

# Simulating the bins: Analysis of Histogram Layout Differences Across Binwidths

Each distribution in the document is visualized using faceted histograms with different bin sizes (0.5, 1, 2, 5, 10). The differences between the histograms provide insights into how binwidth affects the appearance of data distributions. Below, I describe the key differences in layout across bin sizes for each distribution.

- **Discrete Distributions (Poisson, Binomial, Negative Binomial):** Large bins can obscure their count-based nature.
- **Continuous Distributions (Normal, Gamma, Log-Normal, Beta, etc.):** Larger bins help in smoothing but might remove finer shape details.
- **Heavy-Tailed Distributions (Cauchy, Log-Normal):** Large binwidths underrepresent extreme values.
- **Symmetric vs. Skewed Distributions:** Symmetric ones (Normal, Uniform) maintain their shape better across binwidths, while skewed ones (Exponential, Chi-Square) lose more information.

## General Insights Across All Distributions

### *Small Binwidths (0.5, 1)*

- Provide more detail but can make histograms look noisy.
- More suited for fine-grained analysis of distributions.
- Best for discrete distributions where individual values matter.

### *Larger Binwidths (2, 5, 10)*

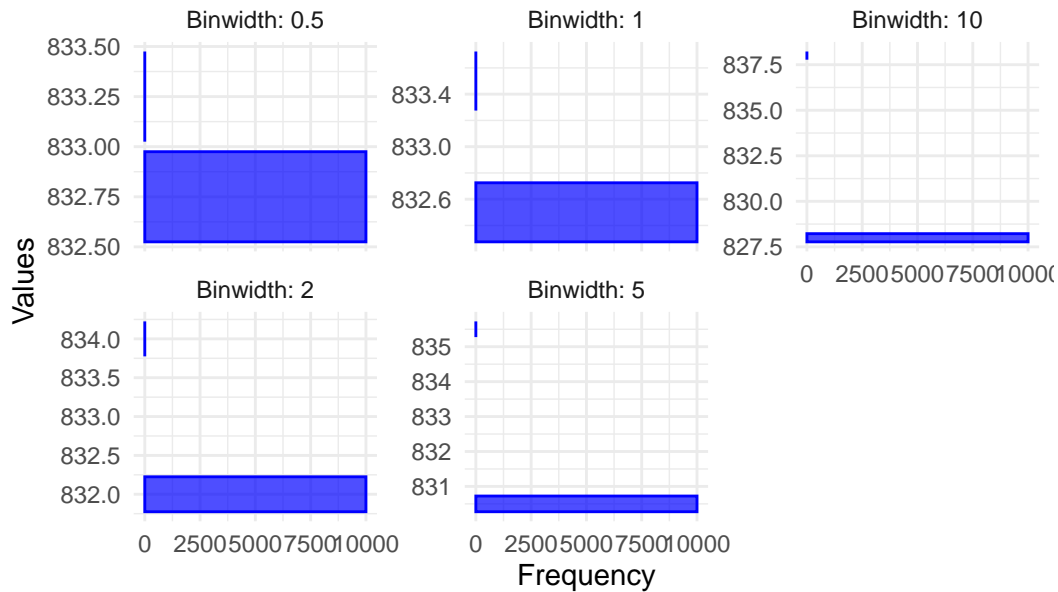
- Smooth out the data, reducing visible fluctuations.
- Best for understanding overall trends rather than fine details.
- Risk of oversimplifying discrete distributions by merging distinct values.

