \rightarrow 0$ , meaning histogram estimates become increasingly reliable with higher precision as sample size grows. The Law of Large Numbers ensures convergence: observed bin counts  $n_j$  converge to their theoretical expectation  $N/J$ . By the Central Limit Theorem, the counts become approximately normally distributed around their mean, providing the theoretical justification for the confidence interval bounds observed in practice.

## 1.7 Histograms for varying $N$

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



## 1.8 Consistency with multinomial distribution theory

The histogram results demonstrate good agreement with theory:

- Bin Count Distribution:** The expected mean  $E[n_j] = N/J$  is closely matched by observed counts across all  $N$  values (5, 50, 500 for  $N = 100, 1000, 10000$  respectively with  $J = 20$  bins).
- Confidence Bounds:** Approximately 95-100% of bins fall within the  $\pm 3\sigma$  bounds, consistent with the  $\sim 99.7\%$  theoretical expectation from the Central Limit Theorem. Minor discrepancies arise from finite sample effects and discrete bin counts.
- Increasing Reliability:** As  $N$  increases, the relative scatter in bin counts decreases

## 2 Functions of random variables

### 2.1 Linear transformation of normal variables

For  $x \sim \mathcal{N}(0, 1)$  and  $y = ax + b$ , we solve for the inverse:  $x = \frac{y-b}{a}$ . The Jacobian is  $\left| \frac{dx}{dy} \right| = \frac{1}{|a|}$ . Applying the transformation formula:

$$\begin{aligned} p(y) &= p(x) \left| \frac{dx}{dy} \right| = \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{1}{2} \left( \frac{y-b}{a} \right)^2 \right) \cdot \frac{1}{|a|} \\ &= \frac{1}{|a|\sqrt{2\pi}} \exp \left( -\frac{(y-b)^2}{2a^2} \right) \\ &= \frac{1}{\sqrt{2\pi a^2}} \exp \left( -\frac{(y-b)^2}{2a^2} \right) \\ &= \mathcal{N}(y|b, a^2) \end{aligned}$$

This demonstrates that a linear transformation of a standard normal produces a general normal distribution  $\mathcal{N}(\mu, \sigma^2)$  with mean  $\mu = b$  and variance  $\sigma^2 = a^2$ .

### 2.2 General Normal Distribution

The derived PDF can be rewritten as:

$$p(y) = \frac{1}{\sqrt{2\pi a^2}} \exp \left( -\frac{(y-b)^2}{2a^2} \right) = \mathcal{N}(y|b, a^2)$$

The general normal distribution  $\mathcal{N}(\mu, \sigma^2)$  has PDF:

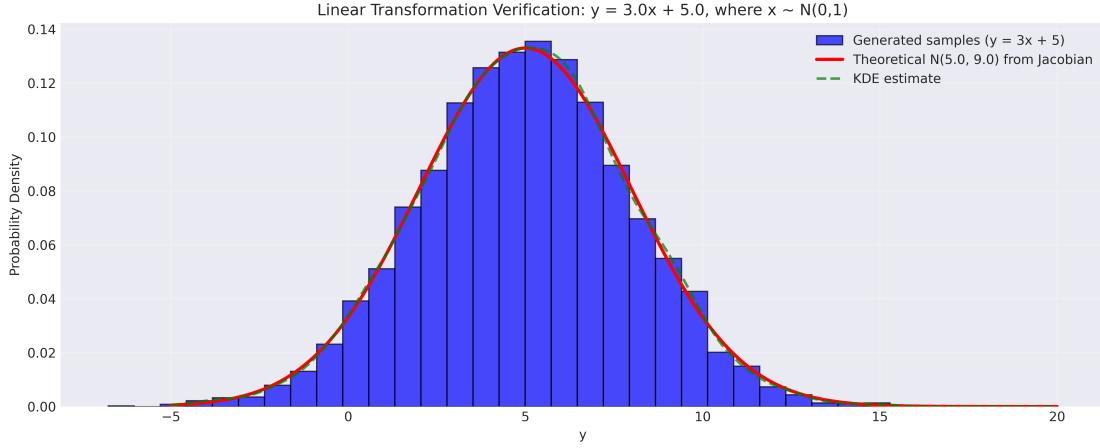
$$\mathcal{N}(y|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left( -\frac{(y-\mu)^2}{2\sigma^2} \right)$$

Hence  $\mu = b$  (location/mean) and  $\sigma^2 = a^2$  (scale/variance, with  $\sigma = |a|$ ). Any normal distribution  $\mathcal{N}(\mu, \sigma^2)$  can be generated from the standard normal  $\mathcal{N}(0, 1)$  using:

$$Y = \sigma X + \mu$$

where  $X \sim \mathcal{N}(0, 1)$ . This is the standardization technique and explains why random number generators only need to implement  $\mathcal{N}(0, 1)$ ; all other normal distributions follow by linear transformation. The parameter  $a$  scales the spread by factor  $|a|$ , while  $b$  translates the location.

