Sampling Distribution and Confidence

What is a Sampling Distribution?

A sampling distribution is the probability distribution of a given statistic based on a random sample. It shows how the statistic (like the mean, proportion, variance, etc.) varies from sample to sample.This is crucial for inferential statistics because it allows us to make conclusions about the population from which the sample was drawn.

Sampling Distribution of the Mean

The sampling distribution of the sample mean is the distribution of sample means from multiple samples of the same size taken from a population. It describes how the sample mean varies from sample to sample and is centered around the population mean.

With Known Standard Deviation, If the population standard deviation (sigma) is known, the sampling distribution of the sample mean (x_bar) will have:

1. **Mean**: (mean(x_bar) =mu).
2. **Standard Deviation (Standard Error)**: sigma(x_bar)= sigma/sqrt(n).

Without Known Standard Deviation, If (sigma) is unknown, we use the sample standard deviation ((s)) instead:

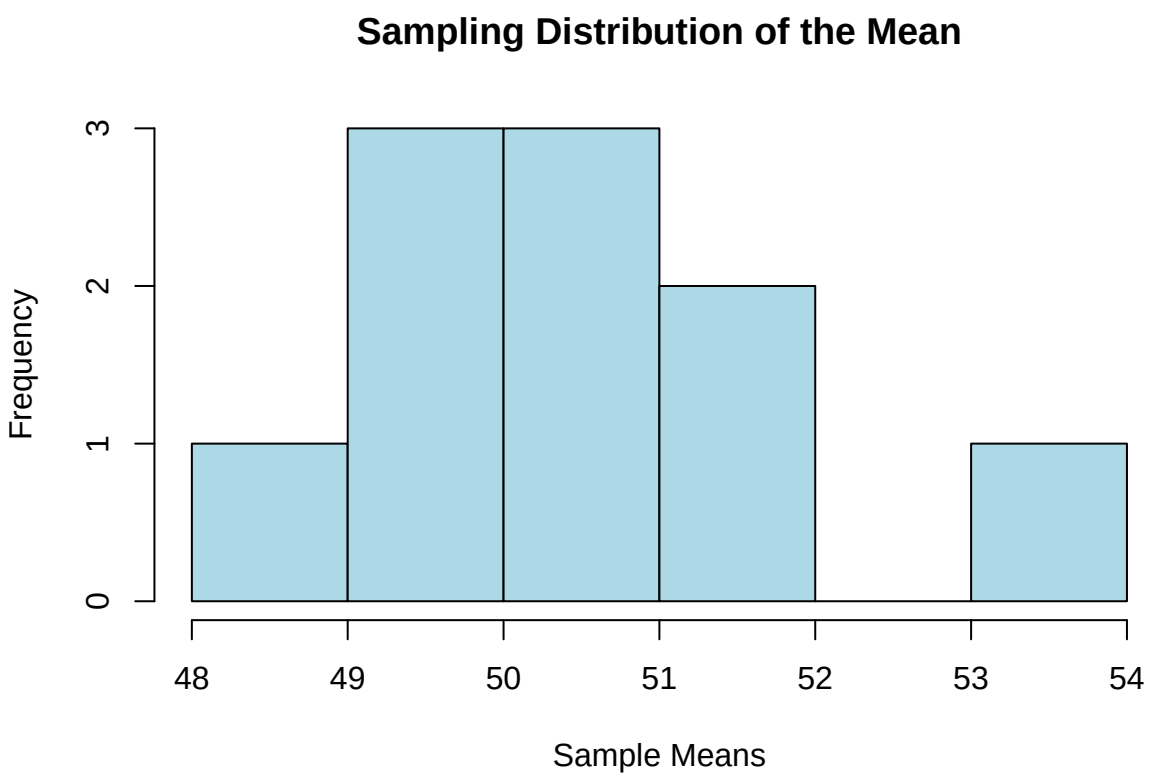
1. **Mean**: mean(x_bar) = mu.
2. **Standard Deviation (Standard Error)**: s_(x_bar) = s/sqrt(n)

```
# Generate a sampling distribution of the mean
set.seed(123)
n <- 30  # sample size
mu <- 50  # population mean
sigma <- 10  # population standard deviation

# Generate 10 sample means
sample_means <- replicate(10, mean(rnorm(n, mean = mu, sd = sigma)))
sample_means
```

```
##  [1] 49.52896 51.78338 50.24420 49.06111 48.16420 51.53717 50.14706 49.10350
##  [9] 50.50765 53.36690
```

```
# Plot the sampling distribution
hist(sample_means, main = "Sampling Distribution of the Mean",
     xlab = "Sample Means", col = "lightblue")
```