## Point Mass Distribution:

- A single value is repeated (5 in this case) for all data points.
- Multiple histograms are shown, faceted by binwidth (0.5, 1, 2, 5, 10), illustrating how bin size affects the visualization.

```
simulate_point_mass <- function(n, value) {  
  rep(value, n)  
}  
  
# Simulate data  
set.seed(42) # Set seed for reproducibility  
n <- 10000  
  
# Define binwidths for faceting  
binwidths <- c(0.5, 1, 2, 5, 10)  
  
point_mass_data <- simulate_point_mass(n, round(n/12, 0))  
  
plot_histogram_facet(point_mass_data, binwidths, "Point Mass Distribution")
```

## Point Mass Distribution



## Exponential Distribution:

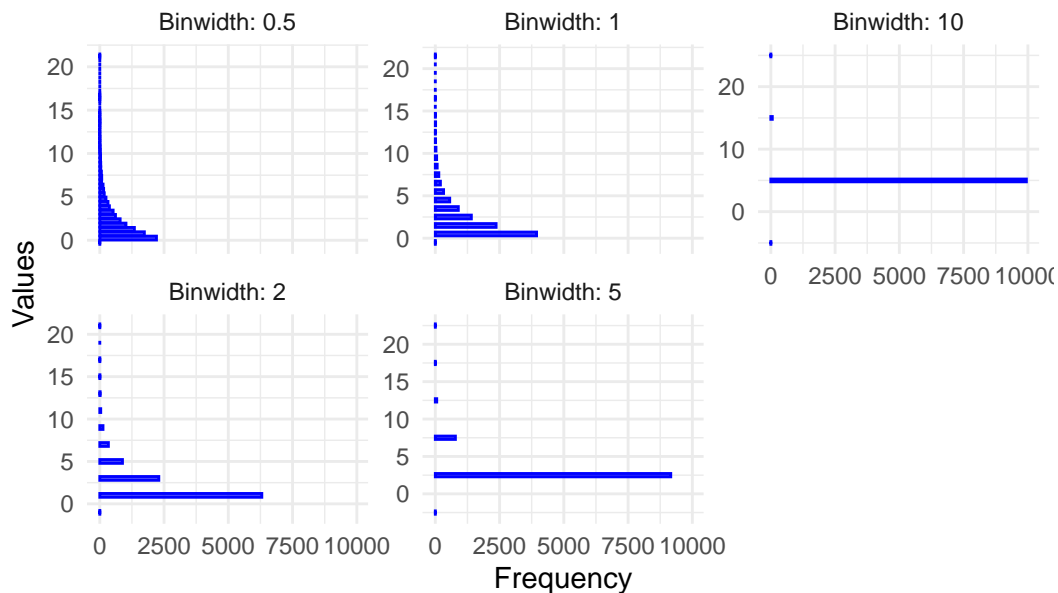
- Simulated data follows an exponential decay pattern.
- Histograms for different binwidths are included, highlighting the rapid decay near lower values and the impact of bin size.

```
# Function to simulate exponential distribution
simulate_exponential <- function(n, rate) {
  rexp(n, rate = rate)
}

exponential_data <- simulate_exponential(n, rate = 0.5)

plot_histogram_facet(exponential_data, binwidths, "Exponential Distribution")
```

## Exponential Distribution



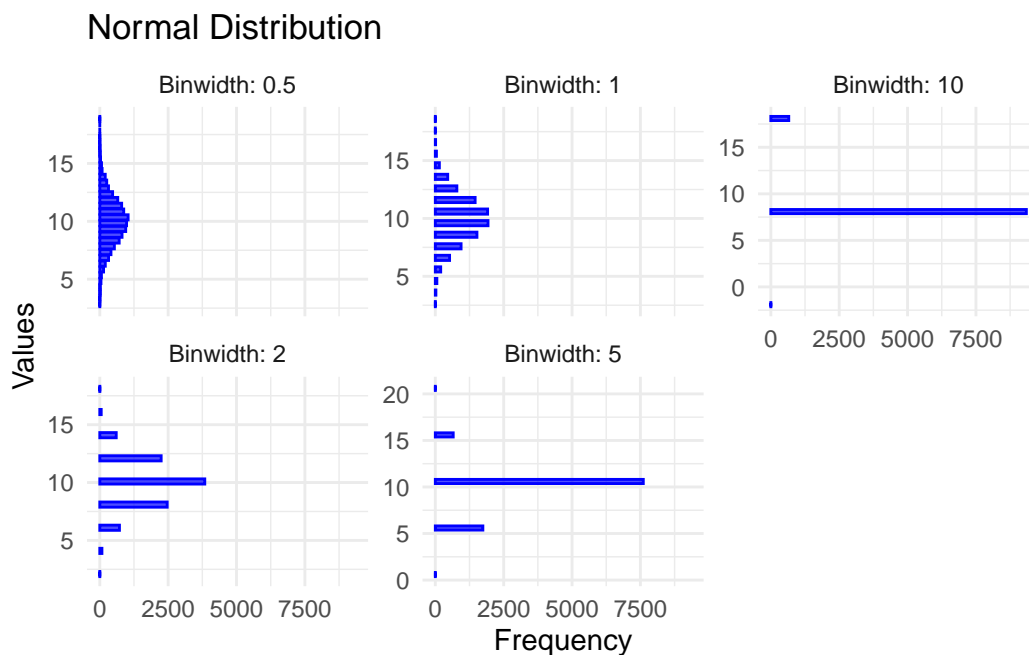
## Normal Distribution:

