

VU LEARNING HUB

vulearninghub.xyz

HIGHLIGHTED HANDOUT

STA6420

Data Analysis by Statistical Softwares-II

STA6420 by VU Learning HUB

Prepared for VU students • Free study resources

<https://vulearninghub.xyz>

STA6420 - Data Analysis by Statistical Softwares-II

A structured study handout compiled from the course's lecture topics. Read each topic's objectives first, study the key concepts, then use the summary and exam tips for revision. 

STA6420

Data Analysis by Statistical Softwares-II

Based on 45 lecture topics

Compiled for VU students as a structured study guide

*Concepts are summarized in the writer's own words;
use alongside your lecture videos for best results.*

Contents

Unit 1: Statistical Software Foundations	3
Topic 1: The Statistical Software Landscape	3
Topic 2: SPSS: Interface and Data Management	4
Topic 3: Data Cleaning and Transformation	5
Topic 4: R and RStudio Essentials	6
Unit 2: Descriptive Analysis and Visualization	6
Topic 5: Descriptive Statistics in SPSS	6
Topic 6: Visualization Principles and SPSS Charts	7
Topic 7: Data Wrangling with dplyr	8
Topic 8: Graphics with ggplot2	9
Unit 3: Hypothesis Testing in Software	10
Topic 9: The Logic of Hypothesis Testing	10
Topic 10: t-Tests: One Sample, Independent, Paired	11
Topic 11: ANOVA: Comparing Three or More Groups	12
Topic 12: Nonparametric Tests	12
Unit 4: Regression and Multivariate Methods	13
Topic 13: Correlation and Simple Linear Regression	13
Topic 14: Multiple Regression and Diagnostics	14
Topic 15: Logistic Regression	15
Topic 16: Factor Analysis and PCA	16
Unit 5: Classification, Reliability, and Time Series	17
Topic 17: Cluster Analysis	17
Topic 18: Reliability Analysis: Cronbach's Alpha	18
Topic 19: Time Series Basics in Software	19
Topic 20: Categorical Data: Chi-Square Tests	19
Unit 6: Professional Practice and Exam Preparation 	20
Topic 21: Missing Data, Outliers, and Assumptions	20
Topic 22: Reporting and Reproducible Research	21
Topic 23: Worked Project: From Raw Data to Report	22
Topic 24: Exam Strategy and Quick-Reference	23
Glossary of Key Terms	24

Course Introduction

Modern statistics is done on computers: massive datasets, complex models, and publication-quality graphics all demand statistical software. This course - the second in the data-analysis sequence - builds professional skill with the industry's standard tools: SPSS for menu-driven analysis and R for programmable, reproducible research.

You will learn to manage real datasets (import, clean, transform), run the full analytical toolkit (descriptives, visualization, hypothesis tests, ANOVA, regression, multivariate methods), diagnose problems (outliers, missing data, violated assumptions), and report results honestly and reproducibly. Every topic pairs the statistical idea with its software implementation.

By the end you should be able to: take a raw dataset to a finished analytical report in SPSS and R, choose the right test or model for a research question, interpret output correctly (p-values, effect sizes, diagnostics), and document your work so others can reproduce it.

How to use this handout: Each topic lists its learning objectives, explains the key concepts in simple words, defines important terms, gives an example where helpful, and closes with a summary plus exam tips. Topics marked **Very important** deserve extra revision.

Unit 1: Statistical Software Foundations

Topic 1: The Statistical Software Landscape

Learning Objectives

- Compare SPSS, R, SAS, Python, Stata
- Choose the right tool for a task
- Explain reproducibility

Key Concepts

The analyst's toolbox: SPSS - menu-driven, dominant in social sciences and Pakistani universities, excellent for standard analyses, but expensive and less flexible. R - free, open-source, the research standard; everything is code, so analyses are reproducible; 20,000+ packages cover every method. SAS - enterprise standard in pharma and banking, powerful but costly. Python (pandas, statsmodels, scikit-learn) - the data-science favorite, strong in machine learning. Stata - popular in economics, clean syntax.

This course uses SPSS and R because they complement each other: SPSS teaches the logic of analysis through dialogs (you learn what each option means), while R teaches automation and reproducibility (you script the same analysis in ten lines). Employers value both: SPSS for quick business reporting, R for serious research.

Reproducibility - the principle that another person can re-run your analysis and get identical results - is the course's backbone. Menu clicks are not reproducible unless logged; SPSS syntax and R scripts are. Every analysis you submit should be a script, not a story about clicks.

Key Points

- SPSS: menu-driven, standard in social sciences; R: free, scripted, research standard.
- Learn both: SPSS for logic, R for reproducibility.

- Every analysis must be a re-runnable script.

Key Terms

Reproducibility: Ability to re-run an analysis and get identical results.

Syntax: SPSS's command language logging every action.

Package: An R add-on library of functions.

Example

- *An R script that imports, cleans, analyzes, and plots - re-runnable in one click.*

Summary: Two tools, one discipline: correct, reproducible data analysis.

Exam Tips

- ★ Key point: SPSS vs. R strengths.
- ★ Remember the definition of reproducibility.

Topic 2: SPSS: Interface and Data Management

Learning Objectives

- Navigate the SPSS environment
- Define variables properly
- Enter and import data

Key Concepts

SPSS has three key windows: the Data Editor (spreadsheet of cases by variables, with Data View and Variable View tabs), the Output Viewer (tables and charts of results), and the Syntax Editor (where commands live). Professional work happens mostly in the Syntax Editor - paste every dialog's syntax (the Paste button) so your analysis is documented.

Variable View is where data quality begins: set the Name (short, no spaces), Type (numeric, string, date), Label (full description), Values (e.g., 1 = 'Male', 2 = 'Female'), Missing (codes like 99), and Measure (nominal, ordinal, scale). Getting Measure right matters - SPSS uses it to decide which analyses and charts are offered. A Likert scale coded 1-5 as nominal will confuse every dialog.

Importing: File > Open > Data reads Excel, CSV, and databases; the Text Import Wizard handles messy CSVs. After import, always run descriptives first (Analyze > Descriptive Statistics > Frequencies) to catch import errors - a variable that should range 1-5 showing 55 means a data-entry slip. Save data as .sav, syntax as .sps, output as .spv: three files, one project.

Key Points

- Three windows: Data Editor, Output Viewer, Syntax Editor.
- Variable View: name, type, labels, values, missing, measure.
- Always paste dialog syntax; always screen imported data.