Sampling Distribution of the Proportion

With Known Standard Deviation, for a population proportion (p), the sampling distribution of the sample proportion (p_hat) has:

1. **Mean**: (mean(p_hat) = p)
2. **Standard Deviation**: (sigma(p_hat) = sqrt((p(1-p))/n)

Without Known Standard Deviation, If (p) is unknown, we estimate it using the sample proportion (p_hat):

1. **Mean**: (mean(p_hat) = p_hat)
2. **Standard Deviation**: (sigma(p_hat) = sqrt((p_hat(1-p_hat))/n)

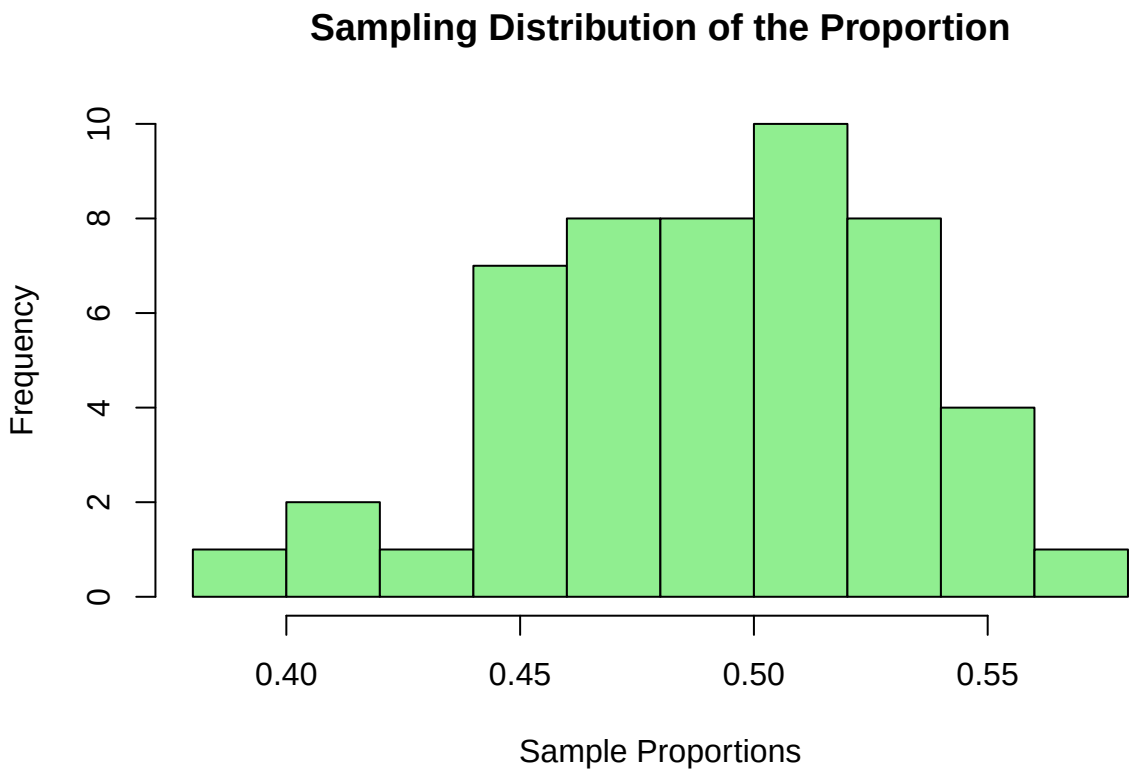
```r
# Generate a sampling distribution of the proportion
set.seed(123)
n <- 100  # sample size
p <- 0.5  # population proportion

# Generate 50 sample proportions
sample_props <- replicate(50, mean(rbinom(n, 1, p)))
sample_props
```

```
##  [1] 0.47 0.50 0.48 0.44 0.46 0.55 0.51 0.48 0.52 0.52 0.47 0.51 0.54 0.41 0.56
```

```
## [16] 0.41 0.53 0.55 0.54 0.45 0.50 0.51 0.47 0.49 0.45 0.53 0.46 0.51 0.53 0.49
## [31] 0.51 0.51 0.49 0.56 0.46 0.54 0.50 0.48 0.48 0.53 0.53 0.46 0.45 0.52 0.49
## [46] 0.47 0.52 0.57 0.39 0.49
```

```r
# Plot the sampling distribution
hist(sample_props, main = "Sampling Distribution of the Proportion"
     , xlab = "Sample Proportions", col = "lightgreen")
```