- Simulated data is centered around a mean (10) with a specified standard deviation (2).
- Histograms are displayed for various binwidths, showing the characteristic bell-shaped curve of the normal distribution.

```
# Function to simulate normal distribution
simulate_normal <- function(n, mean, sd) {
  rnorm(n, mean = mean, sd = sd)
}

normal_data <- simulate_normal(n, mean = 10, sd = 2)

plot_histogram_facet(normal_data, binwidths, "Normal Distribution")
```



## Poisson Distribution:

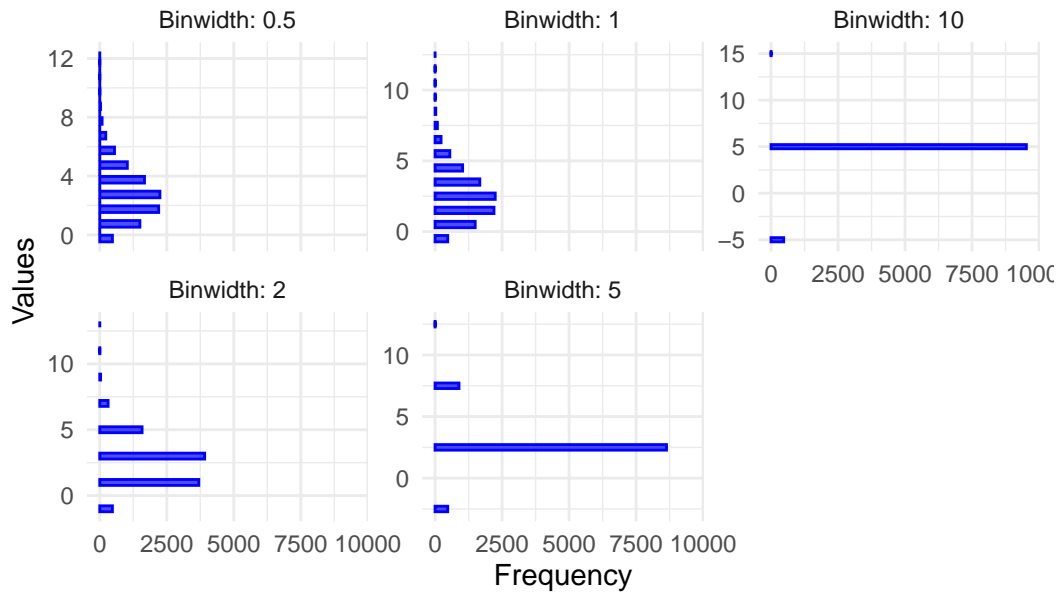
- Simulated count data with a mean occurrence ( $\lambda = 3$ ).
- Histograms display discrete peaks, varying by binwidth, illustrating the distribution of counts.

```
# Function to simulate Poisson distribution
simulate_poisson <- function(n, lambda) {
  rpois(n, lambda = lambda)
}

poisson_data <- simulate_poisson(n, lambda = 3)

plot_histogram_facet(poisson_data, binwidths, "Poisson Distribution")
```

