Statistics Guide

Descriptive vs Inferential Statistics

- Statistics – the science of collecting and analyzing data
- Inferences - drawing conclusions from samples that apply to a broader population
- Looking at a subset of a population – **sample** – representation of the population
- Important to define how we **extract** the sample (defining the. inclusion and exclusion criteria) and to understand the population by looking at trends
- Summarization of data obtained from the sample – **descriptive statistics**
- Infer it back to the broader population - **inferential statistics**

Getting R Studio Environment Ready

Download R

- Go to the Comprehensive R Archive Network (CRAN) website (https://cran.r-project.org/ (https://cran.r-project.org/)) and select the link corresponding to your operating system (Linux, macOS, or Windows).
- For Windows and macOS, click on the link to download the latest release. Linux users will find instructions to add a repository and install R using the package manager specific to their Linux distribution.

Install R

- **Windows:** Run the downloaded .exe file and follow the installation instructions.
- **macOS:** Open the downloaded .pkg file and follow the installation instructions.
- **Linux:** Follow the command-line instructions provided on the CRAN website for your specific Linux distribution.

Installing RStudio

1. **Download RStudio** Visit the RStudio download page (https://www.rstudio.com/products/rstudio/download (https://www.rstudio.com/products/rstudio/download/#download)) and download the RStudio Desktop version that matches your operating system.
2. **Install RStudio**

- **Windows:** Run the downloaded .exe file and follow the installation instructions.
- **macOS:** Open the downloaded .dmg file, and drag the RStudio icon to your Applications folder.
- **Linux:** Depending on the package format you downloaded, you may need to use a package manager or execute a command in the terminal to install it.

Benefits of RStudio

- RStudio is an integrated development environment (IDE) for R. It offers several benefits that enhance the R programming experience, including:
- **Syntax Highlighting and Code Completion:** Makes it easier to read and write code. Console and Script Editor: Allows you to write scripts and execute commands directly. Workspace Management: Easily manage your workspace variables, including viewing data tables and summary statistics.
- **Plotting and Visualization:** View plots and graphical output directly within the IDE, and manage graphical history.
- **Package Management:** Simplified installation and management of R packages.
- **Project Management:** Keep your R projects organized, making it easier to manage files, data, and scripts associated with specific projects.
- **Version Control Integration:** Supports integration with Git and SVN for version control.
- **Extensibility:** Enhance RStudio's capabilities with various add-ins and extensions.
- **Support for R Markdown:** Facilitates dynamic reporting and reproducible research with integration for R Markdown documents, allowing you to compile reports that include code, output, and narrative text.

The Law of Large Numbers

- The **Law of Large Numbers** states that with a large enough sample size, the sample mean will be close to the p opulation mean, meaning a large number of independent and random samples converged to the true value if there exists one.

- **Example:** Consider flipping a fair coin. The probability of getting heads (or tails) is 0.5. If you flip the coin only a few times, you might not get exactly 50% heads and 50% tails due to randomness. However, as you flip the coin more and more times (say, thousands or millions of flips), the proportion of heads to total flips will get closer and closer to 0.5, illustrating the Law of Large Numbers.

```
# Load the necessary library for plotting
library(ggplot2)

# Function to simulate die rolls and calculate averages
simulate_die_rolls <- function(max_rolls) {
  rolls = numeric(max_rolls) # Vector to store the outcomes of the die rolls
  averages = numeric(max_rolls) # Vector to store the running averages

  for (i in 1:max_rolls) {
    roll = sample(1:6, 1) # Simulate rolling the die
    rolls[i] = roll
    averages[i] = mean(rolls[1:i]) # Calculate the average of all rolls up to the current roll
  }

  return(averages)
}

# Simulate rolling a die for 1000 times
max_rolls = 1000
averages = simulate_die_rolls(max_rolls)

# Plotting the result
# The results should show how the average outcome hovers around 3.5, roughly resembling the properties of a norma
l distribution
data = data.frame(Roll = 1:max_rolls, Average = averages)
ggplot(data, aes(x = Roll, y = Average)) +
  geom_line() +
  geom_hline(yintercept = 3.5, linetype="dashed", color = "red") +
  labs(title = "Demonstration of the Law of Large Numbers with Die Rolls",
       x = "Number of Rolls",
       y = "Average Outcome") + theme_minimal()
```

The Assumption(s) of Normality

- To provide a specific probability for observing a particular event or a particular difference between two events, our statistical procedures must make some assumptions.
- One of these assumptions is that the sampling distribution of the mean is **normal**.
- That is, if you took a sample, calculated its mean, and wrote this down; then took another (independent) sample (from the same population) and got its mean and wrote it down; and did this an infinite number of times; then the distribution of the values that you wrote down would always be a perfect bell curve. [1]

[1] https://www2.psychology.uiowa.edu/faculty/mordkoff/GradStats/part 1/l.07 normal.pdf (https://www2.psychology.uiowa.edu/faculty/mordkoff/GradStats/part%201/l.07%20normal.pdf)

Sample Size and Study Design

- The design of any study is more important than analyzing its results, as a poorly designed study can never be recovered, whereas a poorly analyzed study can be reanalyzed to reach a meaningful conclusion.[2]
- Rather, the design of the study decides how the data generated can be best analyzed.
- The sample size is another important factor in a well-designed study, the more samples we acquire the more statistical power we gain thereby capturing the characteristics of a population.

