STATISTICAL DATA ANALYSIS IN R

Statistical Computing in R

Lecturer:Ivan Innocent Sekibenga

1. Statistical Data Analysis overview

Statistical data analysis is the process of collecting, organizing, analyzing, interpreting, and presenting data to uncover patterns and trends. It plays a vital role in various fields such as business, healthcare, social sciences, and more. R is a popular programming language for statistical data analysis due to its extensive libraries and packages that facilitate data manipulation, visualization, and modeling.

Statistical data analysis is a crucial step in understanding and interpreting data. It involves various techniques and methods to summarize, visualize, and draw conclusions from data sets.

1.1 Common Statistical Techniques in R

The common statistical techniques in R include;

- **descriptive statistics**:This technique involves summarizing and describing the main features of a data set. Common descriptive statistics include measures of central tendency (mean, median, mode) and measures of variability (standard deviation, variance, range).
- **correlation analysis**:This technique is used to measure the strength and direction of the relationship between two variables. The most common correlation coefficient is Pearson's correlation coefficient. Other types include Spearman's rank correlation and Kendall's tau.
- **inferential statistics**:This technique involves making inferences about a population based on a sample of data. Common inferential statistics techniques include confidence intervals, hypothesis testing, and regression analysis.
- **regression analysis**: This technique is used to model the relationship between a dependent variable and one or more independent variables. Common types of regression analysis include linear regression, logistic regression, and multiple regression.
- **hypothesis testing**: This technique is used to test a hypothesis about a population based on a sample of data. Common hypothesis testing techniques include t-tests, chi-square tests, and ANOVA.

1.2 Assumptions and Prerequisites

Going forward, i assume that you have a basic statistical theory of these techniques and hence i will not go into the details of the statistical theory behind each technique. However, i will provide a brief overview of the assumptions and prerequisites for each technique where applicable. I will focus on the practical implementation of these techniques in R.

2. Descriptive Statistics

Descriptive statistics provide a summary of the main features of a data set. Common descriptive statistics include measures of central tendency (mean, median, mode) and measures of variability (standard deviation, variance, range).

When applicable:

- During Exploratory Data Analysis (EDA) before applying complex models.
- When you need to summarize a large dataset into meaningful patterns.
- To understand basic structure, distribution, or detect outliers.

Limitations:

- Descriptive statistics do not make inferences beyond the given data.
- They ignore relationships between variables.
- Sensitive to outliers, especially the mean.

Example 2.1: one column vector data

```
data <- c(10, 20, 30, 40, 50) # sample data

mean_value <- mean(data) # calculate mean
median_value <- median(data)  # calculate median
sd_value <- sd(data)     # calculate standard deviation

mean_value     # print mean value
```

```
## [1] 30
```

```
median_value    # print median value
```

```
## [1] 30
```

```
sd_value       # print standard deviation
```

```
## [1] 15.81139
```

Example 2.2: data frame of two variables

```
data <- data.frame(height = c(150, 160, 170, 180, 190),
                   weight = c(50, 60, 70, 80, 90))

mean_height <- mean(data$height) # calculate mean height
mean_weight <- mean(data$weight) # calculate mean weight
                                 # $ accesses columns in data frame
mean_height  # print mean height
```

```
## [1] 170
```

```
mean_weight  # print mean weight
```

```
## [1] 70
```

Example 2.3: Summary statistics for all variables in a data frame

```
# Load the built-in dataset
data(mtcars) # mtcars dataset contains information about various car models such as miles per gallon, h
```

```r
# Display summary statistics for all numeric columns
summary(mtcars)
```

```
##       mpg             cyl             disp             hp
##  Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0
##  1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5
##  Median :19.20   Median :6.000   Median :196.3   Median :123.0
##  Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7
##  3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0
##  Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0
##       drat             wt             qsec             vs
##  Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000
##  1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000
##  Median :3.695   Median :3.325   Median :17.71   Median :0.0000
##  Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375
##  3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000
##  Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000
##        am              gear            carb
##  Min.   :0.0000   Min.   :3.000   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
##  Median :0.0000   Median :4.000   Median :2.000
##  Mean   :0.4062   Mean   :3.688   Mean   :2.812
##  3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

```r
# summary() function provides a quick overview of the dataset,
# including min, max, mean, median, and quartiles for each numeric column.
```

Interpretation:

This gives minimum, 1st quartile, median, mean, 3rd quartile, and maximum — a quick overall picture of your dataset.

Example 2.4: Central tendency and spread

```r
# Calculate basic descriptive statistics for 'mpg' (miles per gallon)

mean(mtcars$mpg)    # Mean (average value)
```

```
## [1] 20.09062
```

```r
median(mtcars$mpg)  # Median (middle value)
```

```
## [1] 19.2
```

```r
sd(mtcars$mpg)      # Standard deviation (variability)
```

```
## [1] 6.026948
```

```r
var(mtcars$mpg)     # Variance
```

```
## [1] 36.3241
```

```r
range(mtcars$mpg)   # Minimum and maximum
```

```
## [1] 10.4 33.9
```

Interpretation:

The mean shows average fuel efficiency; SD and variance show how spread out mpg values are.

Example 2.5: Grouped summary using dplyr

