$$f(x;\lambda) = \lambda e^{-\lambda x}, \quad \text{for } x \geq 0$$

$$f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad \text{for } x \in (-\infty, \infty)$$

$$P(X = k; n, p) = \binom{n}{k} p^k (1-p)^{n-k}, \quad \text{for } k \in \{0, 1, \ldots, n\}$$

$$E[X] = \int_{-\infty}^{\infty} x \cdot f(x)\, dx$$

$$E[X] = \sum_x x \cdot P(x)$$

Exponential, normal, binomial, for continuous, for discrete

To go from ~ bin() to ~(app) N():
mu = n*p
sig = sqrt(n*p*(1-p))

Useful range for “double” plots (plot on a plot)
x=seq(min(vector), max(vector), length=100)

3plik, 12, 13, 15, 16

lambda = 1/mean for exp

N(mu, sig)
bin(n, p)
exp(lambda)

📝 IMPORTANT R & Statistics Review

1. Vector and Matrix Subsetting

Concept	R Code	Note
Exclude element(s)	`a[-4]` or `a[-c(4, 5)]`	Use a **negative** index vector.
Filtering (Logical)	`d = a[a > 4]`	Selects elements where the condition is `TRUE`.
All Indices of Min	`which(a == min(a))`	Use `which()` on the logical vector `a == min(a)` to get all indices where the minimum occurs.
Trace of a Matrix	`sum(diag(A))`	Sum of the diagonal elements.
Matrix Inverse	`A_inverse = solve(A)`	Use `solve(A)`. Verification: `round(A %*% A_inverse)` gives the Identity Matrix.
Vector Multiplication	`t(z1) %*% z2` (Scalar) or `z1 %*% t(z2)` (Matrix)	Use `%*%` for true matrix multiplication.

2. Distribution Distinctions (Continuous vs. Discrete)

Feature	Discrete (e.g., Binomial)	Continuous (e.g., Normal, Exponential)
CDF Notation	$P(X \leq k) = F(k)$	$P(X \leq x) = F(x)$
Equality Check	**Matters!** $P(X \leq k) \neq P(X < k)$	**Does NOT Matter!** $P(X \leq x) = P(X < x)$
`pbinom` **vs** `pexp`	`pbinom(k-1, n, p)` for $P(X < k)$. You **subtract 1** for strict inequality.	`pexp(x, lambda)` for $P(X < x)$. No need to subtract 1.

3. Normal Approximation & Central Limit Theorem (CLT)

Variable	Distribution	Parameters	Formula to Calculate σ
Binomial X	$X \sim \text{Bin}(n, p)$	$\mu = np$	$\sigma = \sqrt{np(1-p)}$
Approximation	$X \sim \text{N}(\mu, \sigma)$	$\mu = np$	$\sigma = \sqrt{np(1-p)}$
Sample Mean $\bar{X}$	$\bar{X} \sim \text{N}(\mu, \frac{\sigma}{\sqrt{n}})$	$\mu = E[X]$	$\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{n}}$

Eksportuj do Arkuszy

- **CLT Applicability:** You can use the Normal approximation for $\bar{X}$ if the sample size is **large ($n \geq 30$)**, even if the original population distribution is unknown.

4. Confidence Intervals and Sample Size

Parameter Estimated	Condition	R Quantile Function	Interval for Parameter θ
Mean (μ)	σ **Unknown**	`qt(1-alpha/2, n-1)`	$\bar{x} \pm t_{\alpha/2,n-1}\frac{s}{\sqrt{n}}$
Mean (μ)	σ **Known**	`qnorm(1-alpha/2)`	$\bar{x} \pm z_{\alpha/2}\frac{\sigma}{\sqrt{n}}$
Variance (σ^2)	**Always**	`qchisq(1-alpha/2, n-1)` and `qchisq(alpha/2, n-1)`	$\left[\frac{(n-1)s^2}{\chi^2_{\alpha/2}}, \frac{(n-1)s^2}{\chi^2_{1-\alpha/2}}\right]$

Eksportuj do Arkuszy

- **Sample Size (n) for Mean:**

$$n = \left(\frac{z_{\alpha/2} \cdot \sigma}{\text{Error}}\right)^2$$

Where Error *is the max acceptable deviation from* $\bar{x}$*. Always use* `ceiling(n)`.

- **Sample Size (n) for Proportion (p):**

$$n = \frac{z^2_{\alpha/2} \cdot \hat{p}(1-\hat{p})}{\text{Error}^2}$$

If $\hat{p}$ *is unknown, assume* $\hat{p} = 0.5$ *for maximum required sample size.*

5. Data Visualization and Handling

- `discrete.histogram` (from `arm`): Best for plotting **discrete data** (like Binomial counts).
- `pie()` function **requires** a frequency table input: `pie(table(data), main=title)`.
- `na.omit(data.frame)`**: Removes entire rows** that contain *any* `NA` value. Use this when you need to keep all columns **balanced** for comparison (e.g., for `boxplot(na.omit(straws))`).
- **Histogram Breaks (** `hist()` **):** For continuous data, using `br = seq(min(data), max(data), length=floor(sqrt(length(data))))` is a good way to estimate bin size.
- **Quantile Functions (** `q` **functions):** They are the **inverse of the** `p` **functions** (CDF). They return the x-value corresponding to a given cumulative probability $P(X \le x) = p$.
  - `qnorm(0.9)` gives the Z-score for the 90th percentile.
  - `qt(0.9, 24)` gives the t-value for the 90th percentile with $df = 24$.