## Poisson Distribution



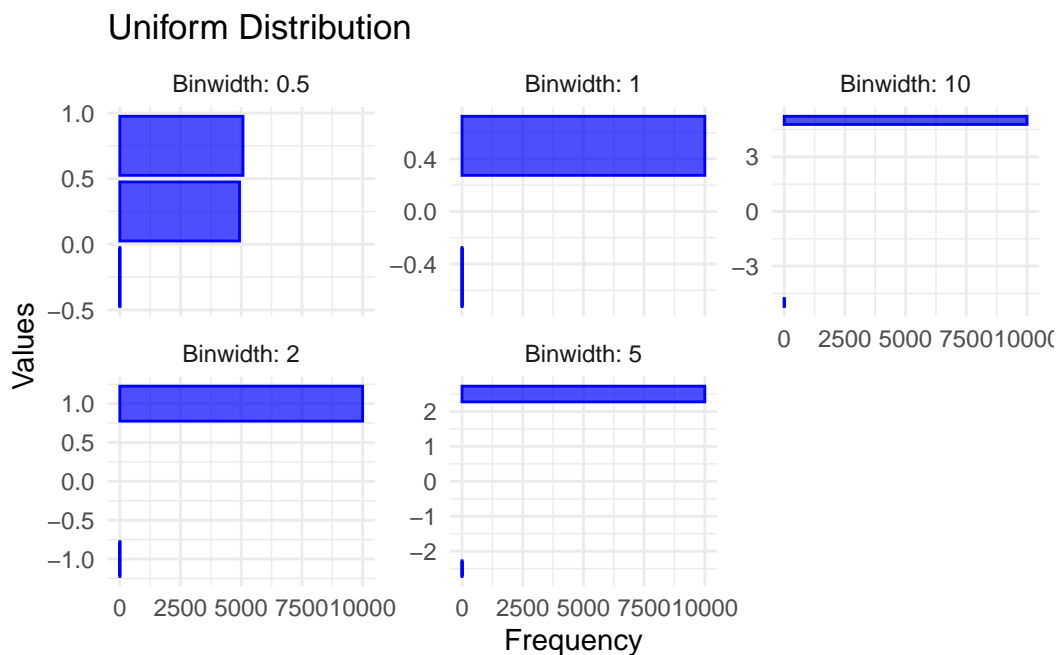
## Uniform Distribution:

- Simulated data is evenly distributed between a minimum (0) and maximum (1).
- Histograms reflect the flat, even distribution across the range for different binwidths.

```
# Function to simulate Uniform distribution
simulate_uniform <- function(n, min, max) {
  runif(n, min = min, max = max)
}

uniform_data <- simulate_uniform(n, min = 0, max = 1)

plot_histogram_facet(uniform_data, binwidths, "Uniform Distribution")
```