```
library(tidyverse) # helps to load dplyr for data manipulation
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.1     v stringr   1.5.2
## v ggplot2   4.0.0     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
# Group data by number of cylinders and summarize mpg
mtcars %>%
  group_by(cyl) %>%
  summarise(
    Mean_MPG = mean(mpg),
    SD_MPG = sd(mpg),
    Count = n()
  )
```

```
## # A tibble: 3 x 4
##     cyl Mean_MPG SD_MPG Count
##   <dbl>    <dbl>  <dbl> <int>
## 1     4     26.7   4.51    11
## 2     6     19.7   1.45     7
## 3     8     15.1   2.56    14
```

Interpretation:

This shows how fuel efficiency (mpg) varies by engine size (cylinders). Cars with more cylinders tend to have lower fuel efficiency. Group summaries are crucial for comparisons across categories.

Example 2.6: Frequency distribution

```
# Count frequency of each number of cylinders
table(mtcars$cyl)
```

```
##
##  4  6  8
## 11  7 14
```

Interpretation:

This shows how many cars have 4, 6, or 8 cylinders. Frequency distributions help understand the composition of categorical variables.

3. Correlation Analysis

Correlation measures how strongly two numeric variables are related and the direction of their relationship (positive, negative, or none). The three common correlation coefficients are **Pearson's**, **Spearman's**, and

Kendall's. Their values range from -1 to +1, where +1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation.

When applicable:

- To explore whether two variables change together (e.g., income vs. education).
- Before regression, to check potential multicollinearity.
- During exploratory analysis for relationship detection.

Limitations

- Correlation does not imply causation.
- Pearson assumes linearity and normality.
- Spearman/Kendall measure monotonic, not necessarily linear, relationships.
- Sensitive to outliers.

Example 3.1: Pearson correlation

```
# Compute Pearson correlation between mpg and horsepower
cor(mtcars$mpg, mtcars$hp, method = "pearson")
```

```
## [1] -0.7761684
```

Interpretation:

A negative value indicates that as horsepower increases, miles per gallon decreases, suggesting an inverse relationship between engine power and fuel efficiency.

Example 3.2: Spearman correlation

```
# Spearman correlation (rank-based, non-parametric)
cor(mtcars$mpg, mtcars$wt, method = "spearman")
```

```
## [1] -0.886422
```

Interpretation:

A negative value indicates that as weight increases, miles per gallon decreases, suggesting an inverse monotonic relationship between vehicle weight and fuel efficiency.

Example 3.3: Kendall's Tau Correlation

```
# Kendall correlation for small or ordinal data
cor(mtcars$mpg, mtcars$qsec, method = "kendall")
```

```
## [1] 0.3153652
```

Interpretation:

A negative value indicates that as quarter-mile time increases, miles per gallon decreases, suggesting an inverse monotonic relationship between acceleration and fuel efficiency.

Example 3.4: Correlation matrix for multiple variables

```
# Generate correlation matrix for selected variables
cor(mtcars[, c("mpg", "hp", "wt", "qsec")])
```

```
##             mpg         hp         wt       qsec
## mpg   1.0000000 -0.7761684 -0.8676594  0.4186840
## hp   -0.7761684  1.0000000  0.6587479 -0.7082234
## wt   -0.8676594  0.6587479  1.0000000 -0.1747159
## qsec  0.4186840 -0.7082234 -0.1747159  1.0000000
```

Interpretation:

This matrix shows pairwise correlations among multiple variables, helping identify which variables are strongly related. It uses Pearson's method by default.

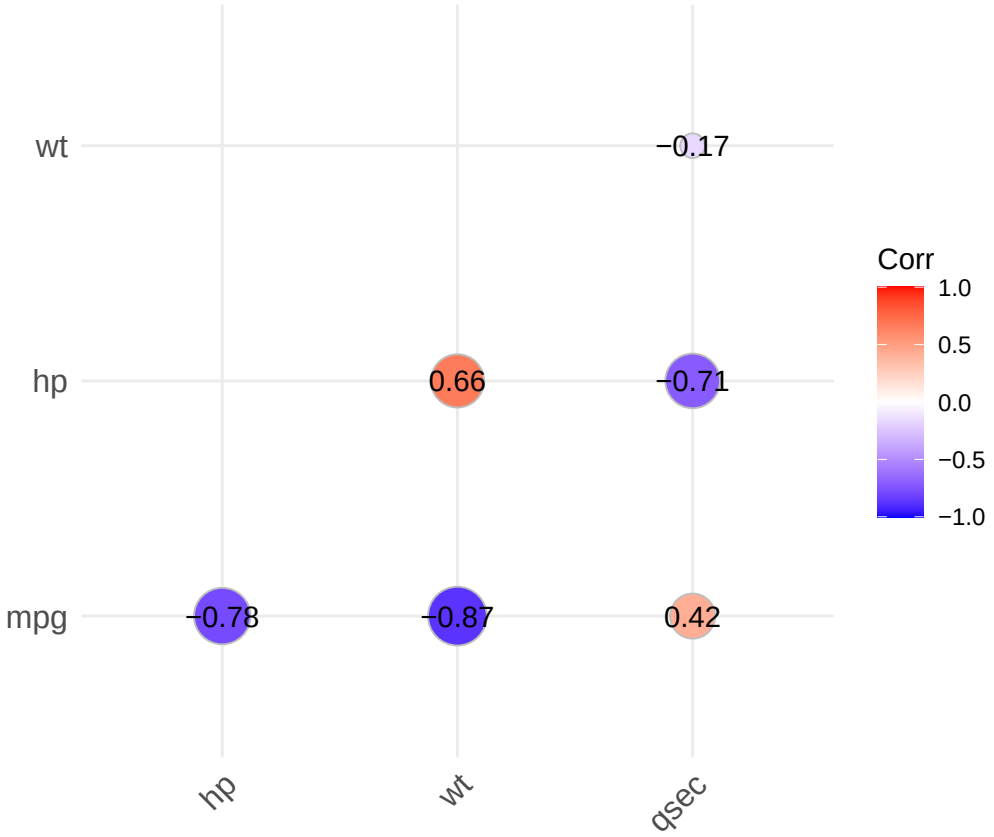
Example 3.5: Visualizing correlations with a heatmap