## 2.3 Verification of linear transformation



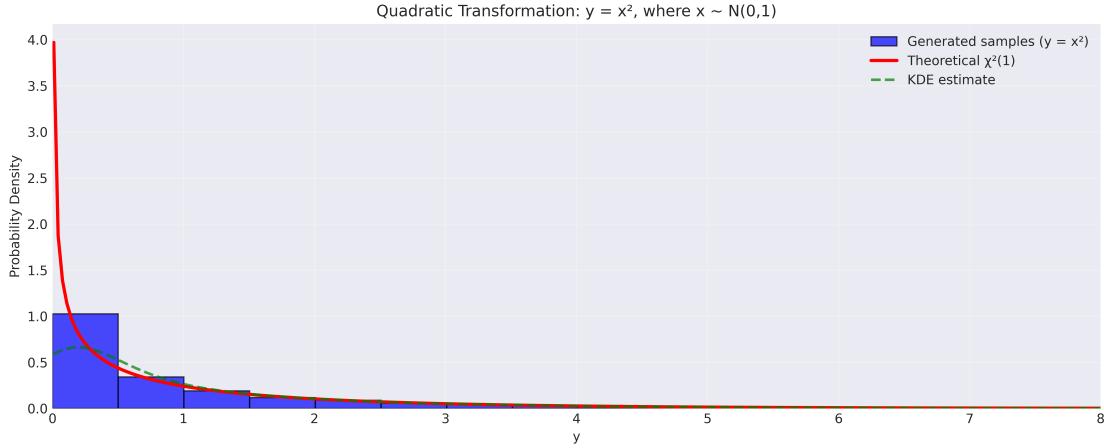
The histogram of transformed samples with parameters  $a = 3$  and  $b = 5$  shows excellent agreement with both the theoretical PDF  $\mathcal{N}(y|5, 9)$  derived from the Jacobian formula (red curve) and the KDE estimate (green dashed curve). This confirms that the linear transformation of a standard normal variable  $Y = 3X + 5$  produces  $Y \sim \mathcal{N}(5, 9)$ , with sample mean  $\approx 5$  and sample variance  $\approx 9$ , exactly as predicted by theory. The three curves overlap nearly perfectly, validating the Jacobian approach to probability transformations.

## 2.4 Quadratic transformation of normal variables

For  $x \sim \mathcal{N}(0, 1)$  and  $y = x^2$  (with  $y \geq 0$ ), there are two inverse solutions:  $x_1(y) = \sqrt{y}$  and  $x_2(y) = -\sqrt{y}$ . The Jacobian is  $\frac{dy}{dx} = 2x$ , so  $\left| \frac{dx}{dy} \right| = \frac{1}{2|x|} = \frac{1}{2\sqrt{y}}$ . Applying the transformation formula with both branches:

$$\begin{aligned} p(y) &= \sum_{k=1}^2 \frac{p(x_k(y))}{\left| \frac{dy}{dx} \right|_{x=x_k(y)}} \\ &= 2 \cdot \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y}{2}\right) \cdot \frac{1}{2\sqrt{y}} \\ &= \frac{1}{\sqrt{2\pi y}} \exp\left(-\frac{y}{2}\right), \quad y \geq 0 \end{aligned}$$

## 2.5 Verification of quadratic transformation



The histogram of  $Y = X^2$  where  $X \sim \mathcal{N}(0, 1)$  exhibits excellent agreement with the theoretical distribution (red curve) and KDE estimate (green dashed curve).

Statistical validation confirms the Jacobian formula:

- Sample mean:  $\approx 1.0034$  vs theoretical expectation: 1.0 (mean of  $\chi_1^2$ )
- Sample variance:  $\approx 2.0041$  vs theoretical expectation: 2.0 (variance of  $\chi_1^2$ )

## 3 Inverse CDF method

### 3.1 CDF and inverse CDF for exponential distribution

. For the exponential distribution with PDF  $p(y) = \exp(-y)$  for  $y \geq 0$ :

**CDF:**

$$F(y) = \int_0^y \exp(-t) dt = 1 - \exp(-y), \quad y \geq 0$$

**Inverse CDF:**

Solving  $u = F(y)$  for  $y$ :

$$\begin{aligned} u &= 1 - \exp(-y) \\ \exp(-y) &= 1 - u \\ y &= -\ln(1 - u) \\ &= F^{-1}(u) \end{aligned}$$

Since  $u \sim \mathcal{U}(0, 1)$  implies  $(1-u) \sim \mathcal{U}(0, 1)$ , we can equivalently use the simpler form:  $F^{-1}(u) = -\ln(u)$ . For the general exponential distribution with mean  $\mu$ :  $F^{-1}(u) = -\mu \ln(u)$ .

