Data_cleaning.R

badrinathsanagavaram

2023-12-11

```
## Data Cleaning and feature engineering
#getwd()
#setwd("/Users/badrinathsanagavaram/Desktop/R Project/")
#install.packages("tidyverse")
#install.packages("dplyr")
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- 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 error
```

```
library(dplyr)
library(tidyr)
library(ggplot2)
data = read_csv("50000 Sales Records.csv")
```

```
## Rows: 50000 Columns: 14
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): Region, Country, Item Type, Sales Channel, Order Priority, Order Da...
## dbl (7): Order ID, Units Sold, Unit Price, Unit Cost, Total Revenue, Total C...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
View(data)
head(data)
```

```
## # A tibble: 6 x 14
##   Region       Country ‘Item Type‘ ‘Sales Channel‘ ‘Order Priority‘ ‘Order Date‘
##   <chr>        <chr>   <chr>       <chr>           <chr>            <chr>
## 1 Sub-Saharan~ Namibia Household   Offline         M                8/31/2015
## 2 Europe       Iceland Baby Food   Online          H                11/20/2010
## 3 Europe       Russia  Meat        Online          L                6/22/2017
## 4 Europe       Moldova Meat        Online          L                2/28/2012
## 5 Europe       Malta   Cereal      Online          M                8/12/2010
## 6 Asia         Indone~ Meat        Online          