```
library(ggcorrplot) # for correlation heatmaps

# Compute correlation matrix
corr_matrix <- cor(mtcars[, c("mpg", "hp", "wt", "qsec")])

# Plot correlation heatmap
ggcorrplot(corr_matrix, method = "circle", type = "lower", lab = TRUE)
```

```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation idioms with `aes()`.
## i See also `vignette("ggplot2-in-packages")` for more information.
## i The deprecated feature was likely used in the ggcorrplot package.
##   Please report the issue at <https://github.com/kassambara/ggcorrplot/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```


wt
hp
mpg
−0.17
0.66
−0.71
−0.78
−0.87
0.42
hp
wt
qsec
Corr
1.0
0.5
0.0
−0.5
−1.0

Interpretation:

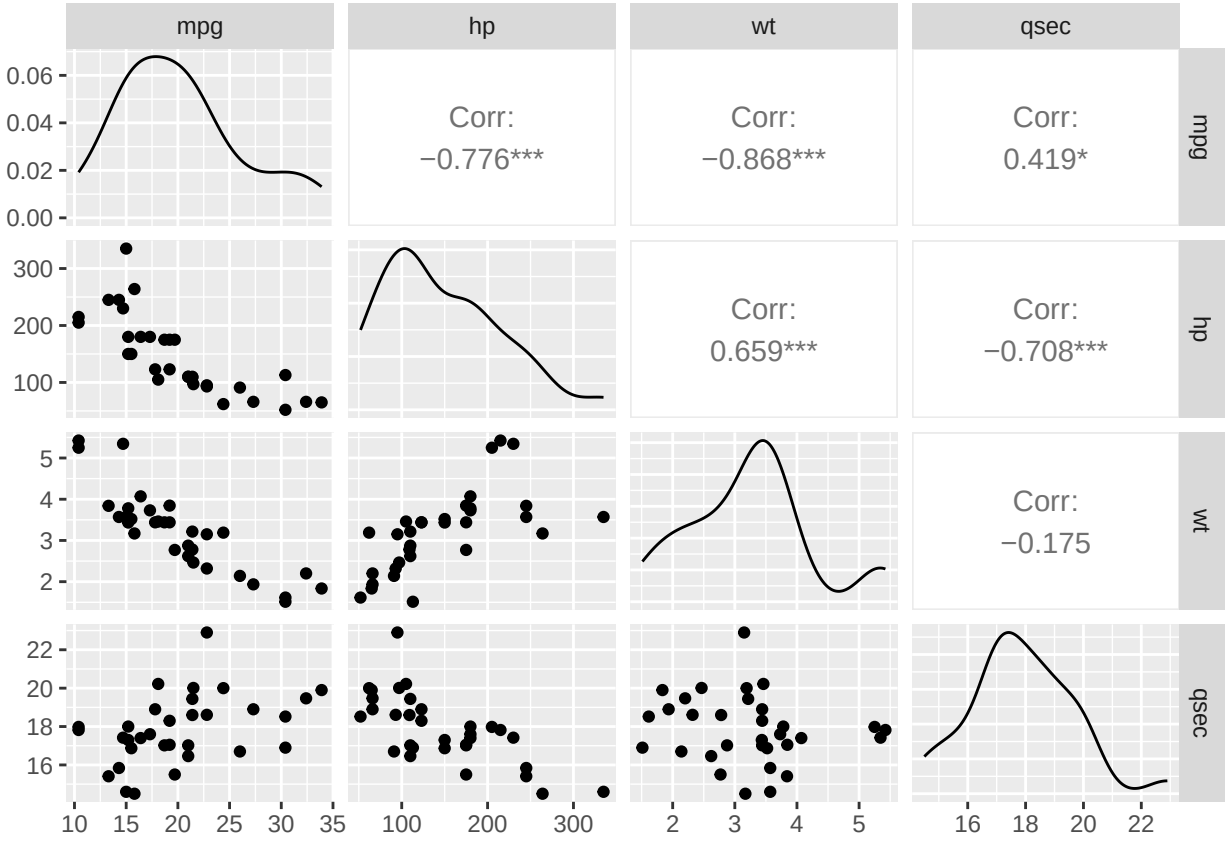
The heatmap visually represents the strength and direction of correlations among variables. Darker colors indicate stronger correlations, while lighter colors indicate weaker correlations.