[2] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2924977/#:~:text=The design of any study,generated can be best analyzed (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2924977/#:~:text=The%20design%20of%20any%20study,generated%20can%20be%20best%20analyzed).

Summarizing Data & Presenting Data in Tables & Graphs

The Measure of Central Tendency (M - estimators)

Measures of central tendency are statistical metrics that describe the center or typical value of a dataset. They are crucial for summarizing a single value representing a dataset's middle. The three main measures of central tendency are:

- **Mean:** The arithmetic average of a set of numbers, calculated by summing all the numbers and then dividing by the count of those numbers.
- **Median:** The middle value in a set of numbers, which divides the dataset into two halves. If the dataset has an even number of observations, the median is the average of the two middle numbers.
- **Mode:** The most frequently occurring value(s) in a dataset. A dataset may have one mode (unimodal), more than one mode (bimodal or multimodal), or no mode at all.

- **Geometric mean** – The geometric mean is a measure of central tendency that is especially useful for sets of numbers whose values are meant to be multiplied together or are of different orders of magnitude (eg. used for calculating growth rates and ratios)
- **Harmonic mean** – The harmonic mean is the reciprocal of the arithmetic mean of the reciprocals of a set of numbers. It is particularly useful for averages of rates or ratios, such as speed or density.
- **Weighted mean** – The weighted mean, also known as the weighted average, is a mean where each data point contributes to the average with a weight that reflects its importance.

Some points to consider

- Extreme data points do not influence the median
- Extreme data points can heavily influence the mean
- Harmonic mean <= Geometric mean <= mean

```
# Load the iris dataset and required packages
data(iris)
library(e1071)

# Calculate Mean
mean_sepal_length <- mean(iris$Sepal.Length)
mean_sepal_length

# Calculate Median
median_sepal_length <- median(iris$Sepal.Length)
median_sepal_length

# Calculate Mode
# In R, there's no built-in function for mode calculation similar to mean() or median()
get_mode <- function(v) {
  uniq_v <- unique(v)
  uniq_v[which.max(tabulate(match(v, uniq_v)))]
}

mode_sepal_length <- get_mode(iris$Sepal.Length)
mode_sepal_length

# Calculate Geometric Mean
# Since R doesn't have a built-in function for geometric mean, we write one.
geometric_mean <- function(x) exp(mean(log(x)))
gm <- geometric_mean(iris$Sepal.Length)

# Calculate Harmonic Mean using the harmonic.mean function from the e1071 package
hm <- harmonic.mean(iris$Sepal.Length)

# Calculate Weighted Mean
# Assuming equal weights for simplicity, as specific weights were not provided.
weights <- rep(1/length(iris$Sepal.Length), length(iris$Sepal.Length))
wm <- weighted.mean(iris$Sepal.Length, w = weights)
```

The Measure of Dispersion or Variance

Measures of dispersion or variability describe the spread or variability of a dataset. These measures give insight into how much the data points diverge from the central value (mean, median, etc.). The main measures of dispersion include the range, variance, standard deviation, and interquartile range (IQR).

- **Range:** The range is the simplest measure of dispersion, representing the difference between the highest and lowest values in a dataset. It gives a basic idea of the spread but can be influenced heavily by outliers.
- **Variance:** Variance measures how far each number in the dataset is from the mean and thus from every other number in the set. It is calculated by averaging the squared differences from the mean. Variance gives a more nuanced view of the dataset's spread, but because it is squared, it is not in the same units as the data.
- **Standard Deviation:** The standard deviation is the square root of the variance. It measures the amount of variation or dispersion of a set of values. Unlike variance, the standard deviation is in the same units as the data, making it more interpretable for describing the variability.
- **Interquartile Range (IQR):** The interquartile range is the difference between the 75th percentile (Q3) and the 25th percentile (Q1) of the dataset. It measures the spread of the middle 50% of the data, providing a view of variability that is less influenced by outliers.

```r
# Load the iris dataset
data(iris)

# Range
range_sepal_length <- range(iris$Sepal.Length)
range_value <- diff(range_sepal_length) # Calculate the actual range value

# Variance
variance_sepal_length <- var(iris$Sepal.Length)

# Standard Deviation
std_dev_sepal_length <- sd(iris$Sepal.Length)

# Interquartile Range (IQR)
iqr_sepal_length <- IQR(iris$Sepal.Length)
```

Relationships to consider

- Harmonic mean <= Geometric mean <= mean
- Semi-IQR (Q3-Q1)/2 – roughly 2/3 of the SD
- Mean deviation – roughly 4/5 of the SD

Graphs, Plots, and Tables

R supports various data types

- **Numeric:** Represents decimal values (e.g., 2.5, -3.14). This is the default for numbers in R.
- **Integer:** Represents whole numbers (e.g., 2L, -3L). The `L` suffix is used to specify an integer.
- **Character:** Represents text (e.g., “Hello”, “R”).
- **Logical:** Represents Boolean values (TRUE or FALSE).
- **Complex:** Represents complex numbers (e.g., 1+4i).
- **Factor:** Used for categorical data. It stores both the actual values and the levels of categorical data.
- **Date and POSIXct:** Represents dates and date-times, respectively.

Data Frames in R

A data frame is a table or a two-dimensional array-like structure in which each column contains values of one variable, and each row contains one set of values from each column. The concept of a data frame comes from the world of statistical software used in empirical research; it has rows and columns. Data frames can contain different types of data in different columns: numeric, character, logical, etc.