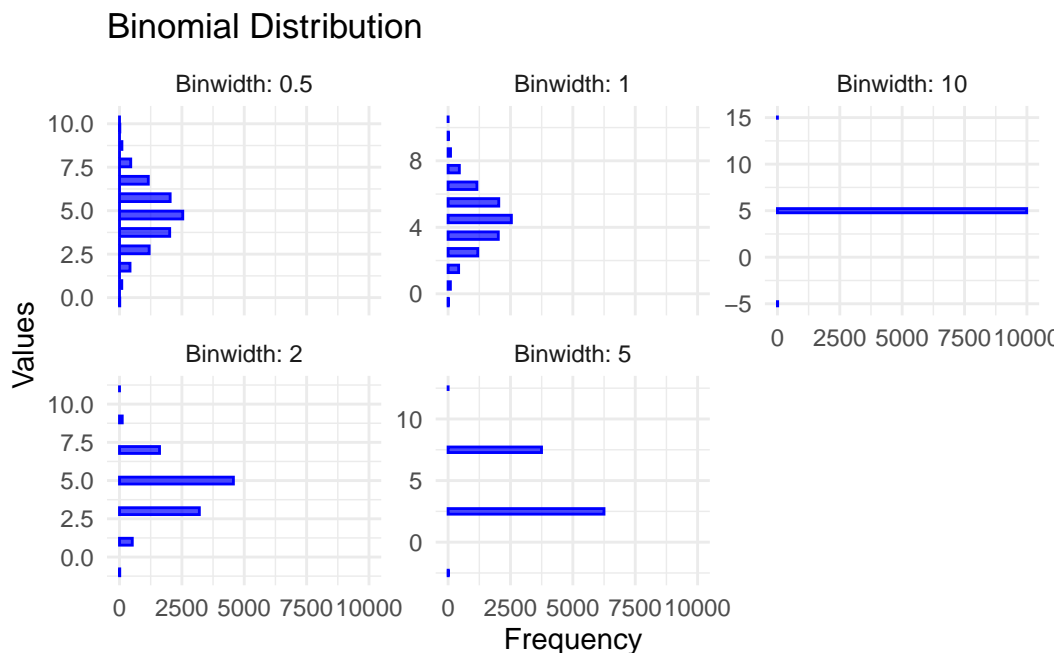
## Binomial Distribution:

- Models the number of successes in a fixed number of trials (size = 10, prob = 0.5).
- Histograms show the discrete nature of the data and its dependency on binwidth.

```
# Function to simulate binomial distribution
simulate_binomial <- function(n, size, prob) {
  rbinom(n, size = size, prob = prob)
}

binomial_data <- simulate_binomial(n, size = 10, prob = 0.5)

plot_histogram_facet(binomial_data, binwidths, "Binomial Distribution")
```



## Beta Distribution:

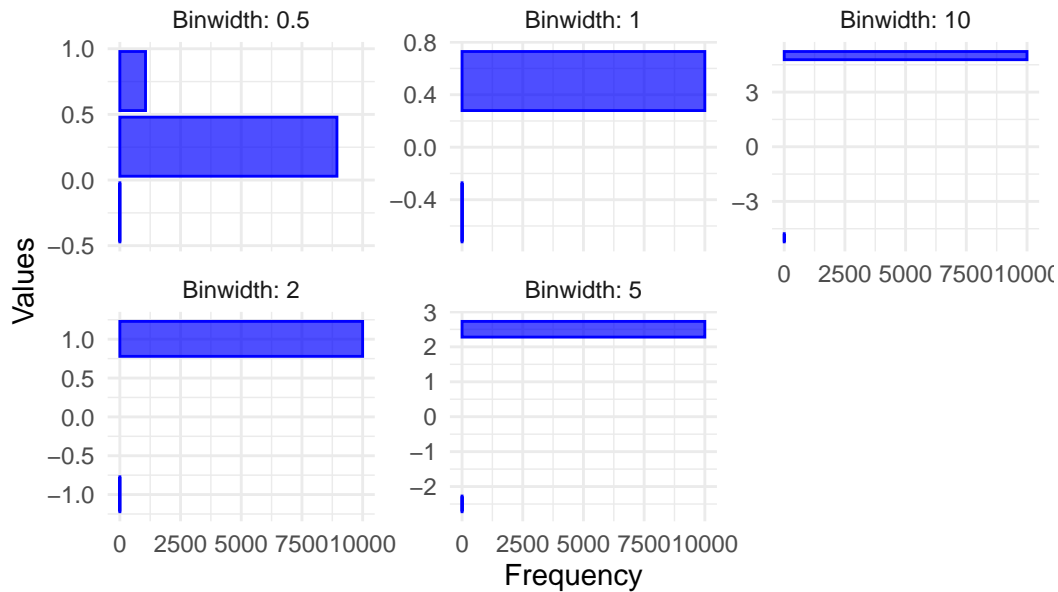
- Continuous data bounded between 0 and 1, modeled with shape parameters (shape1 = 2, shape2 = 5).
- Histograms for different binwidths reveal the skewed shape of the beta distribution.

```
# Function to simulate beta distribution
simulate_beta <- function(n, shape1, shape2) {
  rbeta(n, shape1 = shape1, shape2 = shape2)
}

beta_data <- simulate_beta(n, shape1 = 2, shape2 = 5)

plot_histogram_facet(beta_data, binwidths, "Beta Distribution")
```