Example 3.6: Scatterplot matrix

Alternatively, We can also use GGally package for a more enhanced scatterplot matrix.

```
# install.packages("GGally") # Uncomment if GGally is not installed

library(GGally)
ggpairs(mtcars[, c("mpg", "hp", "wt", "qsec")] )
```



Interpretation:

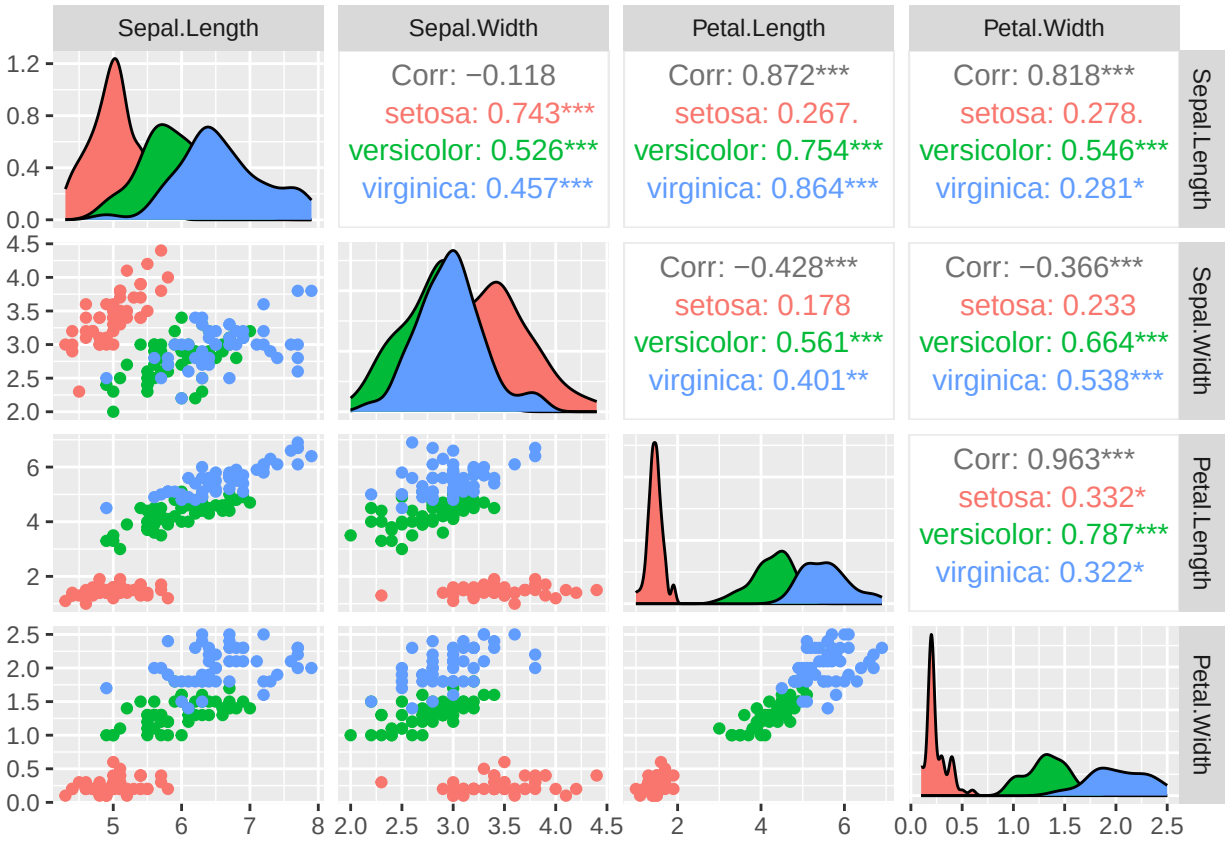
The scatterplot matrix provides pairwise scatter plots for multiple variables, allowing visual assessment of relationships and potential correlations between them.

Example 3.7: Scatterplot matrix with color by species

```
# Example using the iris dataset

data(iris) # iris dataset contains measurements of iris flowers
# from three different species namely: setosa, versicolor, and virginica.

# Colorful scatter plot matrix
ggpairs( iris,aes(color = Species),columns = 1:4)
```



Interpretation:

The scatterplot matrix shows pairwise relationships between the four measurements of iris flowers, colored by species. This helps visualize how different species cluster based on their measurements.

4. Inferential Statistics

Inferential statistics allow us to make inferences about a population based on a sample of data. Common inferential statistics techniques include **confidence intervals** and **hypothesis testing**. In simple terms, Inferential statistics draw conclusions about populations based on sample data.They involve estimation (e.g., confidence intervals) and testing hypotheses about parameters.

When applicable

- When generalizing from a sample to a population.
- When testing hypotheses or comparing means/proportions.
- When estimating unknown parameters.

Limitations

- Assumes random sampling and representative data.
- Many tests require normality and equal variance.
- Misinterpretation of p-values is common.

Example 4.1: Confidence Interval for a Mean

```
# 95% confidence interval for Sepal.Length
t.test(iris$Sepal.Length)$conf.int
```

```
## [1] 5.709732 5.976934
## attr(,"conf.level")
## [1] 0.95
```

Interpretation:

This interval estimates the range in which the true mean Sepal Length of the iris population lies with 95% confidence.