Key Terms

Variable View: Metadata tab defining each variable.

Measure: Nominal/ordinal/scale level of a variable.

Value labels: Text meanings of numeric codes.

Example

- *Frequencies on imported age data reveals a 555 typo instantly.*

Summary: Clean metadata and syntax logging are the foundation of trustworthy SPSS work.

Exam Tips

- ★ Very important: the three windows and their roles.
- ★ Remember why Measure level matters.

Topic 3: Data Cleaning and Transformation

Learning Objectives

- Detect and handle dirty data
- Recode and compute variables
- Treat missing values

Key Concepts

Real data is dirty: typos (age 555), inconsistent codes ('M', 'm', 'Male'), impossible combinations (pregnant males), duplicates, and missing values. Cleaning workflow: (1) Frequencies/Descriptives on every variable - scan minimums, maximums, and category lists; (2) cross-tabulate logically linked variables; (3) sort and eyeball suspicious cases; (4) document every change in syntax comments - never edit cells silently.

Transformation: Transform > Compute Variable creates new variables (e.g., BMI = weight / height²; total score = SUM(q1 to q10)). Transform > Recode into Different Variables collapses categories (age into age-groups) or reverses Likert items - always recode into a NEW variable, preserving the original. Transform > Rank Cases, Count Occurrences, and Automatic Recode handle special needs. Date arithmetic (DATEDIFF) computes durations.

Missing values: distinguish system-missing (blank) from user-missing (coded 99). Listwise deletion (drop incomplete cases) is the default but wastes data and can bias results; pairwise uses available data per analysis. Better: multiple imputation (Analyze > Multiple Imputation) creates several plausible completed datasets and pools results - the modern standard. At minimum, report how much is missing and test whether missingness looks random.

Key Points

- Clean via frequencies, cross-tabs, and documented syntax - never silent edits.
- Compute and Recode (into NEW variables) build analysis-ready measures.
- Handle missingness explicitly; multiple imputation beats deletion.

Key Terms

Listwise deletion: Dropping cases with any missing value.

Multiple imputation: Creating several plausible completed datasets.

System-missing: SPSS blank for truly absent values.

Example

- *Recoding age into 18-25/26-40/41+ groups for a cross-tab.*

Summary: Analysis is only as honest as the cleaning behind it.

Exam Tips

- ★ Key point: recode into different variables, not the same.

- ★ Remember listwise vs. multiple imputation. 

Topic 4: R and RStudio Essentials

Learning Objectives

- Navigate RStudio
- Use core R data structures
- Install and load packages

Key Concepts

RStudio's four panes: Source (scripts), Console (interactive execution), Environment/History (objects in memory), and Files/Plots/Packages/Help. Work in scripts (.R files), never by typing only in the console - scripts are your reproducible record. Projects (.Rproj) keep working directories sane.

Core structures: vectors (`c(1,2,3)` - all elements one type), factors (categorical variables with levels - crucial for models), matrices, data frames (the workhorse: columns of possibly different types, like a spreadsheet), and lists. Key verbs: `read.csv()` / `readxl::read_excel()` to import; `head()`, `str()`, `summary()` to inspect; `df$var` to access columns; `df[df$page > 30, ]` to subset rows.

Packages extend R: `install.packages("tidyverse")` once, then `library(tidyverse)` each session. The tidyverse (`dplyr`, `ggplot2`, `tidyr`, `readr`) gives a consistent grammar for data science and is this course's R dialect. Help: `?function`, and vignettes. R's error messages look cryptic at first - read them slowly; they usually name the exact problem (e.g., 'object not found' means a typo or unloaded data).

Key Points

- RStudio panes: Source, Console, Environment, Files/Plots/Help.
- Data frames are the workhorse; factors encode categories.
- Script everything; install once, `library()` each session.

Key Terms

Data frame: R's spreadsheet-like structure.

Factor: Categorical variable with defined levels.

Tidyverse: Coherent collection of R data-science packages.

Example

- *`summary(df)` instantly profiles every column of an imported dataset.*

Summary: R rewards script-based work with total reproducibility.

Exam Tips

- ★ Key point: the four RStudio panes.
- ★ Remember vectors vs. data frames vs. factors.

Unit 2: Descriptive Analysis and Visualization

Topic 5: Descriptive Statistics in SPSS

Learning Objectives

- Run Frequencies and Descriptives

- Interpret central tendency and dispersion
- Detect outliers

Key Concepts

Analyze > Descriptive Statistics > Frequencies gives counts, percents, and (optionally) mean, median, mode, standard deviation, min, max, quartiles - ideal for categorical variables and first looks. Descriptives gives z-scores (standardized values) and is faster for many scale variables. Explore (Analyze > Descriptive Statistics > Explore) is the analyst's favorite: descriptives by group, stem-and-leaf, boxplots, and normality tests (Shapiro-Wilk, Kolmogorov-Smirnov) with Q-Q plots - one dialog for serious screening.

Interpretation: mean vs. median signals skew (income: mean > median, right-skewed); standard deviation measures spread in the variable's units; quartiles and IQR resist outliers. Outliers: boxplots flag points beyond $1.5 \times \text{IQR}$; z-scores beyond ± 3 deserve inspection. Decide per case: correct (typo), keep (genuine extreme), or analyze robustly (medians, trimmed means) - and report the decision.

Always split by meaningful groups (Explore's Factor List): overall means can hide group differences (Simpson's paradox). A drug's overall recovery rate may look flat while improving in every subgroup - aggregation lies, stratification reveals.

Key Points

- Frequencies for categories; Descriptives/Explore for scale variables.
- Mean vs. median reveals skew; IQR and boxplots handle outliers.
- Stratify by groups - aggregates can mislead (Simpson's paradox).

Key Terms

Skew: Asymmetry of a distribution.

IQR: Interquartile range: $Q3 - Q1$.

Simpson's paradox: Trend reversals between aggregated and stratified data.

Example

- *Income data: mean Rs. 85.000 but median Rs. 45.000 - right-skewed by high earners.*

Summary: Descriptives are not preliminary - they are the analysis's immune system.

Exam Tips

- ★ Very important: mean vs. median under skew.
- ★ Remember the $1.5 \times \text{IQR}$ outlier rule.