Example: Displaying Data Types and Creating a Data Frame

```r
# Displaying different data types
numeric_val <- 3.14
integer_val <- 5L
character_val <- "R is fun"
logical_val <- TRUE

# Print data types
print(paste("Numeric:", numeric_val))
print(paste("Integer:", integer_val))
print(paste("Character:", character_val))
print(paste("Logical:", logical_val))

# Creating a data frame
df <- data.frame(
  Name = c("Alice", "Bob", "Charlie"),
  Age = c(25, 30, 35),
  Height = c(5.5, 6.0, 5.8),
  Married = c(TRUE, FALSE, TRUE)
)

# Display the data frame
print(df)

# Can be converted into a matrix
df <- as.matrix(df)
```

Types of Plots

Scatter Plot - Used to visualize the relationship between two quantitative variables.

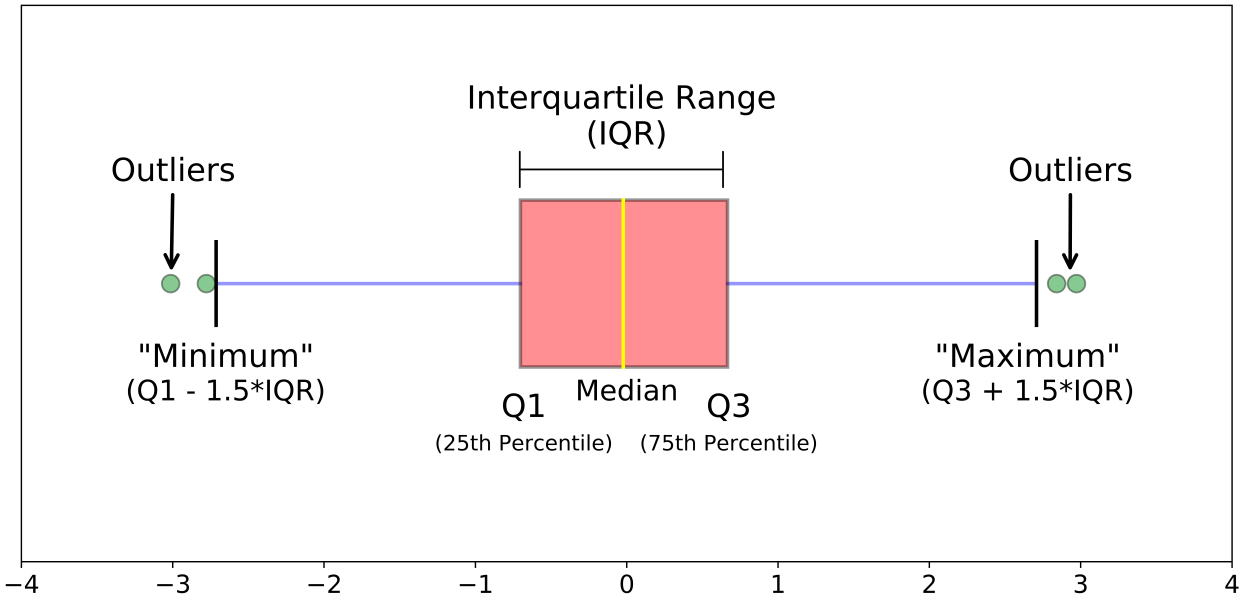
```r
library(ggplot2)
data(iris)

ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width)) +
  geom_point() +
  labs(title = "Scatter Plot of Sepal Length vs Sepal Width",
       x = "Sepal Length",
       y = "Sepal Width")
```

Histogram - Used to visualize the distribution of a single quantitative variable.

```r
ggplot(iris, aes(x = Sepal.Length)) +
  geom_histogram(binwidth = 0.3, fill = "blue", color = "black") +
  labs(title = "Histogram of Sepal Length",
       x = "Sepal Length",
       y = "Count")
```

Box Plot - Used to visualize the distribution of a quantitative variable, highlighting the median, quartiles, and outliers.

- Shows the distribution of a cumulative variable
- For a quantitative variable grouped by a qualitative variable, the distribution within each category is displayed.
- The median inside the box, lower in the upper hinges represents the 25th and 75th percentile
- Hspread –> IQR
- Lower fence = lower hinge + 1.5(Hspread)
- Upper fence = upper hinge + 1.5(Hspread)




```r
ggplot(iris, aes(x = Species, y = Sepal.Length)) +
  geom_boxplot(fill = "lightblue", color = "black") +
  labs(title = "Box Plot of Sepal Length by Species",
       x = "Species",
       y = "Sepal Length")
```

Line Plot - Ideal for visualizing changes in a variable over time or ordered categories.

```r
# Aggregating data for line plot example
iris$ID <- seq_along(iris$Sepal.Length)
iris_aggregated <- aggregate(Sepal.Length ~ Species, data = iris, mean)

ggplot(iris_aggregated, aes(x = Species, y = Sepal.Length, group = 1)) +
  geom_line() +
  geom_point() +
  labs(title = "Line Plot of Mean Sepal Length by Species",
       x = "Species",
       y = "Mean Sepal Length")
```