Sampling Distribution of the Variance and Standard Deviation

The sampling distribution of the sample variance (s^2) and standard deviation (s) can be more complex. Generally, the sample variance follows a chi-square.

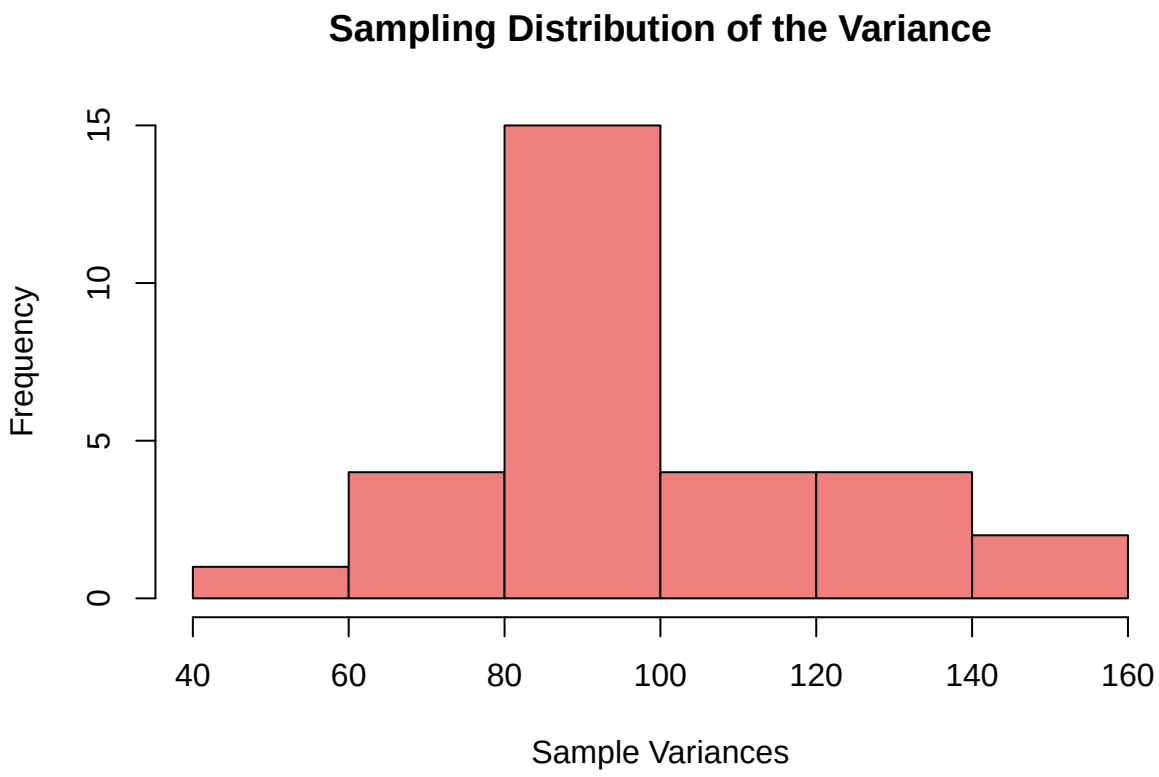
```r
#1.Generate a sampling distribution of the variance
set.seed(123)
n <- 30  # sample size
mu <- 50  # population mean
sigma <- 10  # population standard deviation

# Generate 30 sample variances
sample_vars <- replicate(30, var(rnorm(n, mean = mu, sd = sigma)))
sample_vars
```

```
##  [1]  96.24212  69.74386  75.65412  82.24174 131.90037  88.66058  93.59418
```

```
##  [8]  76.28923 105.90066  79.98859 107.57523 135.82855  81.58651  85.94886
## [15] 128.24190  99.38662  87.02215  57.23671  91.05741 111.43561 121.43143
## [22]  86.60307  99.17597  94.06372 147.94530  91.33887  83.61134 100.88210
## [29] 144.02004  89.93023
```