H                8/20/2010
## # i 8 more variables: ‘Order ID‘ <dbl>, ‘Ship Date‘ <chr>, ‘Units Sold‘ <dbl>,
## #   ‘Unit Price‘ <dbl>, ‘Unit Cost‘ <dbl>, ‘Total Revenue‘ <dbl>,
## #   ‘Total Cost‘ <dbl>, ‘Total Profit‘ <dbl>
```

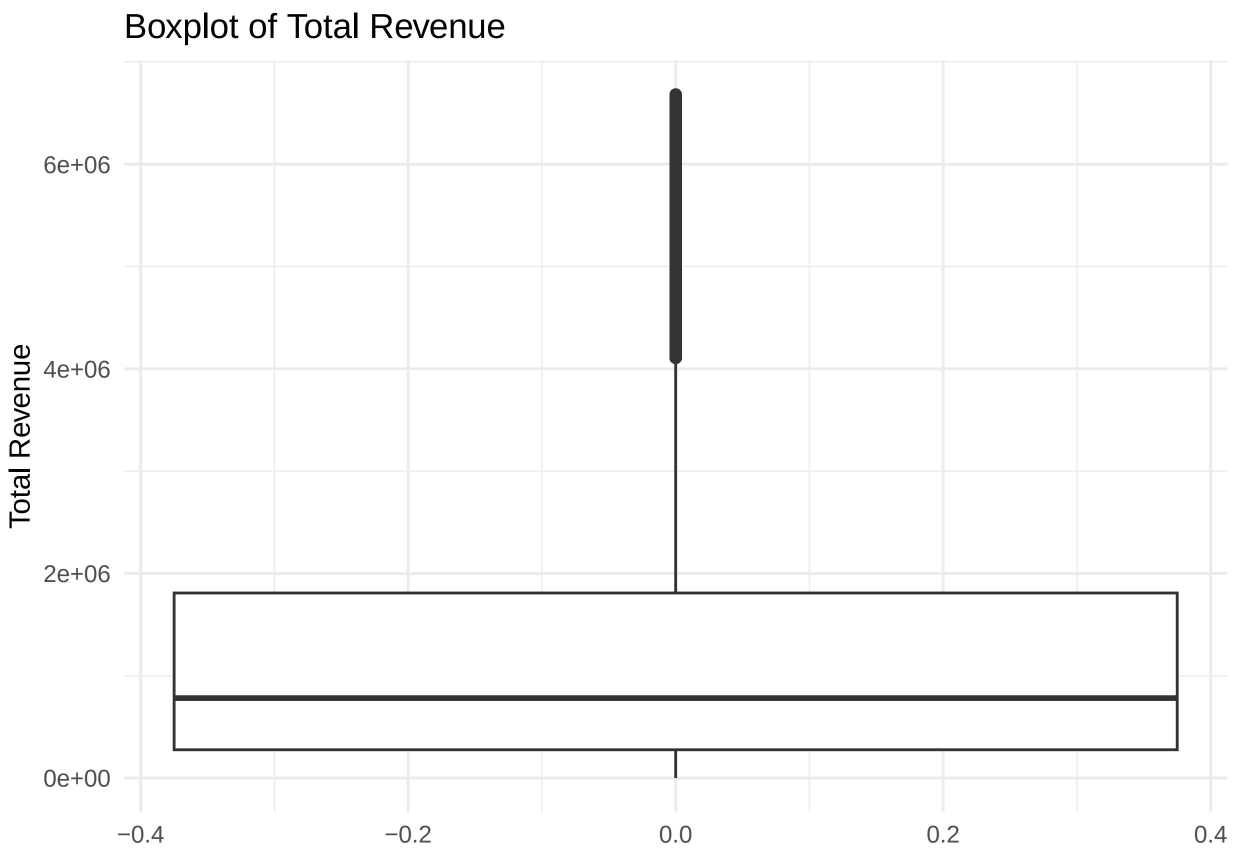
```
colnames(data)
```

```
##  [1] "Region"         "Country"        "Item Type"      "Sales Channel"
##  [5] "Order Priority" "Order Date"     "Order ID"       "Ship Date"
##  [9] "Units Sold"     "Unit Price"     "Unit Cost"      "Total Revenue"
## [13] "Total Cost"     "Total Profit"
```

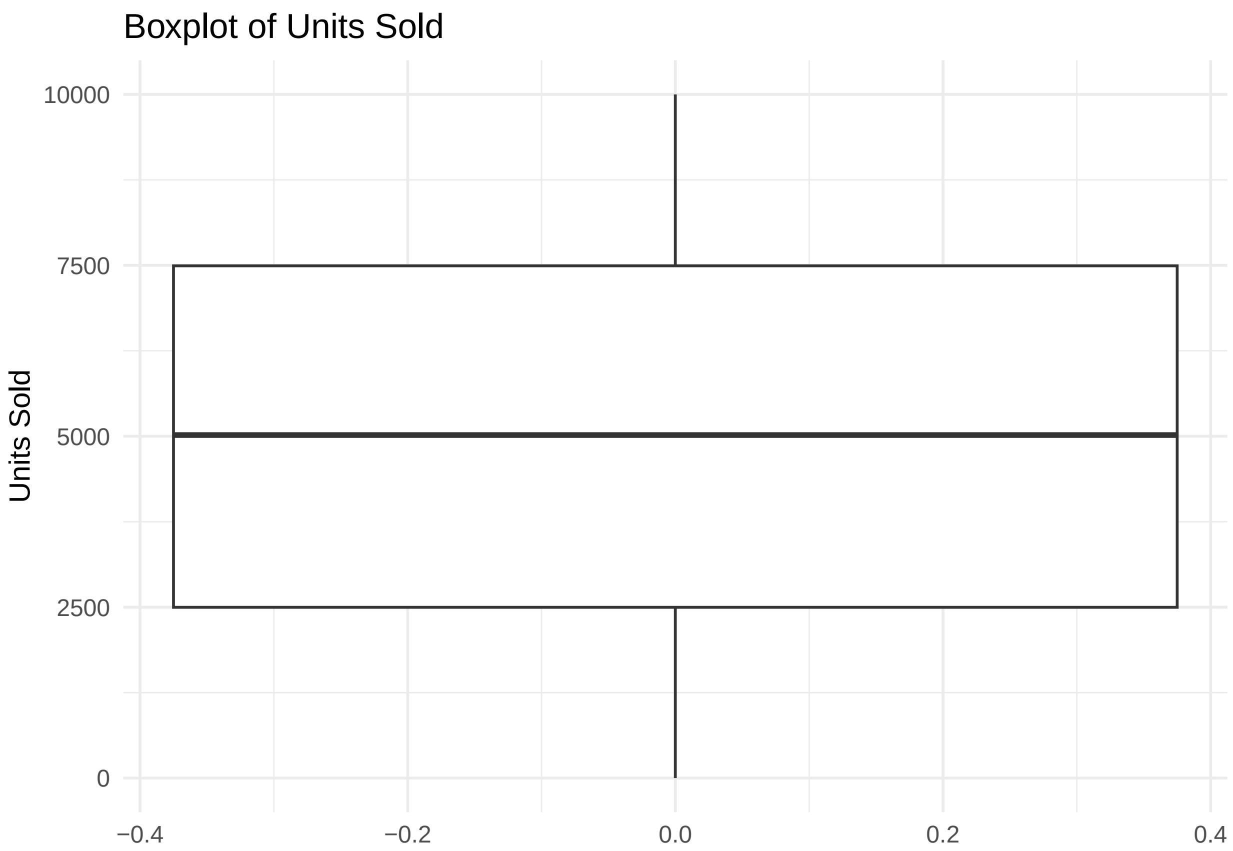
```
# Checking for null values in each column
null_count <- colSums(is.na(data))