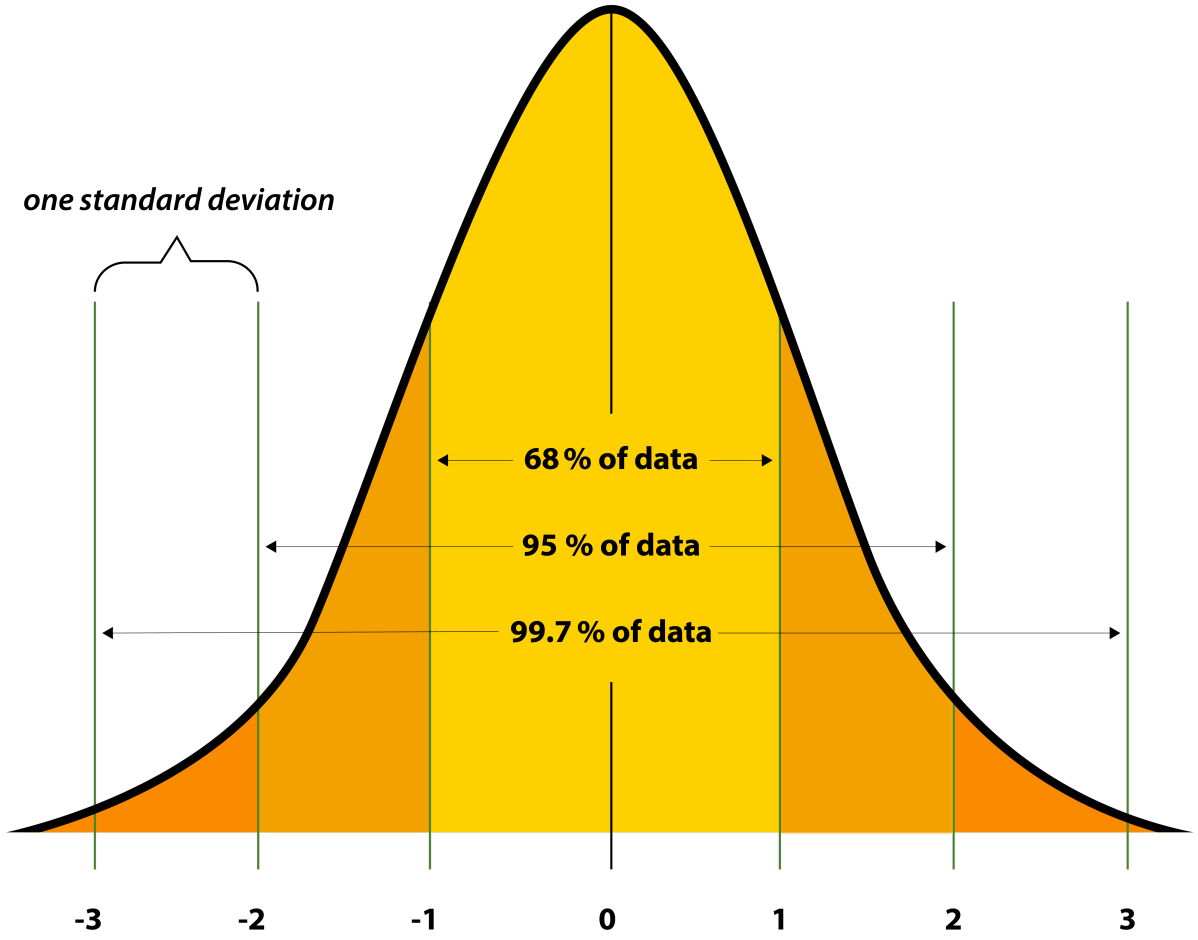
Bar Plot - Used to compare the size of different groups or categories.

```
ggplot(iris, aes(x = Species)) +
  geom_bar(fill = "coral") +
  labs(title = "Bar Plot of Flower Species",
       x = "Species",
       y = "Count")
```

Density Plot - Shows the distribution of a continuous variable with an estimate of the density function.

```
ggplot(iris, aes(x = Sepal.Length)) +
  geom_density(fill = "lightgreen") +
  labs(title = "Density Plot of Sepal Length",
       x = "Sepal Length",
       y = "Density")
```

z-Distribution



- The Z-distribution, often referred to as the standard normal distribution, is a special case of the normal distribution that has a **mean of 0** and a **standard deviation of 1**.
- It is used extensively in hypothesis testing, confidence interval construction, and as a basis for generating other distributions.
- The Z-distribution is particularly useful for standardizing scores from different normal distributions, allowing them to be compared directly. This process is known as Z-transformation or standardization, where scores from a normal distribution are converted into Z-scores.