Topic 6: Visualization Principles and SPSS Charts

Learning Objectives

- Choose the right chart type
- Build charts in SPSS
- Avoid misleading graphics

Key Concepts

Chart grammar: bar charts for categorical comparisons, histograms for distributions of scale variables, boxplots for group comparisons of distributions, scatterplots for bivariate relationships, line charts for trends over time. One variable, one message - a chart trying to say three things says nothing.

SPSS Chart Builder (Graphs > Chart Builder): drag variables onto the canvas, choose elements. Histograms with normal curves overlay; clustered bar charts for two categorical variables; boxplots by group; scatterplots with fit lines. Edit in the Chart Editor: label axes fully ('Monthly income (Rs. thousands)', not 'income'), start bars at zero (truncated axes exaggerate), and keep 3D effects away - they distort perception.

Honesty rules: the area/length must be proportional to the numbers (no pictographs with stretched icons); show the data, not just the summary (jittered points over boxplots); annotate sample sizes; and never cherry-pick axis ranges to manufacture drama. A good chart lets the reader verify your claim at a glance - that verifiability is the point.

Key Points

- Match chart to variable types: bars, histograms, boxplots, scatterplots.
- Label fully, start bars at zero, avoid 3D distortion.
- Honest charts are proportional, annotated, and verifiable.

Key Terms

Chart Builder: SPSS drag-and-drop chart interface.

Truncated axis: Axis not starting at zero - exaggerates differences.

Example

- *Boxplots of test scores by teaching method reveal medians, spread, and outliers together.*

Summary: Charts are arguments - make them honest and legible.

Exam Tips

- ★ Key point: chart-type selection rules.
- ★ Remember why bars must start at zero.

Topic 7: Data Wrangling with dplyr

Learning Objectives

- Use the five dplyr verbs
- Chain operations with pipes
- Reshape data with tidyr

Key Concepts

dplyr's grammar: `filter()` keeps rows (`df %>% filter(age > 30)`), `select()` keeps columns, `mutate()` creates columns (`mutate(bmi = weight/height^2)`), `arrange()` sorts, `summarise()` collapses to statistics - usually after `group_by()`. The pipe `%>%` (or native `|>`) feeds the left result into the right function, turning nested calls into readable pipelines: raw data in, analysis-ready data out, every step visible.

Grouped summaries are the workhorse: `df %>% group_by(gender) %>% summarise(mean_income = mean(income, na.rm = TRUE), n = n())`. Joins combine tables: `left_join()`, `inner_join()` by key columns - essential when survey responses live in one file and demographics in another.

tidyr reshapes: `pivot_longer()` turns wide format (one column per year) into long (year as a variable) - the format ggplot2 and most models want; `pivot_wider()` reverses. `separate()/unite()` split and combine columns. Real datasets arrive misshapen; wrangling is not janitorial work, it is analysis - decisions made here (how to code missingness, which rows count) determine every result downstream.

Key Points

- Five verbs: filter, select, mutate, arrange, summarise (+ group_by).
- Pipes make multi-step wrangling readable and reproducible.
- pivot_longer/wider reshape between wide and long formats.

Key Terms

Pipe: %>% passing data through a chain of verbs.

Long format: One row per observation-unit, variables in columns.

Join: Merging tables on key columns.

Example

- `df %>% group_by(dept) %>% summarise(ava = mean(salary, na.rm=TRUE))`

Summary: dplyr turns messy data into analysis-ready data, transparently.

Exam Tips

- ★ Very important: name the five verbs.
- ★ Remember long vs. wide format.

Topic 8: Graphics with ggplot2

Learning Objectives

- Explain the grammar of graphics
- Build layered plots
- Customize publication-quality figures

Key Concepts

ggplot2 implements the grammar of graphics: a plot = data + aesthetics (mappings of variables to x, y, color, size) + geometries (points, bars, lines) + optional facets/scales/themes. `ggplot(df, aes(x = income, y = spending)) + geom_point() + geom_smooth(method = "lm")` builds a scatterplot with a trend line - each layer added with +.

Facets (`facet_wrap(~ gender)`) split one plot into panels by group - the graphical equivalent of stratification, and far more revealing than rainbow-colored overplotting. Scales control axes and legends (`scale_x_log10()`, `labs()` for titles and axis labels); themes (`theme_minimal()`) clean up chartjunk. Save with `ggsave()` at print resolution.

The grammar prevents common sins: mappings are explicit (no mystery axes), and layering encourages showing data plus summary (jittered points over boxplots: `geom_boxplot() + geom_jitter()`). Publication quality means: readable fonts, labeled axes with units, colorblind-safe palettes (viridis), and a caption stating the sample. A ggplot script regenerates the exact figure forever - reproducibility extends to graphics.

Key Points

- Grammar: data + aesthetics + geometries + facets/scales/themes.
- Facets stratify visually; layers combine raw data with summaries.
- `ggsave()` preserves exact, publication-ready figures.

Key Terms

Aesthetics: Mappings from variables to visual properties.

Geometries: The marks drawn: points, bars, lines.

Facets: Small-multiple panels by group.

Example

```
• aaplot(df, aes(x=studv_hrs, v=marks)) + geom_point() + facet_wrap(~gender)
```

Summary: The grammar of graphics makes plots systematic, honest, and reproducible.

Exam Tips

- ★ Key point: the grammar components.
- ★ Remember facets as visual stratification.

Unit 3: Hypothesis Testing in Software

Topic 9: The Logic of Hypothesis Testing

Learning Objectives

- State null and alternative hypotheses
- Interpret p-values correctly
- Distinguish significance from importance

Key Concepts

Every test follows the same logic: assume the null hypothesis H_0 (no effect, no difference), compute how surprising the data are under H_0 (the p-value = probability of data this extreme or more, if H_0 were true), and reject H_0 if $p < \alpha$ (conventionally 0.05). A small p-value says 'these data are surprising under H_0 ' - it is NOT the probability that H_0 is true, nor the probability the result is a fluke.

Errors: Type I (rejecting a true H_0 - false alarm, probability α) and Type II (missing a real effect - probability β ; power = $1 - \beta$). Tests assume: random sampling, appropriate measurement level, and distributional conditions (normality for t-tests/ANOVA - checked via Explore, Shapiro-Wilk, Q-Q plots). Violations do not always invalidate, but they must be checked and reported.

Crucially, statistical significance is not practical importance: with $n = 10,000$, a Rs. 50 income difference is 'significant' but meaningless. Always report effect sizes (Cohen's d , eta-squared, r) and confidence intervals alongside p-values. The software gives p-values cheaply; judgment about importance remains yours.