# columns with null values and their counts
print(null_count)
```

```
##         Region        Country      Item Type  Sales Channel Order Priority
##              0              0              0              0              0
##     Order Date       Order ID      Ship Date     Units Sold     Unit Price
##              0              0              0              0              0
##      Unit Cost  Total Revenue     Total Cost   Total Profit
##              0              0              0              0
```

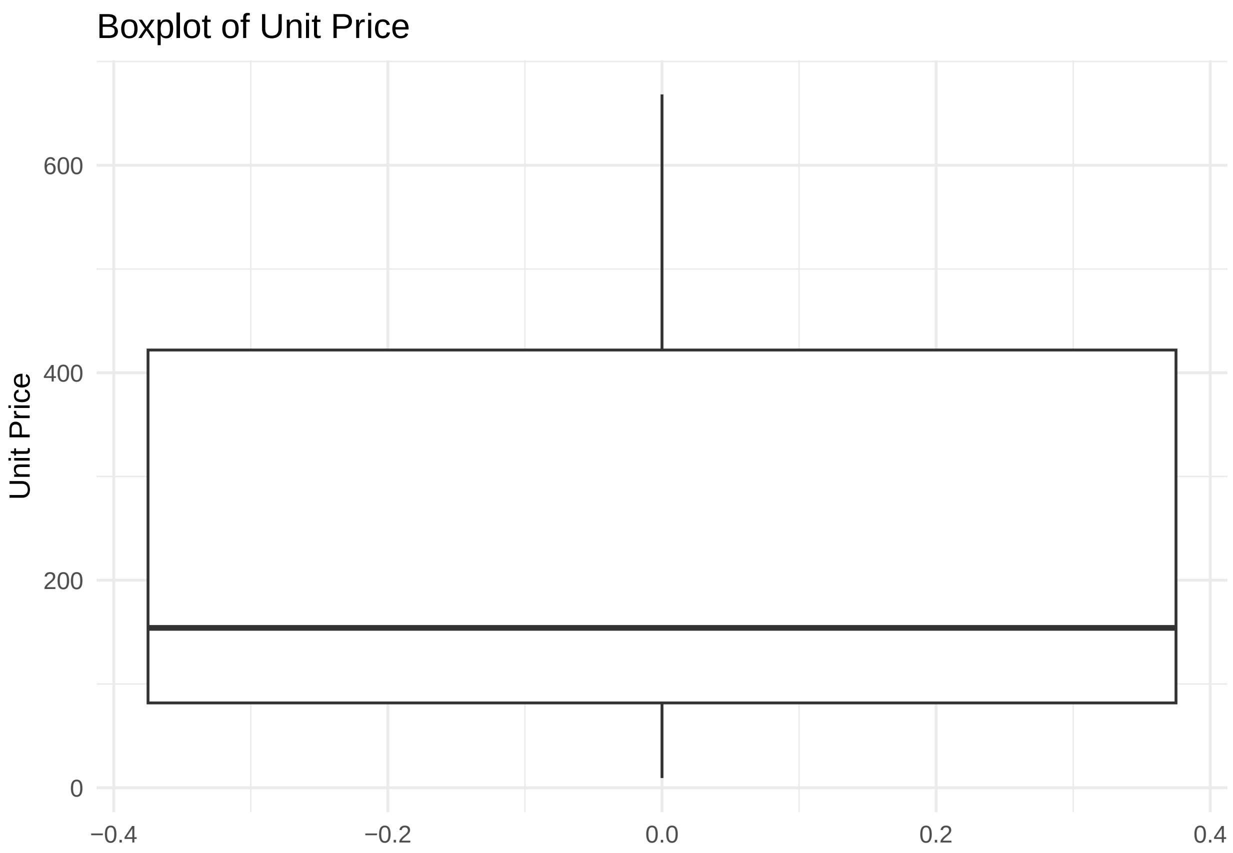
```
#boxplots for necessary columns
boxplot_total_revenue <- ggplot(data, aes(y = `Total Revenue`)) +
  geom_boxplot() +
  labs(title = "Boxplot of Total Revenue") +