Uses of Z-Distribution

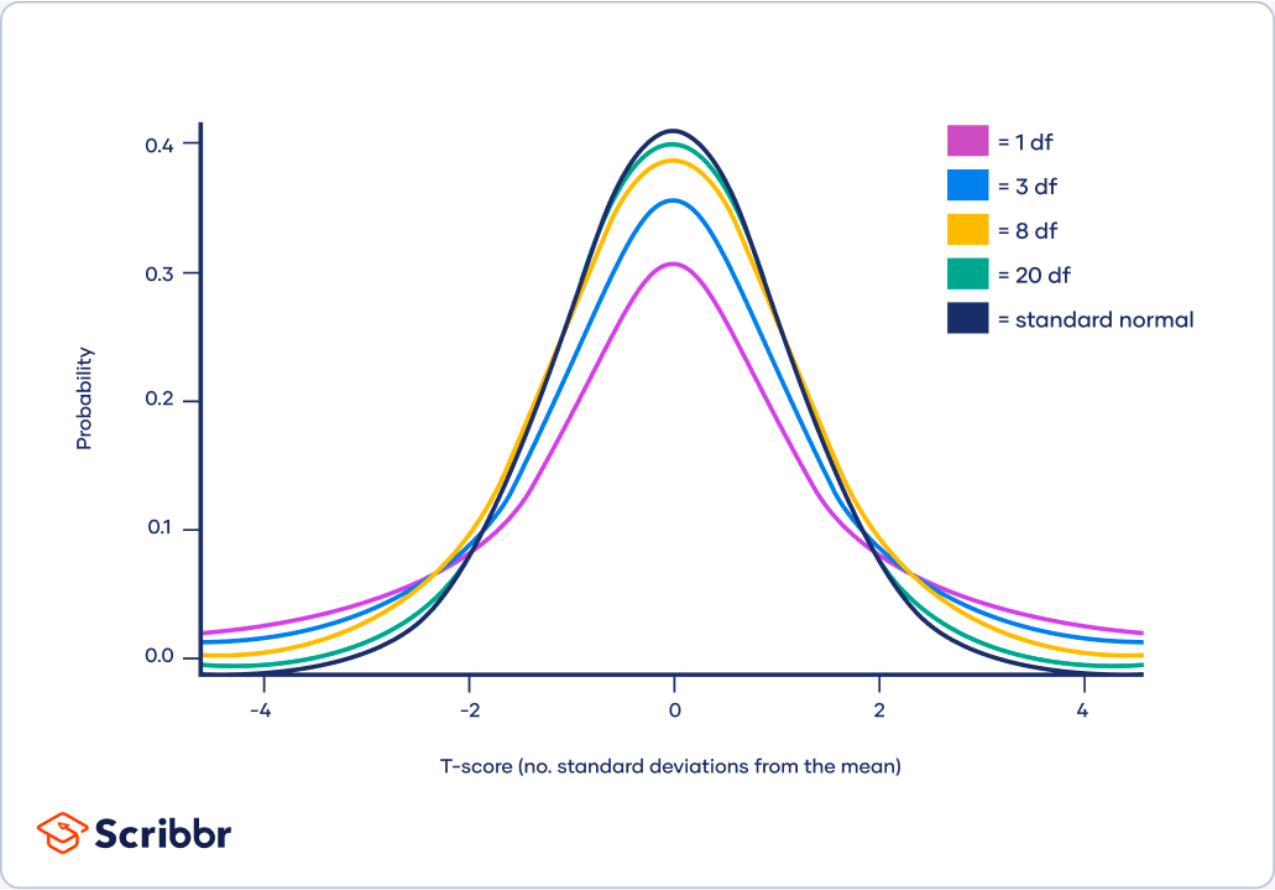
1. **Hypothesis Testing:** It's commonly used in hypothesis testing to determine the likelihood of observing a test statistic as extreme as, or more extreme than, the observed value under the null hypothesis.
2. **Confidence Intervals:** Z-distribution is used to construct confidence intervals for population parameters (like the mean) when the population standard deviation is known or the sample size is large.
3. **Standardization:** Converting raw scores into Z-scores allows comparison across different units or scales, facilitating the comparison of data from different sources.

Z-distribution Assumptions

1. **Normality:** The Z-distribution is used when the population from which the sample is drawn is assumed to be normally distributed. This assumption is less critical when using the Z-distribution for large samples due to the Central Limit Theorem, which states that the sampling distribution of the sample mean approaches a normal distribution as the sample size increases, regardless of the population distribution.
2. **Known Population Standard Deviation:** Another key assumption for using the Z-distribution is that the population standard deviation (σ) is known. This is often unrealistic in practical scenarios, where σ is usually unknown and must be estimated from the sample.
3. **Large Sample Size:** While the Z-distribution can be applied to both large and small samples, in practice, it's typically used for large samples (n > 30) because the Central Limit Theorem assures that the sampling distribution of the mean is approximately normal, facilitating the use of Z-scores.

```
# Load the Iris dataset
data(iris)

# Use the scale function to calculate Z-scores for Sepal.Length
sepal_length_z <- scale(iris$Sepal.Length)

# The scale function returns a matrix. To simplify, we can convert it to a vector
sepal_length_z_vector <- as.vector(sepal_length_z)

# Add the Z-scores as a new column to the Iris dataset
iris$Sepal.Length.Z <- sepal_length_z_vector

# View the first few rows of the updated dataset to see the Z-scores
head(iris)
```

t-Distribution



With >30 degrees of freedom, the t-distribution starts to resemble a standard normal distribution.
The t-distribution, also known as Student's t-distribution, is a type of probability distribution that is symmetric and bell-shaped, similar to the normal distribution, but with heavier tails. It arises when estimating the mean of a normally distributed population in situations where the sample size is small and the population standard deviation is unknown.

Uses of t-Distribution

- **Estimating Means:** The t-distribution is used to estimate the population mean from a small sample size with an unknown population standard deviation.
- **Hypothesis Testing:** It's particularly useful in hypothesis testing scenarios, such as the t-test, for comparing the means of two samples or for testing a hypothesis about a single mean.