Key Points

- p-value = $P(\text{data this extreme} \mid H_0 \text{ true})$; reject if $p < \alpha$.
- Type I = false alarm; Type II = missed effect; power = $1 - \beta$.
- Report effect sizes and CIs - significance is not importance.

Key Terms

p-value: Surprise measure of data under H_0 .

Effect size: Magnitude of the difference/relationship.

Power: Probability of detecting a real effect.

Example

- $p = 0.03$ with $d = 0.1$: statistically significant. practically trivial.

Summary: Tests quantify surprise; effect sizes quantify importance - report both.

Exam Tips

- ★ Very important: correct interpretation of the p-value.
- ★ Remember Type I vs. Type II errors.

Topic 10: t-Tests: One Sample, Independent, Paired

Learning Objectives

- Choose the right t-test
- Run and interpret t-tests in SPSS
- Check assumptions

Key Concepts

One-sample t-test: is the mean different from a hypothesized value? (Analyze > Compare Means > One-Sample T Test.) Example: is average delivery time different from the promised 30 minutes? Independent-samples t-test: do two separate groups differ? (Analyze > Compare Means > Independent-Samples T Test.) First read Levene's test for equal variances - use the 'equal variances not assumed' (Welch) row if significant. Paired-samples t-test: same units measured twice (before/after) - (Paired-Samples T Test); pairing removes between-unit noise, greatly increasing power.

Assumptions: independent observations, roughly normal sampling distribution of the mean ($n \geq 30$ per group is usually safe via CLT; smaller n needs Shapiro-Wilk on the data or differences), and for independent t, roughly equal variances (or use Welch's). SPSS output: means, mean difference, t, df, p-value, and - request it - the 95% CI of the difference and Cohen's d (Options/Effect sizes).

Worked logic: 40 students, method A mean 72 (SD 8), method B mean 68 (SD 9): t-test $p = 0.04$, $d = 0.47$ (medium). Conclusion: method A scores higher by about 4 marks (95% CI: 0.2 to 7.8), a medium effect - statistically significant and plausibly meaningful. That sentence - estimate, interval, effect size, judgment - is the template for reporting every test.

Key Points

- One-sample: vs. a value; independent: two groups (check Levene's); paired: same units twice.
- Assumptions: independence, approximate normality, equal variances (or Welch).
- Report: mean difference, CI, p-value, effect size, judgment.

Key Terms

Levene's test: Test of equal variances across groups.

Welch's t: t-test not assuming equal variances.

Cohen's d: Standardized mean difference.

Example

- Paired test on 25 patients' blood pressure before/after treatment.

Summary: Three t-tests cover most two-group questions; report completely.

Exam Tips

- ★ Very important: which t-test for which design.

★ Remember Levene's test and Welch's fallback. 

Topic 11: ANOVA: Comparing Three or More Groups

Learning Objectives

- Explain ANOVA logic
- Run one-way ANOVA in SPSS
- Conduct post-hoc comparisons

Key Concepts

ANOVA asks whether three or more group means differ, by comparing between-group variation to within-group variation: $F = (\text{variation between groups}) / (\text{variation within groups})$. A large F (small p) says the group means differ more than chance explains - but not which groups differ. Running three t-tests instead inflates the Type I error rate (14% for three tests at $\alpha = 0.05$); ANOVA protects the overall error rate.

SPSS: Analyze > Compare Means > One-Way ANOVA; move the scale variable to Dependent List, the grouping variable to Factor; Options: descriptives, homogeneity of variance test; Post Hoc: Tukey (equal variances) or Games-Howell (unequal). Output: descriptives table, Levene's test, the ANOVA table (F , p), and post-hoc pairwise comparisons with adjusted p -values.

Example: exam scores under three teaching methods, $F(2,87) = 6.2$, $p = 0.003$, eta-squared = 0.12. Tukey: method A vs. C differs ($p = 0.002$), others not. Report: 'Scores differed by method, $F(2,87) = 6.2$, $p = .003$, eta-squared = .12; post-hoc Tukey showed A exceeded C ($p = .002$).' Two-way ANOVA adds a second factor and their interaction - e.g., method by gender, where the interaction asks whether the method effect differs for men and women.

Key Points

- ANOVA: $F = \text{between-group} / \text{within-group variation}$; one test protects error rate.
- Significant F needs post-hoc (Tukey/Games-Howell) to locate differences.
- Report F , df , p , eta-squared, and post-hoc results.

Key Terms

F-statistic: Ratio of between- to within-group variance.

Post-hoc test: Pairwise comparisons after significant ANOVA.

Eta-squared: Proportion of variance explained by groups.

Example

- *Three fertilizers' yields: ANOVA $p = 0.003$. Tukey locates the winning pair.*

Summary: ANOVA first asks 'any difference?', post-hocs ask 'where?'

Exam Tips

- ★ Very important: why not multiple t-tests.
- ★ Remember the ANOVA reporting template.

Topic 12: Nonparametric Tests

Learning Objectives

- Recognize when parametric tests fail

- Run Mann-Whitney, Wilcoxon, Kruskal-Wallis
- Interpret rank-based results

Key Concepts

When normality fails badly (small n , heavy skew, ordinal data like Likert scales), rank-based nonparametric tests replace t-tests and ANOVA. They test differences in distributions/medians rather than means and need no normality - at the cost of slightly less power when normality holds.

SPSS: Analyze > Nonparametric Tests > Independent Samples (Mann-Whitney U replaces independent t-test; Kruskal-Wallis replaces one-way ANOVA) and Related Samples (Wilcoxon signed-rank replaces paired t-test; Friedman replaces repeated-measures ANOVA). The dialogs auto-select based on your data structure; the output gives test statistics, p-values, and pairwise comparisons.

Example: customer satisfaction (1-5 ordinal) for two branches, $n = 15$ each, skewed: Mann-Whitney $U = 62$, $p = 0.03$ - branches differ in satisfaction rankings. Report medians per group (not means - the test is rank-based) with the U statistic and p-value. In R: `wilcox.test()`, `kruskal.test()`. Rule of thumb: with $n \geq 30$ per group, parametric tests are robust enough; with small or ordinal data, go nonparametric - and say why in your report.

Key Points

- Nonparametrics use ranks; no normality needed; for small/ordinal/skewed data.
- Mann-Whitney ~ t-test, Wilcoxon ~ paired t, Kruskal-Wallis ~ ANOVA.
- Report medians with rank-based tests.