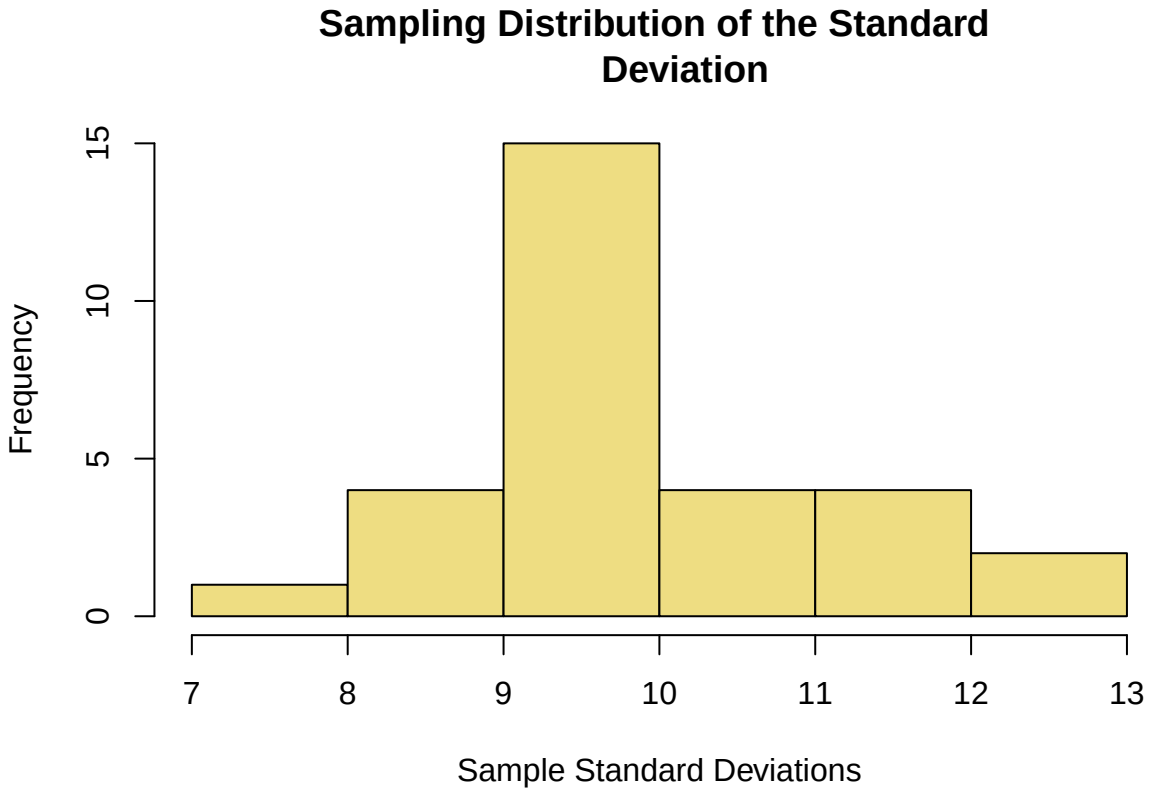
```
# Plot the sampling distribution
hist(sample_vars, main = "Sampling Distribution of the Variance",
     xlab = "Sample Variances", col = "lightcoral")
```




```
#2.Generate 30 sample standard deviations
sample_sds <- sqrt(sample_vars)
sample_sds
```

```
##  [1]  9.810307  8.351279  8.697938  9.068723 11.484789  9.415975  9.674408
##  [8]  8.734371 10.290805  8.943634 10.371848 11.654550  9.032525  9.270861
## [15] 11.324394  9.969284  9.328566  7.565495  9.542401 10.556307 11.019593
## [22]  9.306077  9.958713  9.698645 12.163277  9.557137  9.143923 10.044008
## [29] 12.000835  9.483155
```

```
# Plot the sampling distribution
hist(sample_sds, main = "Sampling Distribution of the Standard
     Deviation", xlab = "Sample Standard Deviations",
     col = "lightgoldenrod")
```