- **Confidence Intervals:** The t-distribution is used to construct confidence intervals for population means when the sample size is small and the population standard deviation is unknown.

t-distribution Assumptions

1. **Normality of the Population:** Similar to the Z-distribution, the t-distribution assumes that the underlying population from which the sample is drawn is normally distributed. However, the t-test, which uses the t-distribution, is relatively robust to deviations from this assumption, especially as the sample size increases.
2. **Unknown Population Standard Deviation:** The t-distribution is specifically designed for situations where the population standard deviation is unknown and has to be estimated from the sample. This makes the t-distribution more applicable in real-world scenarios than the Z-distribution.
3. **Small Sample Size:** The t-distribution is particularly useful for small sample sizes (typically n < 30), where the uncertainty in estimating the population standard deviation significantly affects the shape of the sampling distribution of the mean. As the sample size increases, the t-distribution approaches the Z-distribution (normal distribution), making them virtually indistinguishable for large samples.

Takeaway: Normal distribution of data, random sampling, and scale of measurement are crucial assumptions for using the t distribution.

Characteristics of the t-Distribution

- **Shape:** The t-distribution is similar to the standard normal distribution but with thicker tails. These thicker tails reflect the increased variability and uncertainty in estimating the population mean from a small sample.
- **Degrees of Freedom (df):** The shape of the t-distribution is determined by its degrees of freedom, which are typically related to the sample size. For a single sample, the degrees of freedom are calculated as $df=n-1$, where n is the sample size.

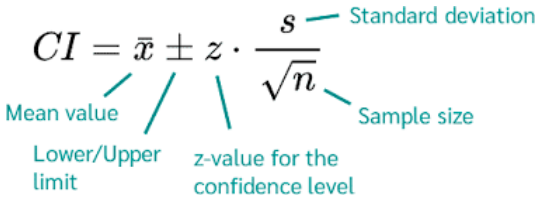
```
# Load the Iris dataset
data(iris)

# Extract sepal lengths for setosa and versicolor species
setosa_sepal_length <- iris$Sepal.Length[iris$Species == 'setosa']
versicolor_sepal_length <- iris$Sepal.Length[iris$Species == 'versicolor']

# Perform a t-test to compare the mean sepal length between setosa and versicolor
t_test_result <- t.test(setosa_sepal_length, versicolor_sepal_length)

# Print the result of the t-test
print(t_test_result)
```

Confidence Intervals

- Confidence intervals (CIs) are a range of values, derived from the sample data, that are believed to contain the value of an unknown population parameter (e.g., the mean, proportion) with a certain level of confidence.
- They provide a measure of the precision of an estimate from a sample and a range within which the true population parameter is likely to lie. T
- he width of the confidence interval gives us an idea of how uncertain we are about the true value of the parameter; narrower intervals indicate less uncertainty.
- Used to estimate the range within which future estimates (like the mean) will fall with a certain level of confidence.

Key Concepts:

- **Confidence Level:** This is the percentage (usually expressed as a decimal) that indicates how confident we can be that the interval contains the true parameter. Common confidence levels include 90%, 95%, and 99%.
- **Margin of Error:** This represents how much we expect the estimate to vary if we were to take multiple samples. It's affected by the sample size and the variability in the data.




```
# Assuming you have a numeric vector `data` representing your sample
data <- c(1, 2, 3, 4, 5) # Example data

# Calculate a 95% confidence interval for the mean
ci <- t.test(data, conf.level = 0.95)

# Print the confidence interval
ci$conf.int
```

Statistical Hypothesis Testing

- It's a process that involves making an assumption (the null hypothesis) and then using statistical tests to determine whether to reject this assumption in favor of an alternative hypothesis.
- Typically starts by assuming no difference (null hypothesis) and then uses the data to test this assumption.
- Aims to determine whether the observed data is significantly different from what would be expected under the null hypothesis.

More information here: https://www.nlm.nih.gov/oet/ed/stats/index.html (https://www.nlm.nih.gov/oet/ed/stats/index.html)

Hypothesis Testing Approach:

Six steps are generally recommended for conducting hypothesis tests.