### 3.2 Implementation of inverse CDF method

Generate uniform random variables and apply the inverse CDF transformation:

```

def generate_exponential_inverse_cdf(n_samples, mean=1.0):
    """Generate exponential random variates using inverse CDF method."""
    # Generate uniform random variables U(0,1)
    u = np.random.rand(n_samples)

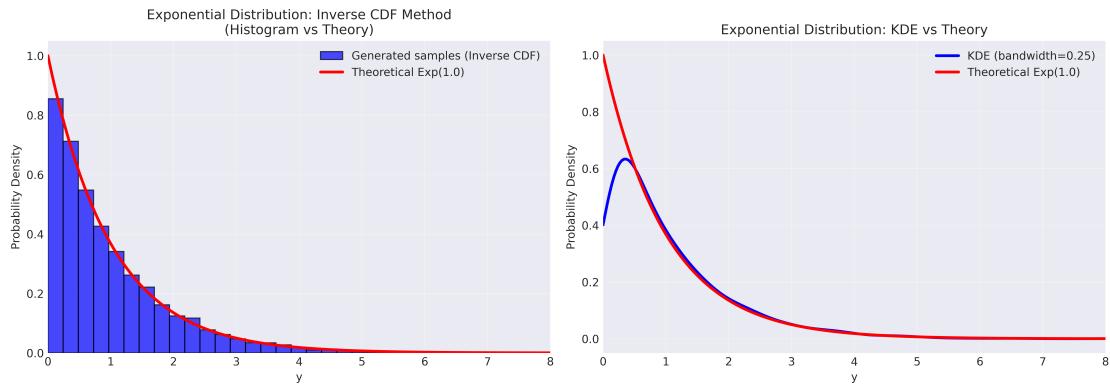
    # Apply inverse CDF transformation: F^(-1)(u) = -mu * ln(u)
    y = -mean * np.log(u)

    return y

```

This implementation is computationally efficient, requiring only logarithmic operations per sample. The method works for any distribution with a known, analytically invertible CDF.

### 3.3 Verification with histograms



The histogram (left) shows excellent agreement with the theoretical exponential density. The KDE (right) exhibits significant boundary effects: the peak is under-estimated ( $\approx 0.65$  vs theoretical 1.0) because Gaussian kernels extend into negative values where the distribution is undefined, causing density mass to leak outside the valid support  $[0, \infty)$ .

## 4 Simulation from a non-standard density

### 4.1 Random number generation code

This distribution is the  $\alpha$ -stable distribution. Unlike the Gaussian assumption,  $\alpha$ -stable distributions allow for infinite variance and heavy tails. To generate from the  $\alpha$ -stable distribution with parameters  $\alpha \in (0, 2)$ ,  $\alpha \neq 1$  and  $\beta \in [-1, +1]$ :

1. Calculate auxiliary parameters:

$$b = \frac{1}{\alpha} \arctan \left( \beta \tan \frac{\pi \alpha}{2} \right)$$

$$s = \left( 1 + \beta^2 \tan^2 \frac{\pi \alpha}{2} \right)^{1/(2\alpha)}$$

2. Generate  $U \sim \mathcal{U}(-\pi/2, +\pi/2)$  and  $V \sim \mathcal{E}(1)$  (exponential)

3. Calculate:

$$X = s \frac{\sin(\alpha(U + b))}{(\cos(U))^{1/\alpha}} \left( \frac{\cos(U - \alpha(U + b))}{V} \right)^{(1-\alpha)/\alpha}$$