Key Terms

Mann-Whitney U: Rank test for two independent groups.

Kruskal-Wallis: Rank test for 3+ independent groups.

Ordinal data: Ordered categories without equal intervals.

Example

- *Likert satisfaction by branch: Mann-Whitney $p = 0.03$ on medians.*

Summary: Ranks rescue analysis when normality cannot be assumed.

Exam Tips

- ★ Key point: match each nonparametric test to its parametric twin.
- ★ Remember to report medians, not means.

Unit 4: Regression and Multivariate Methods

Topic 13: Correlation and Simple Linear Regression

Learning Objectives

- Interpret correlation coefficients
- Fit and interpret simple regression
- Use scatterplots diagnostically

Key Concepts

Correlation (Analyze > Correlate > Bivariate) measures linear association: r in $[-1, 1]$, sign = direction, magnitude = strength ($|r| > 0.7$ strong, 0.3-0.7 moderate, < 0.3 weak - rough guides). Always plot first: $r = 0.8$ with a curved pattern means the linear summary misleads; outliers inflate or deflate r dramatically. Correlation is not causation - ice-cream sales correlate with drowning (both rise in summer).

Simple linear regression predicts Y from X : $Y = a + bX$. SPSS: Analyze > Regression > Linear. Output: Model Summary (R -squared = proportion of Y 's variance explained), ANOVA table (is the model better than just the mean?), Coefficients (intercept a , slope b with p -values and CIs). Interpretation template: 'Each extra study hour predicts 2.3 more marks ($b = 2.3$, 95% CI: 1.8-2.8, $p < 0.001$); the model explains 61% of mark variation (R -squared = 0.61).'

Assumptions live in the residuals (observed minus predicted): linearity (residuals vs. predicted shows no curve), constant variance (no funnel shape), normality (histogram/Q-Q of residuals), independence. SPSS: Plots > ZRESID vs ZPRED, histogram. Violations have fixes: transform Y (log for skew), add polynomial terms for curves, or use robust methods.

Key Points

- r measures linear association; plot before trusting it.
- Regression: $Y = a + bX$; report slope, CI, p , and R -squared.
- Check residuals for linearity, constant variance, normality.

Key Terms

R-squared: Proportion of variance explained by the model.

Residual: Observed minus predicted value.

Slope: Predicted change in Y per unit of X .

Example

- Study hours $\rightarrow$ marks: $b = 2.3$. R -squared = 0.61.

Summary: Regression quantifies prediction; residuals verify the model's honesty.

Exam Tips

- ★ Very important: the regression reporting template.
- ★ Remember correlation is not causation.

Topic 14: Multiple Regression and Diagnostics

Learning Objectives

- Fit multiple regression
- Diagnose multicollinearity
- Select variables wisely

Key Concepts

Multiple regression predicts Y from several X 's: $\text{marks} = a + b_1 \cdot \text{hours} + b_2 \cdot \text{attendance} + b_3 \cdot \text{prior_gpa}$. Each coefficient is the effect holding others constant - the model's great strength. SPSS: same Linear dialog, multiple independents; Statistics: collinearity diagnostics, Durbin-Watson, residual plots.

Multicollinearity - predictors correlating with each other - inflates standard errors and makes coefficients unstable (sign flips between samples). Diagnose: VIF (variance inflation factor) > 10 (or > 5 , strict) signals trouble; correlation matrix of predictors. Fixes: drop redundant predictors, combine them (an

index), or use ridge regression. R-squared always rises when adding predictors - use adjusted R-squared (penalizes complexity) to compare models.

Influence: some points drag the regression line. Cook's distance > 1 (or $> 4/n$) flags influential cases; leverage flags unusual X-combinations. Investigate flagged cases - a data-entry error (income 5,500,000) must be fixed; a genuine extreme case may merit robust regression or reporting with/without it. Dummy variables encode categories: k categories need k-1 dummies (e.g., dept dummies for Science/Arts/Commerce); the omitted category is the reference.

Key Points

- Coefficients = effects holding other predictors constant.
- VIF > 10 signals multicollinearity; use adjusted R-squared for comparison.
- Check influence (Cook's distance); use k-1 dummies for categories.

Key Terms

Multicollinearity: Predictors correlating among themselves.

VIF: Variance inflation factor; >10 is problematic.

Dummy variable: 0/1 encoding of a category.

Example

- *Hours and attendance correlate $r=0.85$: VIF 4.5 - watch. but tolerable.*

Summary: Multiple regression is powerful; diagnostics keep it honest.

Exam Tips

- ★ Very important: VIF rule and its fix.
- ★ Remember adjusted R-squared vs. R-squared.

Topic 15: Logistic Regression

Learning Objectives

- Explain when logistic regression applies
- Interpret odds ratios
- Assess model fit

Key Concepts

When Y is binary (pass/fail, buy/not-buy, diseased/healthy), linear regression breaks (predictions outside $[0,1]$, violated assumptions). Logistic regression models the log-odds: $\ln(p/(1-p)) = a + b_1 \cdot X_1 + b_2 \cdot X_2$. SPSS: Analyze > Regression > Binary Logistic. The S-shaped logistic curve keeps predictions between 0 and 1.

Interpretation via odds ratios: $OR = e^b$. If $b = 0.7$ for study hours, $OR = 2.0$ - each extra hour doubles the odds of passing. $OR > 1$ increases odds, $OR < 1$ decreases. Report: 'Each additional study hour doubled the odds of passing ($OR = 2.0$, 95% CI: 1.4-2.9, $p = 0.001$).' The Classification Table shows hit rate (e.g., 82% correctly classified); pseudo R-squareds (Cox & Snell, Nagelkerke) approximate explained variation; Hosmer-Lemeshow tests fit (want $p > 0.05$ - non-significance means adequate fit).

Assumptions: binary outcome, independent observations, linearity in the logit (check via Box-Tidwell), no severe multicollinearity, adequate sample (10+ events per predictor - small samples with rare outcomes need exact methods or penalization). In R: `glm(y ~ x1 + x2, family = binomial)`. Logistic

regression is the gateway to the whole generalized-linear-model family.

Key Points

- Binary $Y \rightarrow$ logistic: $\ln(p/(1-p)) = a + bX$; predictions stay in $[0,1]$.
- Odds ratio e^b : multiplicative effect on odds per unit of X .
- Check classification rate, pseudo R-squared, Hosmer-Lemeshow.