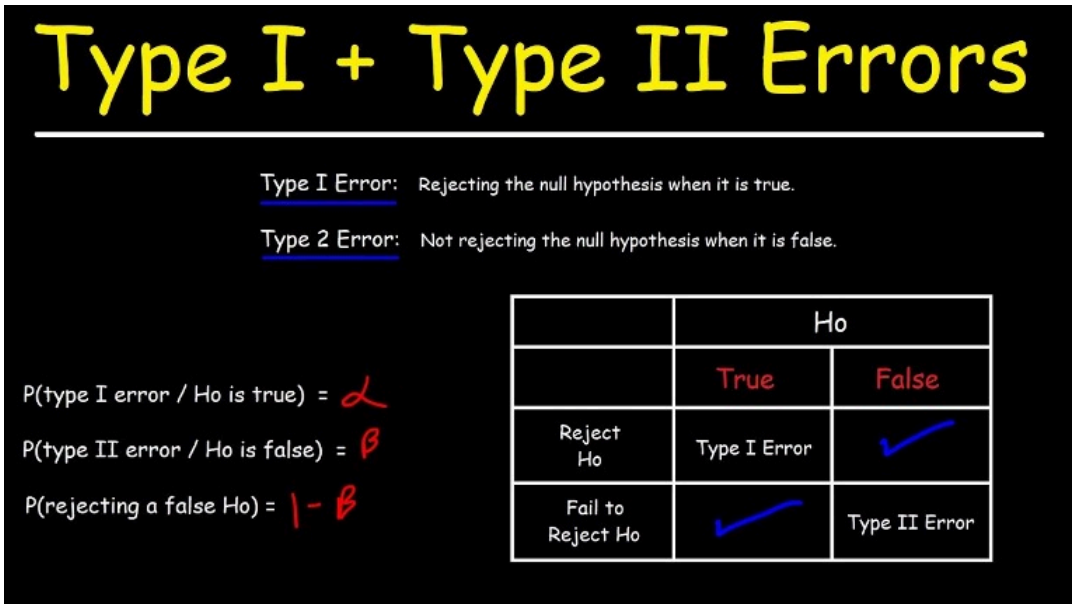
1. **Define the Null and Alternative Hypotheses**
    - **Null Hypothesis (*H0*)**: This is a statement of no effect or no difference, serving as the default assumption. It's what you seek to test against.
    - **Alternative Hypothesis (Ha)**: This statement indicates the presence of an effect or a difference. If the null hypothesis is rejected, the alternative hypothesis is supported.
2. **Choosing significance level *α*(e.g. 0.05 - 95%)**
    - The significance level is the probability of rejecting the null hypothesis when it is true, a type I error. Common choices for *α* include 0.05, 0.01, and 0.10. A lower *α* means you require stronger evidence to reject the null hypothesis.
3. **Select the Appropriate Test and Calculate the Test Statistic**
    - Choose a statistical test based on the type of data, the distribution of the data, and the hypothesis being tested (e.g., t-test for means, chi-square test for categorical data, ANOVA for comparing more than 2 groups, etc.).
    - Calculate the test statistic using sample data, which will measure how far the sample statistic diverges from the null hypothesis.
4. **Determine the Critical Value or p-value**
    - **Critical Value:** This is a threshold that the test statistic must exceed to reject the null hypothesis. It's determined by the chosen significance level and the distribution of the test statistic.
    - **p-value:** This is the probability of observing a test statistic as extreme as, or more extreme than, the observed value, under the assumption that the null hypothesis is true.
5. **Make the Decision**
    - Compare the test statistic to the critical value, or compare the p-value to the significance level:
        - If the test statistic exceeds the critical value, or if the p-value is less than *α*, reject the null hypothesis.
        - Otherwise, do not reject the null hypothesis. (**Note: This does not mean you accept the null hypothesis as true, only that there's not enough evidence to reject it**.)
6. **Conclude**
    - Clearly state your conclusion in the context of the hypothesis test. This includes whether you rejected or did not reject the null hypothesis and what that implies about the alternative hypothesis.

Confidence Intervals vs. Hypothesis Tests

- Confidence intervals are preferred in a clinical setting.
- **Estimation vs. Decision-Making:** Confidence intervals are used for estimation, providing a range of values that likely include the population parameter. Hypothesis tests are used for decision-making, determining whether there is enough evidence to support a specific claim about the population parameter.
- **Information Conveyed:** Confidence intervals provide more information by estimating the parameter value and its precision. Hypothesis tests provide a binary decision (reject or not reject the null hypothesis) and the strength of evidence against the null hypothesis (p-value).
- **Perspective:** CIs offer a range that is likely to contain the parameter, considering the data variability and sample size. Hypothesis testing assesses whether the observed data would be rare if the null hypothesis were true, without directly estimating the parameter value.

Type I and II Errors

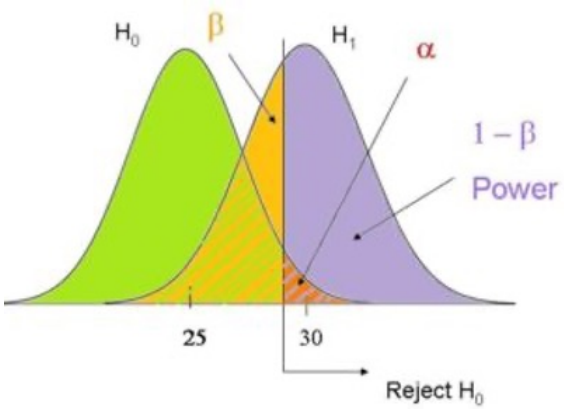
- Type I error (α error): Incorrectly rejecting a true null hypothesis.
- Type II error (β error): Failing to reject a false null hypothesis.



Picture1.jpg

Statistical Power

- Power is the probability of correctly rejecting a false null hypothesis (1 - β).
- Depends on sample size, effect size, and significance level.



Picture1.jpg

Types of t-Tests and How to Perform Them

- There are several types of t-tests, each designed for specific cases.
- Additionally, when comparing the means of more than two groups, an Analysis of Variance (ANOVA) test is used.

Below, we'll explore these tests and provide examples using R's `iris` dataset.

Types of t-Tests

1. **One-Sample t-Test**
   - **Purpose:** To compare the mean of a single group against a known mean.
   - **R Function:** `t.test(x, mu = mean_value)` where `x` is your sample data and `mu` is the known mean you're comparing against.
2. **Two-Sample t-Test (Independent Samples)**
   - **Purpose:** To compare the means of two independent groups.
   - **R Function:** `t.test(x, y)` where `x` and `y` are the two independent samples.

```
data(iris)
# Subset data for two samples
iris = subset(iris, Species %in% c("setosa", "versicolor")

# Calculate the p-value using R's t.test function
t.test(Sepal.Length ~ Species, data =iris, paired=False))
```

3. **Paired t-Test (Dependent Samples)**
   - **Purpose:** To compare the means of two related groups.