## Beta Distribution



## Gamma Distribution:

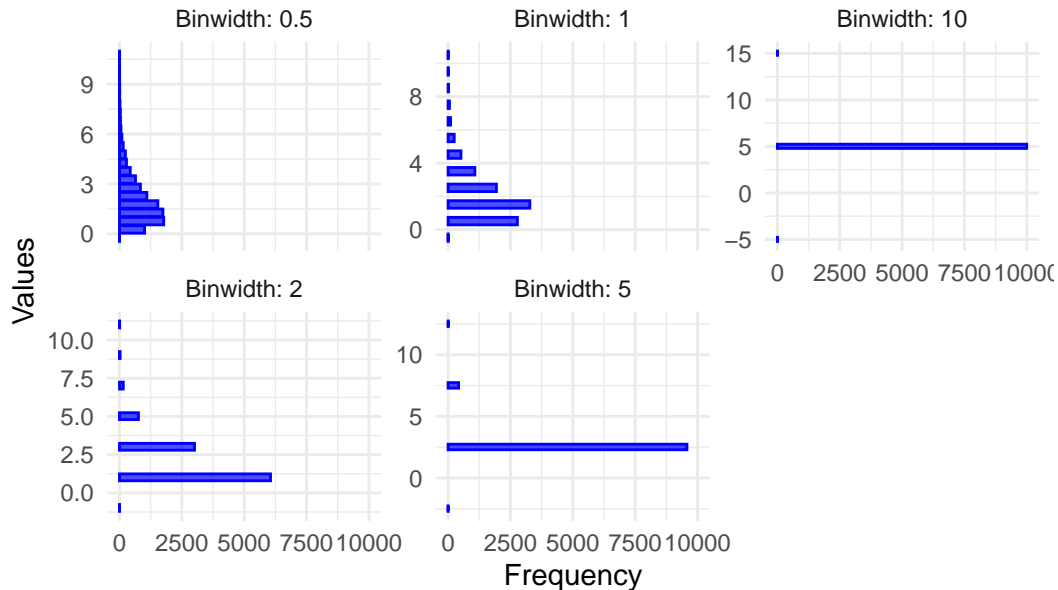
- Models waiting times with shape and rate parameters (shape = 2, rate = 1).
- Histograms demonstrate the positively skewed nature of the distribution.

```
# Function to simulate gamma distribution
simulate_gamma <- function(n, shape, rate) {
  rgamma(n, shape = shape, rate = rate)
}

gamma_data <- simulate_gamma(n, shape = 2, rate = 1)

plot_histogram_facet(gamma_data, binwidths, "Gamma Distribution")
```