Confidence Intervals

A confidence interval is a range of values, derived from sample data, that is likely to contain the value of an unknown population parameter. The interval has an associated confidence level that quantifies the level of confidence that the parameter lies within the interval. For example, a 95% confidence interval suggests that if you were to take 100 different samples and compute a confidence interval for each sample, then approximately 95 of the 100 confidence intervals will contain the population parameter.

Confidence Intervals for the Mean

When the Population Standard Deviation is Known,If the population standard deviation (sigma) is known, the confidence interval for the mean (mean) is calculated using the **Z-distribution**:

```
X-bar ± Z_alpha/2 * (sigma/sqrt(n))
```

where:

1. (x_bar) is the sample mean.
2. (Z_alpha/2) is the Z-value that corresponds to the desired confidence level.
3. (sigma) is the population standard deviation.
4. (n) is the sample size.

```r
ci_mean_known_sigma <- function(sample, sigma, confidence = 0.95) {
  n <- length(sample)
  mean <- mean(sample)
  z <- qnorm(1 - (1 - confidence) / 2)
  margin_of_error <- z * (sigma / sqrt(n))
  return(c(mean - margin_of_error, mean + margin_of_error))
}
```

```r
sample <- c(10, 12, 14, 16, 18)
sigma <- 2
ci <- ci_mean_known_sigma(sample, sigma)
print(paste("Confidence Interval for Mean (known sigma):", ci))
```

```
## [1] "Confidence Interval for Mean (known sigma): 12.2469549188468"
## [2] "Confidence Interval for Mean (known sigma): 15.7530450811532"
```

When the Population Standard Deviation is Unknown, If the population standard deviation is unknown, the confidence interval for the mean is calculated using the **t-distribution**:

```
X_bar ± t_alpha/2(df) * (s/sqrt(n))
```

where:

1. (x_bar) is the sample mean.
2. (t_alpha/2,df) is t-value corresponds to desired confidence level,df of n-1.
3. (s) is the sample standard deviation.
4. (n) is the sample size.

```r
ci_mean_unknown_sigma <- function(sample, confidence = 0.95) {
  n <- length(sample)
  mean <- mean(sample)
  s <- sd(sample)
  t <- qt(1 - (1 - confidence) / 2, df = n - 1)
  margin_of_error <- t * (s / sqrt(n))
  return(c(mean - margin_of_error, mean + margin_of_error))
}
```

```r
ci <- ci_mean_unknown_sigma(sample)
print(paste("Confidence Interval for Mean (unknown sigma):", ci))
```

```
## [1] "Confidence Interval for Mean (unknown sigma): 10.0735136770449"
## [2] "Confidence Interval for Mean (unknown sigma): 17.9264863229551"
```

Confidence Intervals for Proportions

When the Population Proportion is Known,If the population proportion ((p)) is known, the confidence interval for the proportion is:

```
p_hat ± Z_alpha/2 * sqrt((p(1-p))/n)
```

where:

