Homework Grading Report

Student Name:	Trinity Schroeder
Assignment:	Assignment 1 - Introduction to R
Graded On:	October 03, 2025 at 11:40 PM
Final Score:	34.5 / 37.5 points (92.0%)

Score Summary

Overall Performance: Excellent (92.0%)

Instructor Assessment

Your work on this assignment demonstrates strong engagement and solid analytical skills. You successfully imported the messy dataset, inspected its structure, and provided a thorough written assessment of the quality issues you observed. Your reflection responses reveal thoughtful consideration of when to remove versus impute missing data, how to interpret outliers, and the ethical responsibilities of an analyst. The way you connected statistical concepts to real-world business scenarios-such as fraud detection and revenue forecasting-shows developing expertise in business analytics. To elevate your work further, complete the missing code sections (total missing count, boxplot, winsorization) and include the corresponding visual outputs. Incorporating more advanced imputation tools and systematic string-cleaning pipelines will also deepen your skill set. Overall, you are making excellent progress in mastering data cleaning and its business implications; keep building on this strong foundation.

Reflection & Critical Thinking

Your answer to Question 1 clearly distinguishes when removal is preferable (e.g., financial audits or clinical trials) and when imputation makes sense (e.g., retail sales). You linked each scenario to the underlying business risk, which shows strong critical thinking about data integrity. In Question 2 you identified several plausible reasons for outliers-bulk purchases, refunds, data entry errors, and fraud-and argued that the decision to keep or drop them depends on context. This nuanced view demonstrates a solid grasp of the business implications of outlier treatment.

Analytical Strengths

Your code successfully loads the tidyverse, reads the messy_sales CSV, and prints a clear overview of the dataset, including row and column counts. This demonstrates that you can set up a reproducible analysis environment. The structure (str) and summary (summary) calls you included give a comprehensive snapshot of variable types, missingness, and value ranges. Using these functions early in the workflow shows good analytical discipline. Your written assessment of data quality issues-missing values, outliers, inconsistent naming, wrong data types, and negative values-is detailed and directly tied to the output you observed. This connection between code output and narrative is a strength.

Business Application

When you described outliers as potential bulk purchases or fraudulent transactions, you linked a statistical observation to concrete business actions such as fraud detection or inventory planning. This shows you can translate data patterns into operational insights. Your explanation of why imputation preserves sample size in a retail context highlights the impact on revenue forecasting and product performance analysis. Connecting methodological choices to business outcomes is exactly what analysts need to do. By noting that negative quantities are logically impossible for sales data, you identified a data-quality rule that could trigger alerts in an ETL pipeline, preventing downstream reporting errors. This demonstrates an ability to think about data governance.

Learning Demonstration

Your discussion of the trade-offs between removal and imputation indicates you understand concepts such as bias, statistical power, and the assumptions behind each method. This reflects a solid foundation in missing-data theory. You correctly identified that the Purchase_Date column may be imported as a character and need conversion to a Date type, showing awareness of type-casting issues that affect time-series analysis. Your reflection on ethical implications shows you recognize that data cleaning is not just a technical step but a decision-making process with stakeholder impact. This maturity will serve you well in professional settings.

Areas for Development

The TODO line for calculating total missing values (total_mi) is still present; implement a simple sum(is.na()) across the dataframe and store the result. Completing this step will give you a quantitative baseline for missing-data severity. You mentioned creating a boxplot for Sales_Amount but did not include the ggplot2 code. Add a boxplot with geom_boxplot() and annotate the outlier points; this visual evidence will strengthen your outlier-detection narrative. When you discuss capping (winsorization), consider showing the actual transformation using the pmin/pmax functions or the scales::squish() helper. Demonstrating the before-and-after values will make your treatment choice more transparent.

Recommendations for Future Work

Practice using the mice package for multiple-imputation on a larger dataset; it will give you exposure to more sophisticated imputation techniques and variance estimation. Explore the use of dplyr's mutate() together with case_when() to clean inconsistent Product_Category names in a single pipeline. This will improve code readability and reproducibility. Continue documenting every cleaning step in markdown cells, including the rationale and any assumptions. A well-annotated notebook becomes a valuable artifact for teammates and auditors.

Technical Analysis

Code Strengths:

- You successfully implemented the mode function using unique(), tabulate(), match(), and which.max() which correctly identifies the most frequent value in categorical data. Your approach to imputing Customer_Name with the mode value demonstrates solid understanding of handling missing categorical data. The function works as intended and produces accurate results for the imputation task.

• Your handling of missing value removal using na.omit() is correct and produces the expected dimension comparisons. You properly calculated missing values per column and identified rows with missing data, showing good comprehension of data quality assessment techniques. The code structure for removing NAs is clean and functional.

• You effectively used the IQR method for outlier detection by correctly calculating quartiles, IQR, and thresholds. Your filtering approach with the pipe operator to identify outliers demonstrates good understanding of data cleaning workflows. The boxplot visualization you created properly displays outliers with red dots, which is a professional approach to outlier visualization.

Code Improvement Suggestions:

• Your mode function implementation is correct but could be simplified by using the table() function directly: `get_mode <- function(v) { names(which.max(table(v[!is.na(v)]))) }`. This alternative approach is more concise and equally effective for finding the mode of categorical variables.

• For the median imputation of Quantity, you should assign the median value directly to the column: `sales_imputed$Quantity[is.na(sales_imputed$Quantity)] <- median(sales_imputed$Quantity, na.rm = TRUE)`. Your current approach creates an unnecessary intermediate variable and doesn't actually impute the data correctly.

• You duplicated several sections of code including the outlier detection and boxplot creation. You should remove the redundant code blocks to make your script more efficient and easier to maintain. Consider creating a function for the outlier detection process to avoid repeating the same calculations multiple times.

Technical Observations:

• You demonstrate solid understanding of data cleaning workflows through your implementation of missing value handling and outlier detection. Your ability to use base R functions alongside tidyverse syntax shows good versatility in programming approaches. This foundation will serve you well as you tackle more complex business analytics problems.

• Your approach to data validation through summary statistics comparison shows good analytical thinking. You correctly tracked changes in dataset dimensions and summary statistics throughout the cleaning process, which is essential for reproducible data analysis. This attention to detail is crucial for professional analytics work.

• Your code organization supports reproducible analysis through clear sectioning and descriptive comments. You structured the outlier detection process in a logical sequence that makes the analysis easy to follow. The use of print statements for key results helps verify that each step of your cleaning process is working as expected.

Additional Code Enhancement Examples:

Data Exploration Enhancement:

```
# More comprehensive data inspection
```

```
glimpse(sales_df) # dplyr alternative to str()
```

```
skimr::skim(sales_df) # Detailed summary statistics
```

```
DataExplorer::plot_missing(sales_df) # Visualize missing data
```

Data Visualization:

```
# Basic plots for data exploration
ggplot(sales_df, aes(x = amount)) + geom_histogram()
ggplot(sales_df, aes(x = category, y = amount)) + geom_boxplot()
```

Data Cleaning:

```
# Handle missing values
sales_df <- sales_df %>%
filter(!is.na(amount)) %>%
mutate(amount = ifelse(amount < 0, 0, amount))
```

Performance by Category