Homework Grading Report

Student Name:	Kush Patel
Assignment:	Assignment 1 - Introduction to R
Graded On:	October 03, 2025 at 10:40 PM
Final Score:	29.5 / 37.5 points (78.7%)

Score Summary

Overall Performance: Satisfactory (78.7%)

Instructor Assessment

You have laid a solid groundwork by importing the dataset and performing basic inspections, which is essential for any data■cleaning workflow. However, many critical components of the assignment remain incomplete, especially the implementation of imputation, outlier treatment, and the reflective discussion of business implications. Focus on filling those gaps, polishing your written responses, and linking each technical decision back to a real■world business scenario. With those improvements, you will demonstrate a more complete mastery of data cleaning concepts.

Reflection & Critical Thinking

Your answer to the first reflection question mentions that removing rows is preferred when the proportion of missing data is very small, which is a correct intuition, but the response is incomplete and contains several typographical errors that make it hard to follow. You left the second reflection question completely blank, so the grader cannot see how you would interpret outliers in a business context; providing a concrete example (e.g., unusually large sales due to a promotional event) would demonstrate deeper understanding. When you discuss trade■offs between removal and imputation, you did not provide any reasoning or examples, which limits the insight you convey about the impact on analysis results.

Analytical Strengths

You correctly loaded the tidyverse library and used `read_csv` to import the messy_sales dataset, showing you understand the basic data import workflow in R. Your use of `str(messy_sales)` and `summary(messy_sales)` to inspect the structure and summary statistics is appropriate for an initial data assessment, and the printed messages confirm that the dataset was read successfully. The code that prints the number of rows and columns, as well as the column names, demonstrates that you can extract useful metadata from a data frame, which is a solid first step in any cleaning pipeline.

Business Application

You identified that missing values need to be handled, which is a common business problem, but you did not connect specific variables (e.g., Customer_ID or Sales_Amount) to business decisions such as

forecasting or inventory planning. Your brief mention of outliers in the Sales_Amount column hints at a potential impact on revenue analysis, yet you did not elaborate on how those extreme values could affect profit margins or pricing strategies. Linking the cleaning choices (removal vs. imputation) to downstream business outcomes—like model accuracy for sales prediction—would strengthen the relevance of your work.

Learning Demonstration

You demonstrated familiarity with basic R commands for data inspection, which shows you have grasped the foundational syntax needed for data management tasks. The notebook structure follows the assignment template, indicating you understand how to organize a data■cleaning project into logical sections. However, the lack of completed code for imputation, outlier detection, and final justification suggests that you still need practice applying the concepts beyond the initial inspection.

Areas for Development

Complete the missing sections: implement at least one imputation method (e.g., median for numeric, mode for categorical) and show the before■and■after impact on summary statistics. Add visualizations for outlier detection (boxplot or histogram) and explain why you chose removal or capping, referencing the IQR method you were asked to use. Expand your reflection answers with concrete business scenarios, correct spelling, and full sentences to improve clarity and demonstrate critical thinking.

Recommendations for Future Work

Practice applying different imputation techniques on a small sample dataset, then compare the results using visual checks and summary tables to build confidence in handling missing data. Explore the `boxplot.stats` function or the `outliers` package for systematic outlier identification, and try winsorization with the `DescTools::Winsorize` function to see how capping affects the distribution. Read a short article on the business implications of data cleaning (e.g., how imputed values can bias predictive models) and incorporate those insights into future reflection sections.

Technical Analysis

Code Strengths:

- You correctly load the tidyverse package and successfully import the messy_sales dataset using read_csv. Your use of getwd() to check the working directory shows good practice for debugging. The print statements that display dataset dimensions and column names are well-structured and provide clear feedback about the data structure.
- Your approach to calculating missing values is logical and well-implemented. You correctly use colSums(is.na(messy_sales)) to count total missing values and colSums(is.na(messy_sales)) to get missing values per column. Your identification of rows with missing values using which(is.na(messy_sales)) demonstrates solid understanding of data inspection techniques.
- You properly implement the removal of rows with missing values using na.omit() which is a valid and efficient approach. Your comparison of dataset dimensions before and after removal shows clear understanding of data loss implications. The summary statistics comparison for Sales_Amount across different datasets demonstrates good analytical thinking.

Code Improvement Suggestions:

• While your use of na.omit() works correctly for removing rows with missing values, you could also consider using filter(!is.na(Sales_Amount)) to be more explicit about which column to check for missing values. This approach would make your code more readable and maintainable, especially when working with datasets that have missing values in different columns.

• Your mode function implementation is on the right track but needs refinement. Instead of using match() and which.max() directly, you should use table() to count frequencies and then identify the most frequent value. A more robust version would be: get_mode <- function(x) { names(which.max(table(x))) }. This approach handles edge cases better and is more commonly used in practice.

• For imputing the Quantity column with median, you should use median(Quantity, na.rm = TRUE) instead of just median(Quantity). Your current approach will produce an error because median() doesn't handle NA values by default. This is a common mistake when working with missing data in R, and fixing it will make your imputation process work correctly.

Technical Observations:

• You demonstrate solid understanding of data quality assessment through your systematic approach to identifying missing values, outliers, and data inconsistencies. Your ability to use colSums(is.na()) for missing value counting shows good technical competency in data inspection. This foundation will serve you well as you tackle more complex data cleaning challenges.

• Your implementation of outlier detection using IQR method is well-structured and produces correct results. You correctly calculate quartiles, IQR, and thresholds, then identify outliers using proper logical conditions. Your use of ggplot2 for boxplot visualization shows good integration of visualization with data analysis, which is essential for business analytics work.

• Your code organization supports reproducible analysis through clear section breaks and logical flow from data import to final dataset selection. You structured your analysis in a way that makes the logic clear and the results verifiable. This attention to code quality and organization is important for professional analytics work and demonstrates good programming practices.

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