Key Terms

Odds ratio: Multiplicative change in odds per unit of X .

Logit: Log-odds, the modelled quantity.

Hosmer-Lemeshow: Goodness-of-fit test for logistic models.

Example

- *OR = 2.0 for study hours: each hour doubles pass odds.*

Summary: Logistic regression brings regression's logic to yes/no outcomes.

Exam Tips

- ★ Very important: interpret an odds ratio.
- ★ Remember why linear regression fails for binary Y .

Topic 16: Factor Analysis and PCA

Learning Objectives

- Explain dimensionality reduction
- Run PCA/factor analysis in SPSS
- Decide the number of factors

Key Concepts

Surveys produce dozens of correlated items; factor analysis finds the few latent dimensions behind them (20 satisfaction questions $\rightarrow$ 'service quality', 'price fairness', 'ambiance'). Principal Components Analysis (PCA) is the common starting point: it builds uncorrelated components capturing maximum variance.

SPSS: Analyze > Dimension Reduction > Factor. Descriptives: KMO and Bartlett's test (KMO > 0.6 and Bartlett $p < 0.05$ mean factor analysis is appropriate). Extraction: principal components; decide the number via eigenvalues > 1 (Kaiser), the scree plot elbow, and - best - interpretability. Rotation: Varimax (orthogonal) clarifies which items load on which factor. The Rotated Component Matrix shows loadings (correlations of items with factors); loadings above ~0.5 define the factor's meaning.

Example: 12 course-evaluation items $\rightarrow$ 2 components: 'teaching quality' (items on clarity, feedback, enthusiasm) and 'course design' (materials, workload fairness, assessment clarity), together explaining 68% of variance. Save factor scores (Scores > Save as variables) for use as predictors in regression - a clean way to handle multicollinear survey items. Caution: factors are interpretations, not discoveries - name them from the loadings, and distrust solutions that contradict theory without strong evidence.

Key Points

- Factor analysis/PCA reduces many correlated items to few dimensions.
- Check KMO/Bartlett; choose factors via eigenvalues, scree, interpretability.

- Rotate (Varimax); interpret loadings; optionally save factor scores.

Key Terms

Latent factor: Unobserved dimension behind correlated items.

Loading: Item-factor correlation.

Scree plot: Eigenvalue plot for choosing factor count.

Example

- 12 evaluation items -> 'teaching quality' and 'course design' factors.

Summary: Dimensionality reduction turns item sprawl into interpretable structure.

Exam Tips

- ★ Key point: KMO/Bartlett as gatekeepers.
- ★ Remember how factors get their names (loadings).

Unit 5: Classification, Reliability, and Time Series

Topic 17: Cluster Analysis

Learning Objectives

- Explain cluster analysis
- Run k-means in SPSS
- Validate clusters

Key Concepts

Cluster analysis groups cases by similarity - finding customer segments, grouping regions by development indicators, typing survey respondents. Unlike classification (which predicts known labels), clustering discovers the groups. SPSS offers: K-Means (Analyze > Classify > K-Means Cluster) - you specify k, it assigns cases to minimize within-cluster distance; Hierarchical (plots a dendrogram, letting you choose k by cutting the tree); TwoStep (handles large data and mixed variable types, auto-selects k).

K-means workflow: standardize variables first (z-scores - otherwise income in rupees dominates age in years), choose k (try several; use the elbow in within-cluster sum of squares or silhouette widths), run, then profile clusters via means: 'Cluster 1: young, high-spending urbanites (n=320)'. Validate: split-half replication (do clusters reappear?), and - most convincing - clusters should differ on variables NOT used to form them.

In R: kmeans(), hclust(), factoextra for visualization. Honest reporting: clustering always returns clusters, even from random noise - so validation and substantive interpretation are mandatory, not optional. A cluster solution nobody can interpret is a failed analysis, whatever the statistics say.

Key Points

- Clustering discovers groups; standardize before distance-based methods.
- K-means needs chosen k; hierarchical shows the merge tree.
- Validate: replication, external variables, interpretability.

Key Terms

K-means: Partitioning into k clusters minimizing within-distance.

Dendrogram: Tree diagram of hierarchical merges.

Standardization: Z-scoring so scales don't dominate distance.

Example

- Customers -> 'bargain hunters'. 'loyal regulars'. 'premium seekers'.

Summary: Clustering finds structure; validation proves it is real.

Exam Tips

- ★ Key point: why standardize before k-means.
- ★ Remember clustering discovers - it doesn't confirm.

Topic 18: Reliability Analysis: Cronbach's Alpha

Learning Objectives

- Explain reliability
- Compute Cronbach's alpha
- Improve a scale

Key Concepts

A questionnaire scale (10 items measuring 'statistics anxiety') is useful only if its items consistently measure the same thing - reliability. Cronbach's alpha estimates internal consistency: $\alpha = (k/(k-1)) * (1 - \sum(\text{item variances})/\text{total variance})$, ranging 0-1. Guides: > 0.9 excellent, 0.8-0.9 good, 0.7-0.8 acceptable for research, below 0.6 questionable.

SPSS: Analyze > Scale > Reliability Analysis; move items in; check 'Scale if item deleted' - it shows alpha if each item were dropped, identifying bad items (dropping a confusing or reversed-unscored item raises alpha). Reverse-scored items must be recoded first (a 'I enjoy statistics' item in an anxiety scale) - forgetting this is the classic alpha-killer.

Example: 10-item anxiety scale, $\alpha = 0.68$; 'item 4 if deleted' $\alpha = 0.81$ - item 4 is faulty (bad wording); dropping it yields an acceptable 0.81. Report: 'The 9-item scale showed good internal consistency ($\alpha = 0.81$).' In R: `psych::alpha()`. Note: alpha grows with item count, so long scales look reliable automatically - and alpha assumes unidimensionality; a scale mixing two constructs needs factor analysis first, then alpha per factor.

Key Points

- Alpha measures internal consistency; > 0.7 acceptable, > 0.8 good.
- Use 'alpha if item deleted' to find bad items; recode reversed items first.
- Alpha assumes one dimension - factor-analyze mixed scales first.

Key Terms

Cronbach's alpha: Internal-consistency reliability coefficient.

Reverse-scored item: Item worded opposite to the scale direction.

Example

- Dropping faulty item 4 raises alpha from 0.68 to 0.81.

Summary: Reliability analysis certifies that a scale measures one thing consistently.