## Gamma Distribution



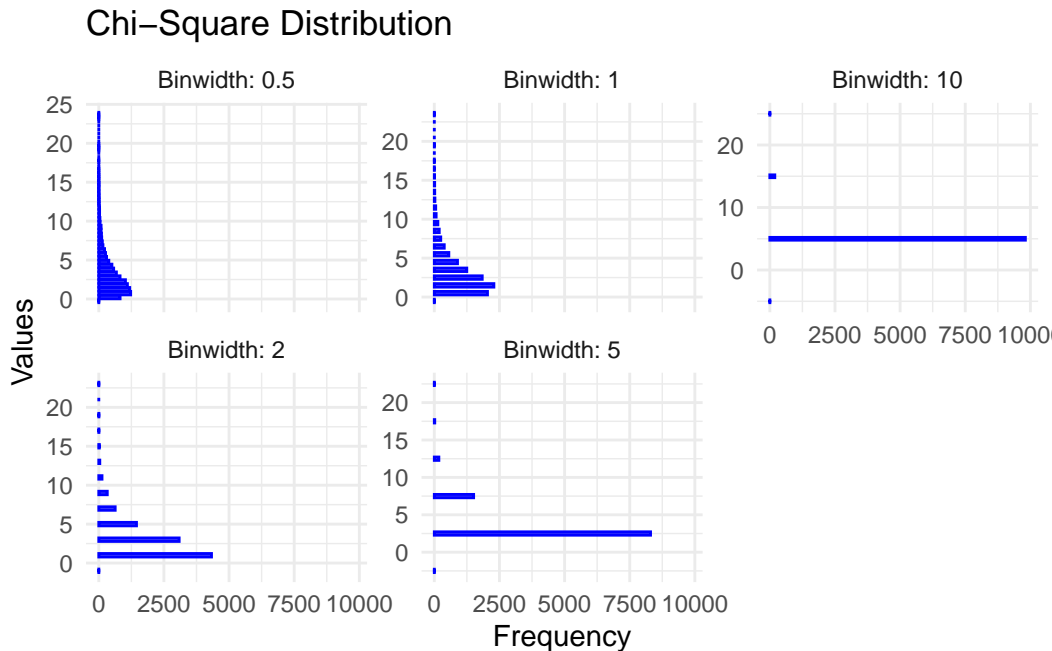
### Chi-Square Distribution:

- Data follows a chi-square distribution with degrees of freedom ( $df = 3$ ).
- Histograms illustrate the skewed nature of the chi-square distribution.

```
# Function to simulate chi-square distribution
simulate_chisquare <- function(n, df) {
  rchisq(n, df = df)
}

chisquare_data <- simulate_chisquare(n, df = 3)

plot_histogram_facet(chisquare_data, binwidths, "Chi-Square Distribution")
```



### Negative Binomial Distribution:

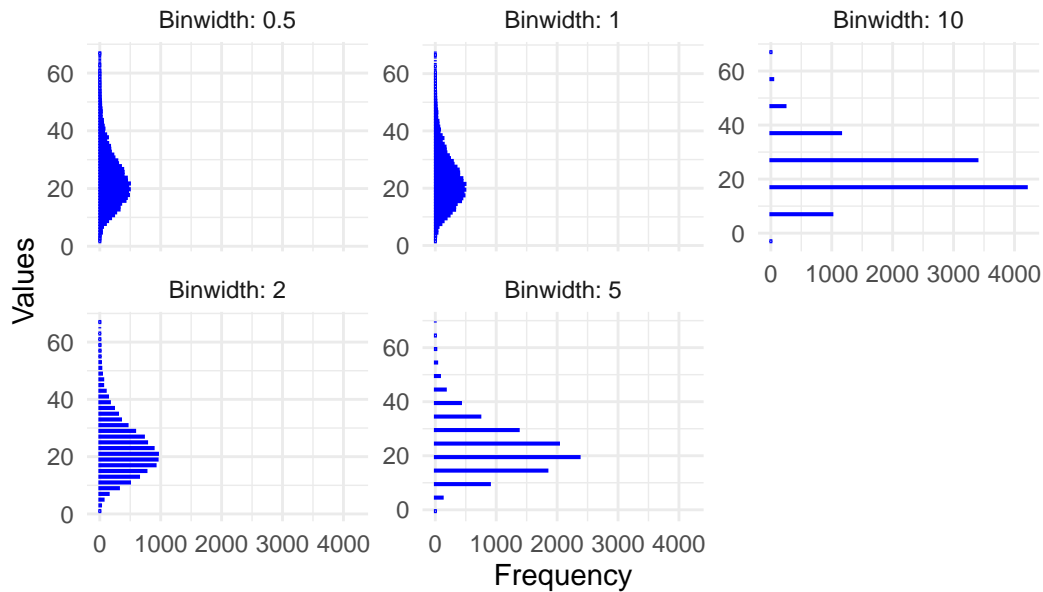
- Models the number of failures before achieving a set number of successes.
- Histograms display the count data and its dependence on binwidth.

```
# Function to simulate negative binomial distribution
simulate_negative_binomial <- function(n, size, prob) {
  rnbinom(n, size = size, prob = prob)
}

negative_binomial_data <- simulate_negative_binomial(n, size = 10, prob = 0.3)

plot_histogram_facet(negative_binomial_data, binwidths, "Negative Binomial Distribution")
```