1. (hat_p) is the sample proportion
2. (Z_alpha/2) is the Z-value that corresponds to the desired confidence level.
3. (n) is the sample size.

```r
ci_proportion_known_p <- function(sample_size, p, confidence = 0.95) {
  z <- qnorm(1 - (1 - confidence) / 2)
  margin_of_error <- z * sqrt(p * (1 - p) / sample_size)
  return(c(p - margin_of_error, p + margin_of_error))
}
```

```r
sample_size <- 100
p <- 0.5
ci <- ci_proportion_known_p(sample_size, p)
print(paste("Confidence Interval for Proportion (known p):", ci))
```

```
## [1] "Confidence Interval for Proportion (known p): 0.402001800772997"
## [2] "Confidence Interval for Proportion (known p): 0.597998199227003"
```

When the Population Proportion is Unknown,than the confidence interval for the proportion is:

```
p_hat ± Z_alpha/2 * sqrt((p_hat(1-p_hat))/n)
```

where:

1. (hat_p) is the sample proportion.
2. (Z_alpha/2}) is the Z-value that corresponds to the desired confidence level.
3. (n) is the sample size.

```r
ci_proportion_unknown_p <- function(sample, confidence = 0.95) {
  n <- length(sample)
  p_hat <- mean(sample)
  z <- qnorm(1 - (1 - confidence) / 2)
  margin_of_error <- z * sqrt(p_hat * (1 - p_hat) / n)
  return(c(p_hat - margin_of_error, p_hat + margin_of_error))
}
```

```r
sample <- c(1, 0, 1, 1, 0, 1, 0, 1, 1, 0)
ci <- ci_proportion_unknown_p(sample)
print(paste("Confidence Interval for Proportion (unknown p):", ci))
```

```
## [1] "Confidence Interval for Proportion (unknown p): 0.296363685148402"
## [2] "Confidence Interval for Proportion (unknown p): 0.903636314851598"
```

Confidence Intervals for Variance and Standard Deviation

Confidence Interval for Variance,The confidence interval for the population variance (sigmaˆ2) is calculated using the **chi-square distribution**:

((n-1)s^2 * chi_alpha/2(df_2), (n-1)s^2 * chi_1-alpha/2(df_2)

where:

1. (s^2) is the sample variance
2. chi^2_alpha/2(df) and chi^2_{1-alpha/2(df) are the chi-square values for desired confidence level and df = n - 1.

```r
ci_variance <- function(sample, confidence = 0.95) {
  n <- length(sample)
  s2 <- var(sample)
  chi2_lower <- qchisq((1 - confidence) / 2, df = n - 1)
  chi2_upper <- qchisq(1 - (1 - confidence) / 2, df = n - 1)
  lower <- (n - 1) * s2 / chi2_upper
  upper <- (n - 1) * s2 / chi2_lower
  return(c(lower, upper))
}
```

```r
ci <- ci_variance(sample)
print(paste("Confidence Interval for Variance:", ci))
```

```
## [1] "Confidence Interval for Variance: 0.126164605771584"
## [2] "Confidence Interval for Variance: 0.88876067693844"
```

Confidence Interval for Standard Deviation

The confidence interval for the population standard deviation (sigma) is derived from the variance confidence interval:

(sqrt((n-1)s^2 * chi_alpha/2(df_2)), sqrt((n-1)s^2 * chi_1-alpha/2(df_2)))

Where:

1. (s) is the sample standard deviation
2. chi^2_alpha/2(df) and chi^2_{1-alpha/2(df) are the chi-square values for desired confidence level and df = n - 1.

```r
ci_std_dev <- function(sample, confidence = 0.95) {
  var_ci <- ci_variance(sample, confidence)
  return(sqrt(var_ci))
}
```

```r
ci <- ci_std_dev(sample)
print(paste("Confidence Interval for Standard Deviation:", ci))
```

```
## [1] "Confidence Interval for Standard Deviation: 0.355196573423201"
## [2] "Confidence Interval for Standard Deviation: 0.942741044475332"
```