Exam Tips

- ★ Very important: interpret alpha values. ☰
- ★ Remember to recode reversed items before analysis.

Topic 19: Time Series Basics in Software

Learning Objectives

- Plot and decompose time series
- Explain autocorrelation
- Fit simple forecasting models

Key Concepts

Time series data (monthly sales, daily temperatures) have order - and that order matters. First plot: Graphs > Legacy Dialogs > Line, or in R ggplot with `geom_line`. Look for trend (long-run direction), seasonality (yearly cycles), and irregular noise. Decomposition (classical additive: $\text{series} = \text{trend} + \text{seasonal} + \text{residual}$) separates these; SPSS: Analyze > Forecasting > Seasonal Decomposition.

Autocorrelation - correlation of the series with its own past - is the key diagnostic: ACF/PACF plots (Analyze > Forecasting > Autocorrelations) show which lags matter. Sales correlating with last month's sales (lag-1 ACF = 0.7) means naive models ignoring history will fail. Forecasting: exponential smoothing (weights recent observations more) and ARIMA (autoregressive integrated moving average) are the workhorses; SPSS's Expert Modeler (Analyze > Forecasting > Create Models) auto-selects.

Example: 5 years of monthly ice-cream sales show strong yearly seasonality and upward trend; Holt-Winters exponential smoothing captures both, forecasting next summer's peak within 8%. Caution: forecasting assumes the future resembles the past - structural breaks (a pandemic, a new competitor) break models; always plot forecasts against holdout data (fit on 4 years, test on the 5th) before trusting them.

Key Points

- Plot first: trend, seasonality, noise; decompose to separate them.
- ACF/PACF reveal dependence on the past.
- Validate forecasts on holdout data; beware structural breaks.

Key Terms

Autocorrelation: Correlation of a series with its own lags.

Seasonality: Regular cycles (e.g., yearly).

Holdout validation: Testing forecasts on unseen later data.

Example

- *Ice-cream sales: Holt-Winters captures trend + yearly seasonality.*

Summary: Time series analysis respects order: plot, decompose, check autocorrelation, forecast carefully.

Exam Tips

- ★ Key point: components of a time series.
- ★ Remember holdout validation for forecasts.

Topic 20: Categorical Data: Chi-Square Tests

Learning Objectives

- Run chi-square tests of association
- Interpret cross-tabulations
- Measure association strength

Key Concepts

For two categorical variables (gender by voting preference, treatment by outcome), cross-tabulation (Analyze > Descriptive Statistics > Crosstabs) with the chi-square test asks whether the variables are associated. The test compares observed cell counts to expected counts under independence; large discrepancies give small p-values. Assumptions: independent observations, expected counts ≥ 5 in most cells (else use Fisher's exact test).

Always read the percentages, not just the p-value: a significant chi-square with 52% vs. 48% is a trivial association in a huge sample. Direction comes from the table (which group is higher where); strength from Cramer's V (0-1; ~ 0.1 small, ~ 0.3 medium, ~ 0.5 large) or odds ratios for 2x2 tables. SPSS: Crosstabs > Statistics > Chi-square, Phi and Cramer's V; Cells > Row/Column percentages.

Example: training (yes/no) by promotion (yes/no), $n = 400$: chi-square $p = 0.002$, 35% of trained promoted vs. 22% of untrained, OR = 1.9. Report: 'Training was associated with promotion (chi-square $p = .002$); trained staff had 1.9 times the odds of promotion.' In R: `chisq.test(table(x, y))`. For paired categorical data (before/after), use McNemar's test.

Key Points

- Chi-square tests association between categorical variables; needs expected counts ≥ 5 .
- Read percentages for direction, Cramer's V / odds ratios for strength.
- McNemar for paired categorical data.

Key Terms

Expected counts: Cell counts under independence.

Cramer's V: 0-1 strength of categorical association.

Fisher's exact test: Exact alternative for small expected counts.

Example

- *Training vs. promotion: $p = .002$. OR = 1.9.*

Summary: Chi-square detects association; tables and effect sizes explain it.

Exam Tips

- ★ Key point: assumptions of chi-square.
- ★ Remember to report direction and strength, not just p.

Unit 6: Professional Practice and Exam Preparation

Topic 21: Missing Data, Outliers, and Assumptions

Learning Objectives

- Diagnose missing-data patterns
- Treat outliers principledly
- Run assumption checks routinely

Key Concepts

Missing data mechanisms: MCAR (missing completely at random - harmless, just lose power), MAR (missingness depends on observed data - e.g., income missing more for young respondents; multiple imputation fixes it), MNAR (missingness depends on the missing value itself - high earners hide income; needs special models). SPSS: Analyze > Missing Value Analysis shows patterns and Little's MCAR test. Strategy: report the rate, test MCAR, impute if MAR (5+ imputations, pool results), and sensitivity-check conclusions.

Outliers: detect (boxplots, $z > 3$, Mahalanobis distance for multivariate), then decide - correct errors, keep genuine extremes (and use robust methods: medians, trimmed means, robust regression), or transform (log income). Never delete silently to get significance - that is fabrication. Winsorizing (capping at percentiles) is a middle path.

Assumption checklist before every parametric analysis: independence of observations (design question), normality (Shapiro-Wilk, Q-Q - but $n > 30$ is forgiving), equal variances (Levene's), linearity (scatterplots/residual plots). Build the habit: descriptives, plots, assumption checks, then the test - in that order, every time. The few minutes this costs prevent the most embarrassing errors.

Key Points

- MCAR/MAR/MNAR determine the fix; multiple imputation for MAR.
- Outliers: correct, keep robustly, or transform - never silently delete.
- Checklist: independence, normality, equal variances, linearity - in that order.

Key Terms

MCAR/MAR/MNAR: Missing-data mechanisms.

Winsorizing: Capping extremes at percentiles.

Mahalanobis distance: Multivariate outlier measure.

Example

- *Income missing for young respondents (MAR) -> multiple imputation.*

Summary: Principled handling of data problems separates analysts from button-clickers.

Exam Tips

- ★ Very important: the three missing-data mechanisms.
- ★ Remember the assumption checklist order.

Topic 22: Reporting and Reproducible Research

Learning Objectives

- Write a results section
- Use APA-style reporting
- Build reproducible workflows

Key Concepts

A results section tells a story with numbers: describe the sample, then each analysis in logical order - descriptives first, then tests/models, each with the statistic, df, p-value, and effect size in APA style: $t(38) = 2.4$, $p = .02$, $d = 0.76$; $F(2, 87) = 6.2$, $p = .003$, $\eta^2 = .12$; $r(48) = .62$, $p < .001$. Tables carry the detail (means, SDs, ns); the text highlights the message. Figures need numbered captions and

in-text references.