- **R Function:** `t.test(x, y, paired = TRUE)` where `x` and `y` are the two related samples, such as measurements taken before and after a treatment on the same subjects.
- We set `paired = TRUE` when we know that they are testing for differences within the same group (paired t-test).

ANOVA Test

- **Purpose:** To compare the means of three or more groups to see if at least one of them is statistically different from the others.
- **R Function:** `aov(response ~ group, data = your_data)` where `response` is the continuous variable you're testing, and `group` is the categorical variable defining the groups.
- When you perform an ANOVA and find a significant effect, this tells you that at least one group's mean is different from the others, but it doesn't tell you which groups are different.
- To identify the specific differences between group means, you can use post hoc tests such as Tukey's Honest Significant Difference (HSD) test.
- Tukey's HSD is a popular method for pairwise comparisons because it controls for the Type I error rate across multiple comparisons.
- Post-hoc (a posteriori) comparisons – after an ANOVA has resulted in a significant F test – decide which comparisons to make after looking at the data – t-test is not suitable here

Multiple comparison procedures

- A posteriori, or post-hoc, comparisons

Approach	Tukey's HSD Procedure	Scheffé's Procedure	Newman–Keuls Procedure	Dunnett's Procedure	Games-Howell procedure
Assumption	Equal variance assumed, equal sample size NOT assumed	Equal variance assumed, equal sample size NOT assumed	Equal variance and sample size assumed	Equal variance NOT assumed, equal sample size NOT assumed	Equal variance NOT assumed, equal sample size NOT assumed
Details	• Only pairwise comparisons • Has a multiplier based on the number of treatment levels and the degrees of freedom for error mean square $HSD = \text{Multiplier} \times \sqrt{\frac{MS_E}{n}}$ The values for the multipliers can be found in White: Table 7-6, Page 16/33.	• All possible simple and complex paris of comparisons (most versatile) • E.g., compare the overall mean of two or more dosage level with a placebo • A higher critical value is used to determine significance (most conservative) $S = \sqrt{(j-1)F_{\alpha,df}} \sqrt{MS_E \sum \frac{C_j^2}{n_j}}$	• Only pairwise comparisons • Depends on the steps that separate pairs of means • Less conservative than Tukey's test • Can't form confidence intervals for mean differences $\text{Newman-Keuls} = \text{Multiplier} \times \sqrt{\frac{MS_E}{n}}$ can be found in White: Table 7-6, Page 16/33.	• Only in situations in which several treatment means are compared with a single control mean • Can't compare between treatment means • Relatively low critical value $\text{Dunnett's procedure} = \text{Multiplier} \times \sqrt{\frac{2MS_E}{n}}$ The values for the multipliers can be found in White: Table 7-6, Page 16/33.	• When statistical power is to be considered, this test consistently provides narrower confidence intervals • Robust to non normality
Example	For three pairwise comparisons, α = 0.05 and assuming approximately 120 degrees of freedom, the multiplier is 3.36		For three pairwise comparisons, α = 0.05 and assuming approximately 120 degrees of freedom, the multiplier is 2.8	For three pairwise comparisons, α = 0.05 and assuming approximately 120 degrees of freedom, the multiplier is 1.95	

Picture1.png

- Use the rule of thumb ratio. As a rule of thumb, if the ratio of the larger variance to the smaller variance is less than 4, then we can assume the variances are approximately equal and use the two-sample t-test.
- Utilized with more than two groups.
- Between-group variation and overall group variation

Assumption for AVNOVA

1. Variables are normally distributed.
2. Population variance is the same in each group – Levene's and Bartlett's test for equality of variance (check beforehand).
3. The observations are independent in that the values of one observation are not related to the value of another observation – equal, not equal

Takeaway: normality, homogeneity of variance, and Independence of observation,

```
data(iris)

# ANOVA
anova_result <- aov(Sepal.Length ~ Species, data = iris)
summary(anova_result)

# Conduct Tukey's HSD test
tukey_result <- TukeyHSD(anova_result)

# Print the results
print(tukey_result)
```

Here, we show an example of how to add significance bars for results from a simple t.test

```
data(ToothGrowth)

library(rstatix)
library(ggpubr)

p <- ggboxplot(ToothGrowth, x = "supp", y = "len",
          color = "supp", palette = "jco",
          add = "jitter")

#  Add p-value
p + stat_compare_means()

# Change method
p + stat_compare_means(method = "t.test")
```

Here, we show a plot with the global p-value from the ANOVA test. This helps us confirm that the variances between the groups are not the same.

```
# Global test
compare_means(len ~ dose,  data = ToothGrowth, method = "anova")
```

```
# Change method to anova
ggboxplot(ToothGrowth, x = "dose", y = "len",
          color = "dose", palette = "jco")+
  stat_compare_means(method = "anova")
```

Here we compute Tukey's post hoc test after performing ANOVA to identify significant pairs.

```
# Data preparation
df <- ToothGrowth
df$dose <- as.factor(df$dose)

# Tukey HSD
library(rstatix)  # https://github.com/kassambara/rstatix
stat.test <- aov(len ~ dose, data = df) %>% tukey_hsd()
stat.test

# Visualize
library(ggpubr)
ggboxplot(df, x = "dose", y = "len") +
  stat_pvalue_manual(
    stat.test, label = "p.adj",
    y.position = c(29, 35, 39))
```