Example 4.2: One-sample t-Test

```
# Test if mean Sepal.Length differs from 5.5

t.test(iris$Sepal.Length, mu = 5.5)
```

```
##
##  One Sample t-test
##
## data:  iris$Sepal.Length
## t = 5.078, df = 149, p-value = 1.123e-06
## alternative hypothesis: true mean is not equal to 5.5
## 95 percent confidence interval:
##  5.709732 5.976934
## sample estimates:
## mean of x
##  5.843333
```

```
# Null hypothesis is  mean = 5.5, Alternative hypothesis is  mean is not 5.5
# If p value is less than 0.05, reject the null; the true mean differs significantly from 5.5
```

Interpretation:

The p-value indicates whether we can reject the null hypothesis that the mean Sepal Length is 5.5. A low p-value (< 0.05) suggests a significant difference.

Example 4.3: Two-sample t-Test

```
# Compare Sepal.Length between Setosa and Versicolor

setosa <- subset(iris, Species == "setosa")$Sepal.Length
versicolor <- subset(iris, Species == "versicolor")$Sepal.Length

t.test(setosa, versicolor)
```

```
##
##  Welch Two Sample t-test
##
## data:  setosa and versicolor
## t = -10.521, df = 86.538, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  -1.1057074 -0.7542926
## sample estimates:
## mean of x mean of y
##     5.006     5.936
```

Interpretation:

The p-value indicates whether there is a significant difference in mean Sepal Length between the two species. A low p-value (< 0.05) suggests a significant difference. A p-value less than 0.05 implies there is a significant difference in average Sepal.Length between species.

Example 4.4: Chi-Square Test of Independence

```r
# Create a contingency table of Species vs. Sepal.Width category

iris$Sepal.Width.Category <- ifelse(iris$Sepal.Width > 3.0, "Wide", "Narrow")

contingency_table <- table(iris$Species, iris$Sepal.Width.Category)

# Perform Chi-Square test
chisq.test(contingency_table)
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 50.225, df = 2, p-value = 1.241e-11
```

Interpretation:

The p-value indicates whether there is a significant association between species and Sepal Width category. A low p-value (< 0.05) suggests a significant association.

5. Regression Analysis

Regression models the relationship between a dependent variable and one or more independent variables. Regression models are used for prediction and explanation.

When applicable

- To predict an outcome based on known factors.
- To estimate effect sizes of predictors.
- When examining relationships between continuous variables.

Limitations

- Assumes linearity, independence, and homoscedasticity.
- Sensitive to outliers.
- Multicollinearity can distort estimates.

Example 5.1: Simple Linear Regression

```r
# Model mpg as a function of weight
model1 <- lm(mpg ~ wt, data = mtcars)

# View summary of model

summary(model1)
```

```
##
## Call:
## lm(formula = mpg ~ wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5432 -2.3647 -0.1252  1.4096  6.8727
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
## wt           -5.3445     0.5591  -9.559 1.29e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.046 on 30 degrees of freedom
## Multiple R-squared:  0.7528, Adjusted R-squared:  0.7446
## F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```

Interpretation:

The summary provides coefficients, R-squared, and p-values. The coefficient for wt indicates how much mpg decreases for each unit increase in weight. A low p-value (< 0.05) for wt suggests it is a significant predictor of mpg.

Slope: change in mpg per unit of wt

R^2: proportion of variance explained

$p < 0.05$ implies predictor is significant.

Example 5.2: Multiple Linear Regression

```r
# Model mpg as function of multiple predictors
model2 <- lm(mpg ~ wt + hp + cyl, data = mtcars)
summary(model2)
```

```
##
## Call:
## lm(formula = mpg ~ wt + hp + cyl, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9290 -1.5598 -0.5311  1.1850  5.8986
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 38.75179    1.78686  21.687  < 2e-16 ***
## wt          -3.16697    0.74058  -4.276 0.000199 ***
```

