Homework Grading Report

Student Name:	Davion Hosein
Assignment:	Assignment 1 - Introduction to R
Graded On:	October 03, 2025 at 11:40 PM
Final Score:	32.3 / 37.5 points (86.1%)

Score Summary

Overall Performance: Good (86.1%)

Instructor Assessment

Your submission shows a solid grasp of the fundamental data-cleaning concepts we covered in class. You correctly identified several data-quality issues, chose reasonable strategies for handling missing values and outliers, and linked those choices to business considerations. The reflection answers are concise but would benefit from deeper analysis of when each method is appropriate and what the downstream effects might be. Completing the unfinished code sections and providing quantitative evidence of the impact of your cleaning steps will elevate your work to the next level. Keep building on this foundation by exploring more advanced imputation tools and by documenting every transformation you make. Overall, you are on the right track and demonstrating growing competence in business analytics.

Reflection & Critical Thinking

Your answer to the missing-value strategy question correctly identifies mean/median imputation for numeric fields and mode for categorical fields, showing you understand the basic idea, but you did not discuss when each method might be inappropriate or how it could affect downstream analysis. When you compared removal versus imputation, you noted that removal can lead to data loss and imputation can prevent bias; however, you did not elaborate on the potential introduction of artificial variance or the assumptions behind each technique.

Analytical Strengths

You successfully loaded the tidyverse and readr packages, imported the messy_sales CSV, and printed the dataset dimensions and column names, which demonstrates solid foundational R skills. Your initial data assessment identified several concrete quality issues-missing purchase dates, negative quantities, inconsistent product-category capitalization, and extreme sales amounts-showing you can spot red flags in raw data. Choosing a capping (winsorization) approach for the Sales_Amount outliers reflects an understanding of preserving sample size while limiting the influence of extreme values, which is appropriate for many business analytics scenarios.

Business Application

By selecting the capped dataset as your final version, you linked the technical decision to a business rationale: retaining all transactions while reducing distortion from outliers, which is valuable for accurate revenue reporting. Your brief note that negative quantities are logically impossible hints at an awareness of how data errors can mislead inventory forecasts, but a deeper discussion of the financial impact would make the connection clearer. Mentioning that imputation provides a more accurate representation of the data shows you recognize the trade-off between completeness and potential bias, an important consideration for any data-driven decision making.

Learning Demonstration

Your work demonstrates that you understand the concepts of missing data, outlier detection, and basic cleaning strategies, as evidenced by the observations you recorded after running str() and summary(). You correctly identified the need to treat both missing values and outliers before any downstream analysis, indicating you are building a solid workflow for data preparation. The inclusion of reflection questions and your answers indicate you are beginning to think critically about the implications of each cleaning step, which is a key learning outcome for this module.

Areas for Development

Complete the missing-value calculation (e.g., total_missing <- sum(is.na(messy_sales))) and implement the actual imputation code so you can compare before-and-after statistics; this will allow you to quantify the impact of your chosen strategy. When discussing trade-offs, provide concrete examples such as how mean imputation could inflate average sales in a high-variance product line, or how listwise deletion might bias customer-segmentation results.

Recommendations for Future Work

Practice using the dplyr mutate() and if_else() functions to perform conditional imputation, and explore the mice package for more sophisticated multiple imputation techniques. Apply robust statistical methods such as the median absolute deviation (MAD) for outlier detection on additional variables like Quantity, and compare the results of removal versus capping on key business metrics. Create a reproducible data-cleaning script that logs every change (e.g., using the logger package) and generates a before-and-after summary table; this will reinforce good documentation habits and improve auditability.

Technical Analysis

Code Strengths:

- Your implementation of the mode function using unique(), tabulate(), match(), and which.max() successfully calculates the most frequent value for categorical data. You correctly applied this function to impute missing Customer_Name values, which shows solid understanding of handling missing data in R. The code executes without errors and produces the expected results for categorical imputation.
- Your approach to outlier detection using the IQR method is well-executed and produces accurate results. You correctly calculated quartiles, IQR, and thresholds before identifying outliers in the Sales_Amount column. Your use of filter() to remove outliers and create a capped version demonstrates good analytical thinking and proper data cleaning techniques.
- You effectively used na.omit() to remove rows with missing values and properly compared dimensions before and after removal. Your code structure for data cleaning shows good

organization and logical flow. The summary statistics comparison between original, removed NA, and imputed datasets clearly demonstrates the impact of different cleaning approaches.

Code Improvement Suggestions:

- While your use of ifelse() for imputing missing values works correctly, you could also consider using the tidyr::replace_na() function which might simplify the code: sales_imputed$Customer_Name <- replace_na(sales_imputed$Customer_Name, get_mode(sales_imputed$Customer_Name)). This alternative approach offers cleaner syntax and is commonly used in professional analytics work.

- Your outlier detection section could be enhanced by incorporating the boxplot visualization more effectively: ggplot(sales_imputed, aes(y = Sales_Amount)) + geom_boxplot(outlier.colour = 'red', outlier.shape = 16) + ggtitle('Sales Amount Outliers') + ylab('Sales Amount'). This would provide better visualization of the outlier detection results and make your analysis more robust.

- An alternative approach for creating the final comparison summary would be to use dplyr::bind_rows() to combine the data frames more efficiently: comparison_summary <- bind_rows(original = original_data, final = final_data, .id = 'Dataset'). This technique offers cleaner code structure and builds on the skills you've already demonstrated.

Technical Observations:

- You demonstrate solid understanding of data cleaning concepts through your implementation of multiple imputation strategies including mode imputation for categorical data and median imputation for numeric data. Your ability to handle missing values using different approaches shows good grasp of fundamental data preprocessing techniques. This foundation will serve you well as you tackle more complex analyses.

- Your appropriate use of base R functions like na.omit(), is.na(), and quantile() for data cleaning and outlier detection shows good analytical thinking. You correctly applied statistical concepts to achieve meaningful data transformations. This demonstrates developing competency in business analytics programming with R.

- Your code organization supports reproducible analysis through clear sectioning and logical flow from data import to final dataset creation. You structured the outlier detection section in a way that makes the logic clear and the results verifiable. This attention to code quality is important for professional analytics work and shows 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