## Negative Binomial Distribution



## Log-Normal Distribution:

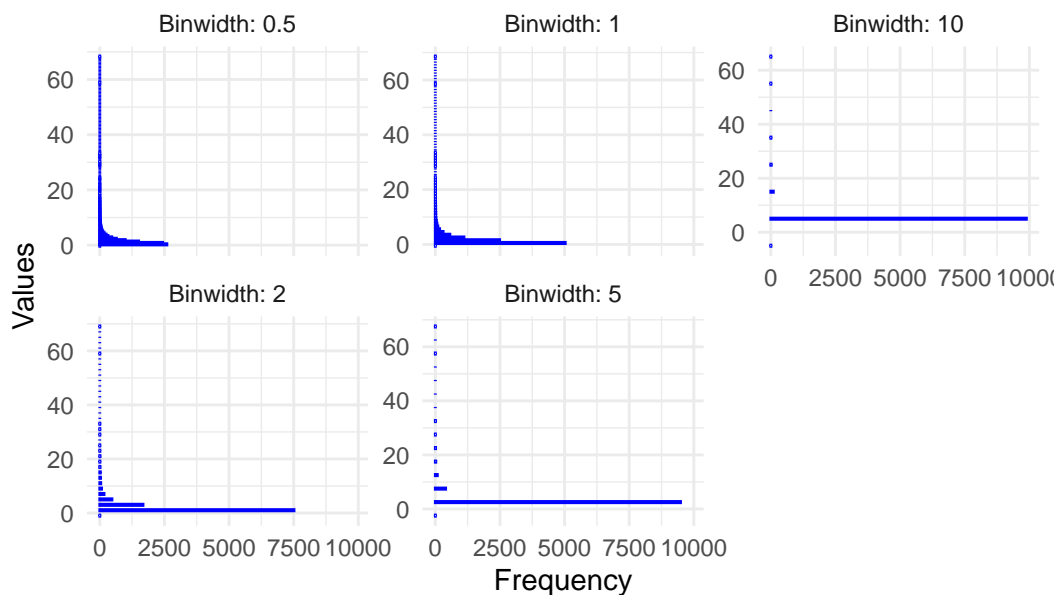
- Data is distributed such that the logarithm of the values follows a normal distribution ( $\text{meanlog} = 0$ ,  $\text{sdlog} = 1$ ).
- Histograms reveal the heavy tail characteristic of log-normal data.

```
# Function to simulate log-normal distribution
simulate_lognormal <- function(n, meanlog, sdlog) {
  rlnorm(n, meanlog = meanlog, sdlog = sdlog)
}

lognormal_data <- simulate_lognormal(n, meanlog = 0, sdlog = 1)

plot_histogram_facet(lognormal_data, binwidths, "Log-normal Distribution")
```

## Log-normal Distribution





## Cauchy Distribution:

- A heavy-tailed distribution (location = 0, scale = 1).
- Histograms demonstrate extreme variability, with very large positive and negative values for different binwidths.

```
# Function to simulate Cauchy distribution
simulate_cauchy <- function(n, location, scale) {
  rcauchy(n, location = location, scale = scale)
}

cauchy_data <- simulate_cauchy(n, location = 0, scale = 1)

plot_histogram_facet(cauchy_data, binwidths, "Cauchy Distribution")
```