```
## hp          -0.01804    0.01188  -1.519 0.140015
## cyl         -0.94162    0.55092  -1.709 0.098480 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.512 on 28 degrees of freedom
## Multiple R-squared:  0.8431, Adjusted R-squared:  0.8263
## F-statistic: 50.17 on 3 and 28 DF,  p-value: 2.184e-11
```

Interpretation:

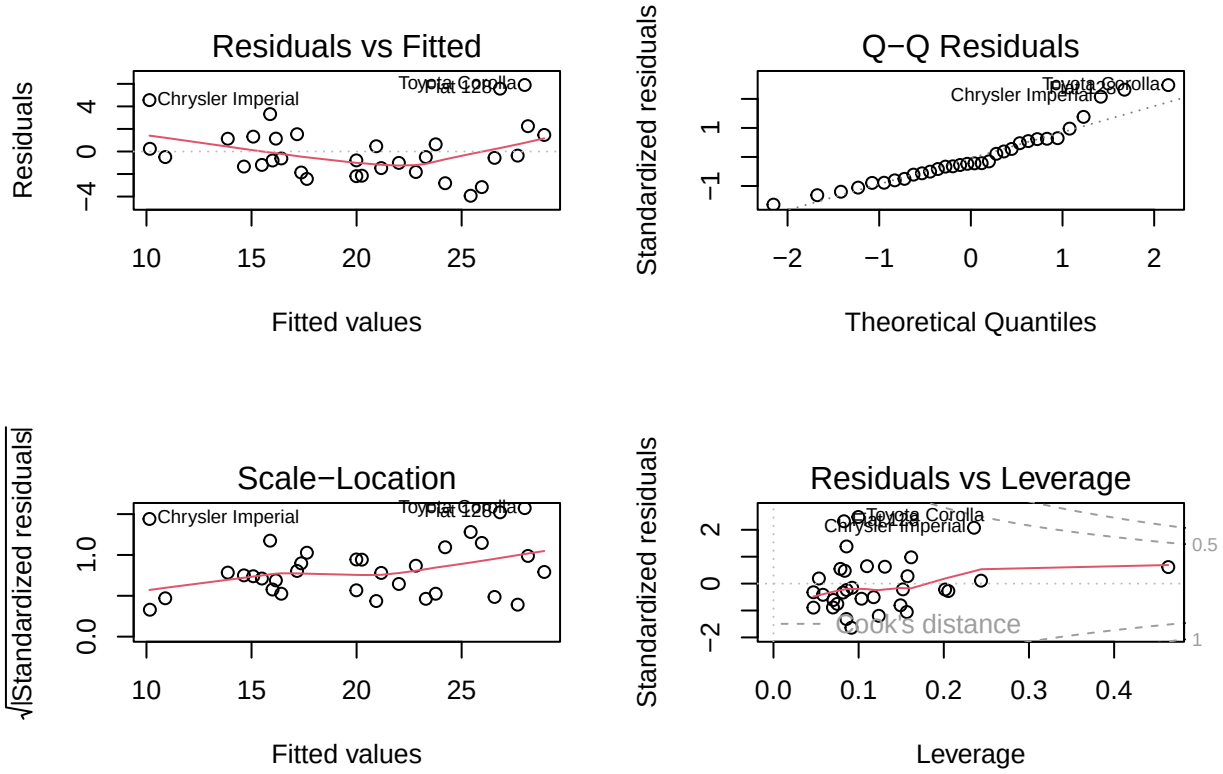
The summary provides coefficients, R-squared, and p-values for each predictor. Each coefficient indicates the effect of that predictor on mpg, holding other variables constant. Low p-values (< 0.05) suggest significant predictors.

Example 5.3: Diagonostic plots for Regression

```r
# Diagnostic plots for model2

par(mfrow = c(2, 2)) # arrange plots in 2x2 grid

plot(model2)
```



Interpretation:

The diagnostic plots help assess model assumptions. Look for linearity, homoscedasticity, and normality of residuals. Deviations may indicate issues with the model fit. What to look for:

- **Residuals vs Fitted:** Look for random scatter (no patterns).This indicates linearity and homoscedasticity.
- **Normal Q-Q:** Points should follow the diagonal line. This indicates normality of residuals.
- **Scale-Location:** Look for horizontal line with equal spread.This indicates homoscedasticity.
- **Residuals vs Leverage:** Identify influential points.This helps detect outliers.

Example 5.4: Akaike Information Criterion (AIC) for Model Comparison

AIC balances model fit and complexity. Lower AIC indicates a better model.

```
# Compare model1 and model2 using AIC

AIC(model1, model2)
```

```
##        df      AIC
## model1  3 166.0294
## model2  5 155.4766
```

Interpretation:

The model with the lower AIC value is preferred, as it indicates a better balance between model fit and complexity.

Example 5.5: Predictions

Predictions can be made using the fitted model.This is done using the `predict()` function.

```
# Predict mpg for new observations

newdata <- data.frame(wt = c(2, 3), hp = c(110, 150), cyl = c(4, 6))

predict(model2, newdata)
```

```
##        1        2
## 26.66718 20.89545
```

Interpretation:

The predicted mpg values for the new observations based on the fitted multiple regression model.Predicts expected mpg given car characteristics.

Example 5.6: Visualizing Regression lines

You can also plot both models to visually compare their fits.

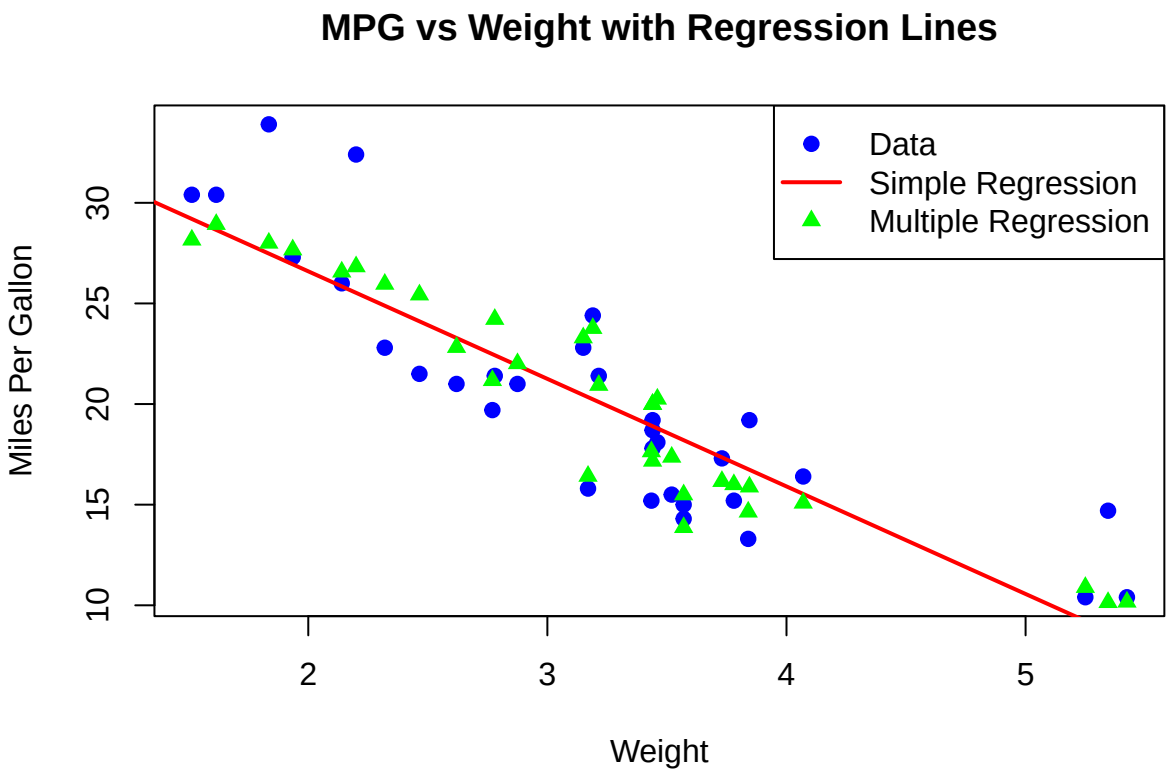
```
# Plot data and both regression lines