Reproducible workflow: one project folder with data/ (raw, never edited), scripts/ (01_clean.R, 02_analyze.R, ...), output/ (tables, figures), and a README. In R, R Markdown weaves code, output, and narrative into one document that regenerates the whole report - the gold standard. In SPSS, keep a master syntax file with comments, run top-to-bottom.

Version control (Git) tracks every change; sharing the script + data lets anyone verify your work. The exam may ask you to critique a bad report - look for: missing sample description, p-values without effect sizes, unreported assumption checks, cherry-picked results, and irreproducible methods ('we analyzed the data in SPSS'). Knowing these sins also keeps your own work clean.

Key Points

- APA style: statistic(df) = value, p, effect size.
- Workflow: raw data untouched, scripts numbered, README included.
- R Markdown / master syntax = one-click reproducible reports.

Key Terms

APA style: Standard statistical reporting format.

R Markdown: Document weaving R code, output, and text.

Example

- $t(38) = 2.4$. $p = .02$. $d = 0.76$ - complete in one line.

Summary: Good reporting is honest, complete, and reproducible.

Exam Tips

- ★ Key point: APA reporting format.
- ★ Remember the project folder structure.

Topic 23: Worked Project: From Raw Data to Report

Learning Objectives

- Execute a complete analysis pipeline
- Interpret full SPSS/R output
- Write up findings

Key Concepts

Scenario: 120 employees; variables: training_hours (scale), dept (nominal: Sales/HR/IT), satisfaction (1-5 ordinal), promoted (0/1), salary (scale). Research questions: (1) Does training relate to promotion? (2) Do departments differ in satisfaction? (3) What predicts salary?

Pipeline. Clean: frequencies catch salary = 5500000 (typo, fix to 55000); 8% salary missing, MCAR test $p = 0.4$ -> listwise acceptable but impute to be safe. Descriptives: mean salary Rs. 62,000 (SD 18,000), median satisfaction 4. Visualize: boxplots of salary by dept; bar chart of promotion rate by training groups.

Analysis. (1) Logistic regression: promoted ~ training_hours + dept: training OR = 1.35 per 10 hours ($p = 0.004$); dept differences n.s. Classification 78%. (2) Satisfaction is ordinal and skewed -> Kruskal-Wallis by dept: $p = 0.02$; post-hoc: IT > HR. Report medians. (3) Multiple regression: salary ~ training + dept + satisfaction: R-squared_adj = 0.52; VIFs < 2; residuals fine; training b = Rs. 1,200/hour

($p < 0.001$).

Write-up: one paragraph per question with APA statistics, two figures, a limitations note (single company, cross-sectional - no causality), and the syntax/script appended. This pipeline - clean, explore, test, diagnose, report - is the course in miniature.

Key Points

- Pipeline: clean -> explore -> test -> diagnose -> report.
- Match method to variable type (ordinal -> Kruskal-Wallis).
- Append syntax/script; note limitations honestly.

Key Terms

Cross-sectional: Data at one time point - no causal claims.

Example

- *Training OR = 1.35 per 10 hours for promotion ($n = .004$).*

Summary: The full pipeline, end to end, on one realistic dataset.

Exam Tips

- ★ Very important: be able to outline the pipeline.
- ★ Remember method-to-variable matching.

Topic 24: Exam Strategy and Quick-Reference

Learning Objectives

- Choose methods under exam pressure
- Avoid classic software-exam mistakes
- Consolidate key decision rules

Key Concepts

Method-selection cheat sheet (memorize this table): comparing means, 2 groups, normal -> t-test (paired if same units); 3+ groups -> ANOVA (+Tukey); non-normal/small/ordinal -> Mann-Whitney / Kruskal-Wallis / Wilcoxon; two categorical -> chi-square; predict scale Y -> (multiple) regression; predict binary Y -> logistic; reduce items -> factor analysis/PCA; group cases -> cluster; scale quality -> Cronbach's alpha; time-ordered -> time series.

Classic mistakes: running ANOVA then t-tests without correction; reporting p without effect size; forgetting Levene's before independent t; using means for ordinal data; interpreting correlation as causation; deleting outliers silently; and - the most penalized - choosing the wrong test for the design (paired data with independent t-test). In practical exams, paste syntax for every dialog; label output; and never 'fish' - decide the analysis before seeing results.

Time plan: 10 minutes to read and plan (which test per question?), then execute; leave 15 minutes to write interpretations - interpretation marks often exceed computation marks. State assumptions checked, quote one key number per claim, and end each answer with a one-sentence conclusion in plain words. Examiners reward analysts who think, not just click.

Key Points

- Memorize the method-selection table above.
- Paste syntax, label output, decide analyses before running.

- Interpretations earn more marks than computations.

Key Terms

Fishing: Running many analyses hoping for significance.

Example

- *Paired design + independent t-test = the classic wrong choice.*

Summary: Choose right, check assumptions, interpret fully, report honestly.

Exam Tips

- ★ Very important: reproduce the method-selection table from memory.
- ★ Remember: interpretations > computations for marks.

Glossary of Key Terms

SPSS: Statistical package: menu-driven analysis, syntax-logged.

R: Free statistical programming language; research standard.

Reproducibility: Re-running analysis gives identical results.

p-value: Probability of data this extreme under H_0 .

Effect size: Magnitude of a difference or relationship.

t-test: Compares means: one-sample, independent, or paired.

ANOVA: Compares 3+ means via F = between/within variation.

Chi-square: Tests association between categorical variables.

Regression: Predicts scale Y from X 's; reports slopes and R -squared.

Logistic regression: Predicts binary Y via log-odds; reports odds ratios.

Factor analysis: Finds latent dimensions behind correlated items.

Cluster analysis: Groups cases by similarity.

Cronbach's alpha: Internal-consistency reliability of a scale.

VIF: Variance inflation factor; multicollinearity diagnostic.

Missing data (MCAR/MAR/MNAR): Mechanisms determining the proper fix.

ggplot2: R's grammar-of-graphics plotting system.

dplyr: R's data-wrangling grammar: filter/select/mutate/arrange/summarise.

APA style: $t(38) = 2.4$, $p = .02$, $d = 0.76$ reporting format.