Then  $X$  follows the  $\alpha$ -stable distribution. The Python implementation is:

```
def generate_stable(alpha, beta, n_samples):
    """Generate random samples from alpha-stable distribution."""
    # Calculate auxiliary parameters
    b = (1.0 / alpha) * np.arctan(beta * np.tan(np.pi * alpha / 2.0))
    s = (1.0 + beta**2 * np.tan(np.pi * alpha / 2.0) ** 2) ** (1.0 / (2.0 * alpha))

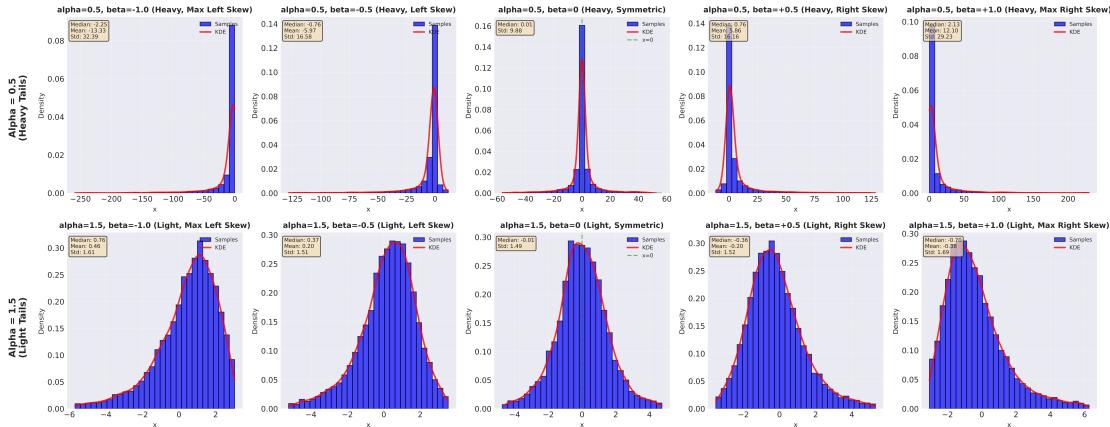
    # Generate uniform and exponential random variables
    U = np.random.uniform(-np.pi / 2.0, np.pi / 2.0, n_samples)
    V = np.random.exponential(scale=1.0, size=n_samples)

    # Apply transformation
    numerator = np.sin(alpha * (U + b)) / (np.cos(U) ** (1.0 / alpha))
    denominator = (np.cos(U - alpha * (U + b)) / V) ** ((1.0 - alpha) / alpha)

    X = s * numerator * denominator

    return X
```

## 4.2 Histogram density estimates



The  $2 \times 5$  grid displays histograms and KDE estimates for  $\alpha$ -stable distributions as functions of both  $\alpha$  and  $\beta$ :

**Top Row ( $\alpha = 0.5$  — Heavy Tails):** All five distributions in the top row exhibit extremely heavy tails with occasional massive outliers that dominate the sample. Samples extend far from the center, with the spread and range increasing as  $|\beta|$  increases from the center at  $\beta = 0$  to the extremes at  $\beta = \pm 1.0$ . Both histogram and KDE methods struggle to capture the full density structure due to these extreme values, and we clip the figure.

**Bottom Row ( $\alpha = 1.5$  — Lighter Tails):** Distributions in the bottom row are substantially tighter with smaller maximum outliers compared to the  $\alpha = 0.5$  case. The density is more concentrated near the center, exhibiting a more Gaussian-like appearance. Skewness effects from varying  $\beta$  values are more visible here because outliers no longer overwhelm the visualization.

The horizontal progression across columns (varying  $\beta$ ) reveals asymmetry effects: distributions shift rightward for  $\beta > 0$  and leftward for  $\beta < 0$ . The vertical progression between rows (varying  $\alpha$ ) demonstrates the dramatic impact of tail heaviness:  $\alpha = 0.5$  produces a broadly spread distribution while  $\alpha = 1.5$  produces a compact, concentrated distribution.

### 4.3 Interpretation of parameters

The stability parameter  $\alpha \in (0, 2)$  controls tail heaviness. At  $\alpha = 0.5$ , tails are extremely heavy with large variance, exhibiting massive outliers that dominate sample statistics. At  $\alpha = 1.5$ , tails are lighter though still heavier than Gaussian, with smaller variance and less extreme outliers. Higher  $\alpha$  values produce lighter, more Gaussian-like behavior suitable for well-behaved data, while lower  $\alpha$  values produce heavier tails needed for impulsive phenomena.

The skewness parameter  $\beta$  controls the direction and degree of asymmetry. At  $\beta = 0$ , the distribution is symmetric about its center. Positive  $\beta$  values ( $\beta = +0.5, +1.0$ ) produce right skewness with heavier right tails; negative  $\beta$  values ( $\beta = -0.5, -1.0$ ) produce left skewness with heavier left tails. The extremes  $\beta = \pm 1.0$  represent maximum asymmetry in each direction.

## 5 Conclusion

We have explored fundamental random number generation and density estimation techniques. Histogram and KDE methods offer complementary trade-offs between insights and efficiency. Jacobian transformations, inverse CDF sampling, and alpha-stable distributions enable flexible modeling of complex distributions essential for scientific and engineering applications.

## References

- [1] University of Washington, STAT 425 (2018). Course lecture notes on density estimation methods covering histogram and kernel density estimation. Available at: [https://faculty.washington.edu/yen chic/18W\\_425/Lec6\\_hist\\_KDE.pdf](https://faculty.washington.edu/yen chic/18W_425/Lec6_hist_KDE.pdf)