plot(mtcars$wt, mtcars$mpg, main = "MPG vs Weight with Regression Lines",
     xlab = "Weight", ylab = "Miles Per Gallon", pch = 19, col = "blue")

abline(model1, col = "red", lwd = 2) # Simple regression

points(mtcars$wt, predict(model2), col = "green", pch = 17) # Multiple regression predictions

legend("topright", legend = c("Data", "Simple Regression", "Multiple Regression"),
       col = c("blue", "red", "green"), pch = c(19, NA, 17), lwd = c(NA, 2, NA))
```



Example 5.7: Logistic Regression

In logistic regression, the dependent variable is binary (0/1). It models the log-odds of the outcome as a linear combination of predictors.

```
# Binary classification example
iris$IsSetosa <- ifelse(iris$Species == "setosa", 1, 0)

log_model <- glm(IsSetosa ~ Petal.Length, data = iris, family = binomial)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(log_model)
```

```
##
## Call:
## glm(formula = IsSetosa ~ Petal.Length, family = binomial, data = iris)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)     91.67   47334.35   0.002    0.998
## Petal.Length   -37.22   18357.58  -0.002    0.998
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.9095e+02  on 149  degrees of freedom
```

```
## Residual deviance: 7.3324e-09  on 148  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 25
```

Interpretation:

The summary provides coefficients and p-values. The coefficient for Petal.Length indicates how the log-odds of being Setosa change with Petal Length. A low p-value (< 0.05) suggests it is a significant predictor.

6. Hypothesis Testing

Hypothesis testing is a statistical method used to make inferences about population parameters based on sample data. It involves formulating a null hypothesis(H_o) and an alternative hypothesis (H_1), collecting data, and using statistical tests to determine whether to reject or fail to reject the null hypothesis. Hypothesis testing evaluates assumptions about population parameters using sample data.

When applicable

- To test group differences or associations.
- When validating experimental or survey claims.
- When comparing categorical frequencies.

Limitations

- Results depend on sample size.
- Violated assumptions may invalidate results.
- Misinterpretation of p-values is common.

Example 6.1: One-Way ANOVA

One way ANOVA tests for differences in means across multiple groups. For example, testing if mean weights differ across treatment groups.

```r
# Test difference in mean weights across groups

data(PlantGrowth)
# PlantGrowth dataset contains weight measurements of plants under different treatment groups.

anova_model <- aov(weight ~ group, data = PlantGrowth) # Perform one-way ANOVA

summary(anova_model) # View ANOVA table
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## group        2  3.766  1.8832   4.846 0.0159 *
## Residuals   27 10.492  0.3886
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation:

The p-value indicates whether there are significant differences in mean weights among the treatment groups. A low p-value (< 0.05) suggests at least one group mean differs significantly.

Example 6.2: Two-Way ANOVA

Two-way ANOVA tests for differences in means across two categorical factors. For example, testing if mean weights differ by treatment and time.

```
# Test difference in mean weights across groups and time

data(ToothGrowth)
# ToothGrowth dataset contains tooth length measurements of guinea pigs under different supplement type

anova_model2 <- aov(len ~ supp * dose, data = ToothGrowth) # Perform two-way ANOVA

summary(anova_model2) # View ANOVA table
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## supp         1  205.4   205.4  12.317 0.000894 ***
## dose         1 2224.3  2224.3 133.415  < 2e-16 ***
## supp:dose    1   88.9    88.9   5.333 0.024631 *
## Residuals   56  933.6    16.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation:

The p-values indicate whether there are significant differences in mean tooth lengths based on supplement type, dose, and their interaction. Low p-values (< 0.05) suggest significant effects.

Example 6.3: Post-hoc tests

If ANOVA indicates significant differences, post-hoc tests identify which groups differ.

```
# install.packages("agricolae") # Uncomment if agricolae is not installed
library(agricolae) # for post-hoc tests
# agricolae package provides functions for agricultural research, including post-hoc tests.

# Tukey's HSD post-hoc test for PlantGrowth data
tukey_result <- HSD.test(anova_model, "group", group = TRUE)

print(tukey_result)
```

```
## $statistics
##     MSerror Df  Mean       CV       MSD
##   0.3885959 27 5.073 12.28809 0.6912161
##
## $parameters
##    test name.t ntr StudentizedRange alpha
##   Tukey  group   3         3.506426  0.05
##
## $means
##      weight       std  r        se  Min  Max    Q25   Q50    Q75
## ctrl  5.032 0.5830914 10 0.1971284 4.17 6.11 4.5500 5.155 5.2925
## trt1  4.661 0.7936757 10 0.1971284 3.59 6.03 4.2075 4.550 4.8700
## trt2  5.526 0.4425733 10 0.1971284 4.92 6.31 5.2675 5.435 5.7350
##
## $comparison
## NULL
##
```

```
## $groups
##      weight groups
## trt2  5.526      a
## ctrl  5.032     ab
## trt1  4.661      b
##
## attr(,"class")
## [1] "group"
```

Interpretation:

The post-hoc test results indicate which treatment groups have significantly different mean weights. Groups sharing the same letter are not significantly different from each other.

Another example of post-hoc test using the `TukeyHSD()` function:

```
# Identify which groups differ
TukeyHSD(anova_model)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = weight ~ group, data = PlantGrowth)
##
## $group
##             diff        lwr       upr     p adj
## trt1-ctrl -0.371 -1.0622161 0.3202161 0.3908711
## trt2-ctrl  0.494 -0.1972161 1.1852161 0.1979960
## trt2-trt1  0.865  0.1737839 1.5562161 0.0120064
```

Interpretation:

The output shows pairwise comparisons between treatment groups, along with confidence intervals and p-values. Significant differences are indicated by p-values less than 0.05.

Example 6.4: Chi-Square test for Independence

Chi-square test assesses whether two categorical variables are independent. For example, testing if gender is associated with survey response (yes/no).

```
# Create contingency table
gender_response <- matrix(c(30,10,20,40), nrow=2)
rownames(gender_response) <- c("Male","Female")
colnames(gender_response) <- c("Yes","No")

# Test for independence
chisq.test(gender_response)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  gender_response
## X-squared = 15.042, df = 1, p-value = 0.0001052
```

Interpretation:

The p-value indicates whether there is a significant association between gender and survey response. A low p-value (< 0.05) suggests dependence.

Example 6.5: Wilcoxon Rank-Sum test

The Wilcoxon rank-sum test (Mann-Whitney U test) is a non-parametric test used to compare two independent groups when the data does not meet the assumptions of a t-test, such as normality. It assesses whether the distributions of the two groups differ significantly.

```
# Compare Sepal.Length between Setosa and Versicolor using Wilcoxon test

setosa <- subset(iris, Species == "setosa")$Sepal.Length

versicolor <- subset(iris, Species == "versicolor")$Sepal.Length
wilcox.test(setosa, versicolor)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  setosa and versicolor
## W = 168.5, p-value = 8.346e-14
## alternative hypothesis: true location shift is not equal to 0
```

Interpretation:

The p-value indicates whether there is a significant difference in the distributions of Sepal Length between the two species. A low p-value (< 0.05) suggests a significant difference.

Example 6.6: Kruskal-Wallis test

The Kruskal-Wallis test is a non-parametric alternative to one-way ANOVA. It is used to compare three or more independent groups when the data does not meet the assumptions of ANOVA, such as normality. It assesses whether the distributions of the groups differ significantly.

```
# Test difference in Sepal.Length across species using Kruskal-Wallis test

kruskal.test(Sepal.Length ~ Species, data = iris)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Sepal.Length by Species
## Kruskal-Wallis chi-squared = 96.937, df = 2, p-value < 2.2e-16
```

Interpretation:

The p-value indicates whether there are significant differences in the distributions of Sepal Length among the species. A low p-value (< 0.05) suggests at least one species differs significantly.

Example 6.7: Independent Two-Sample t-Test

The independent two-sample t-test is used to compare the means of two independent groups. It assesses whether the means of the two groups are significantly different from each other.

```
# Compare control vs treatment 2

control <- subset(PlantGrowth, group=="ctrl")$weight

trt2 <- subset(PlantGrowth, group=="trt2")$weight

t.test(control, trt2)
```

```
##
##  Welch Two Sample t-test
##
## data:  control and trt2
## t = -2.134, df = 16.786, p-value = 0.0479
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.98287213 -0.00512787
## sample estimates:
## mean of x mean of y
##     5.032     5.526
```

Interpretation:

The p-value indicates whether there is a significant difference in mean weights between the control and treatment 2 groups. A low p-value (< 0.05) suggests a significant difference.

7. Conclusion

Statistical data analysis in R encompasses a wide range of techniques for summarizing, visualizing, and modeling data.

Descriptive statistics provide insights into data distribution, while correlation analysis reveals relationships between variables. Inferential statistics allow for population-level conclusions based on sample data, and regression analysis models relationships for prediction and explanation. Hypothesis testing evaluates assumptions about population parameters. R's extensive libraries facilitate the practical implementation of these techniques, making it a powerful tool for statistical data analysis across various fields.