  theme_minimal()

boxplot_units_sold <- ggplot(data, aes(y = `Units Sold`)) +
  geom_boxplot() +
  labs(title = "Boxplot of Units Sold") +
  theme_minimal()

boxplot_unit_price <- ggplot(data, aes(y = `Unit Price`)) +
  geom_boxplot() +
  labs(title = "Boxplot of Unit Price") +
  theme_minimal()

boxplot_total_profit <- ggplot(data, aes(y = `Total Profit`)) +
  geom_boxplot() +
  labs(title = "Boxplot of Total Profit") +
  theme_minimal()

# printing boxplots and bar charts
print(boxplot_total_revenue) ## has outliers
```



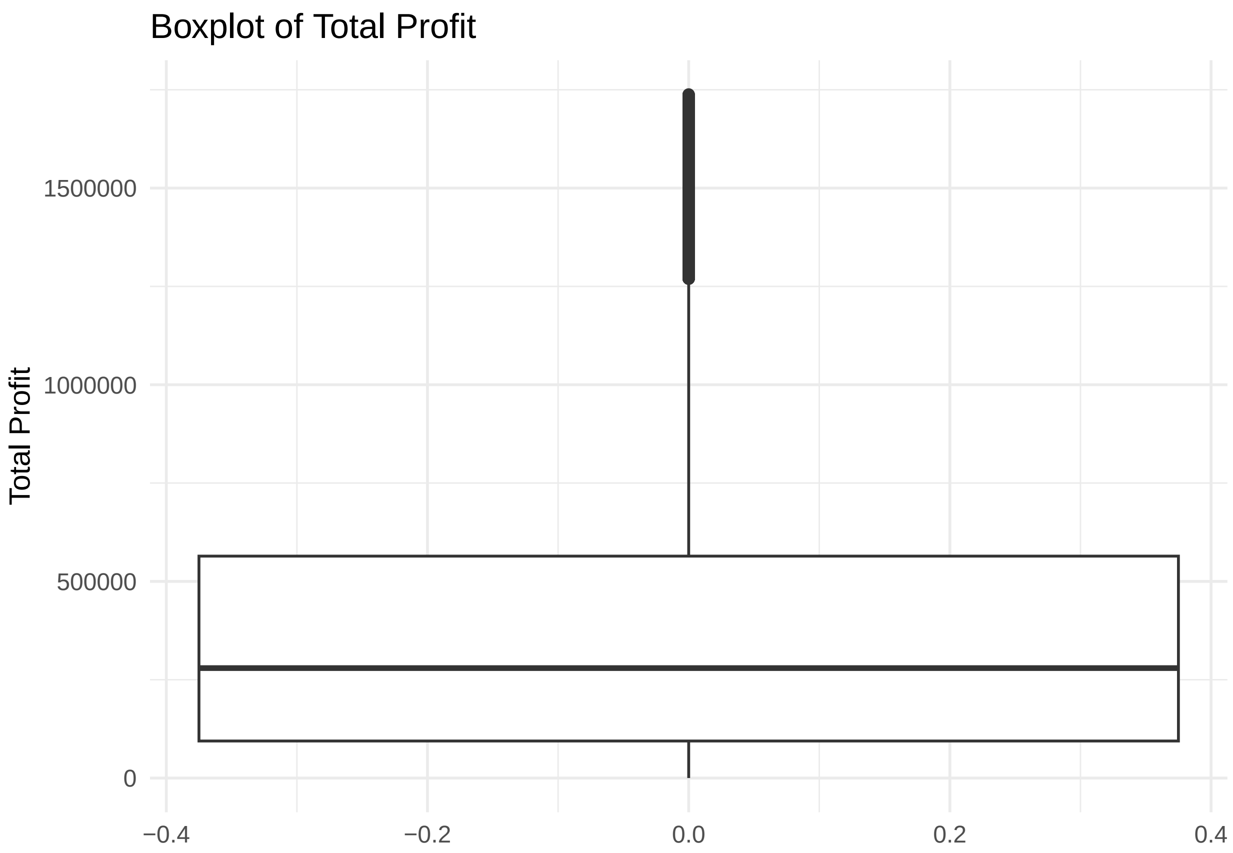
```
print(boxplot_units_sold)
```



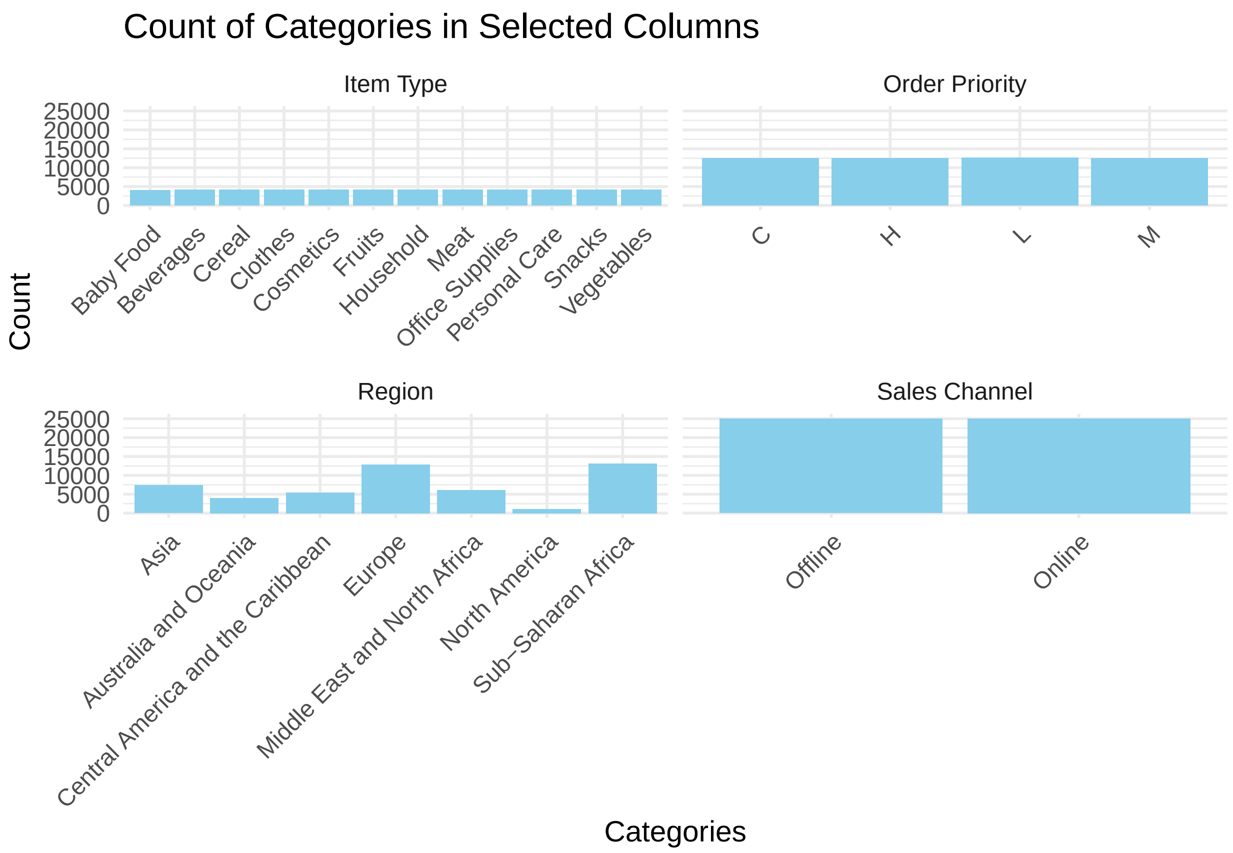
```
print(boxplot_unit_price)
```



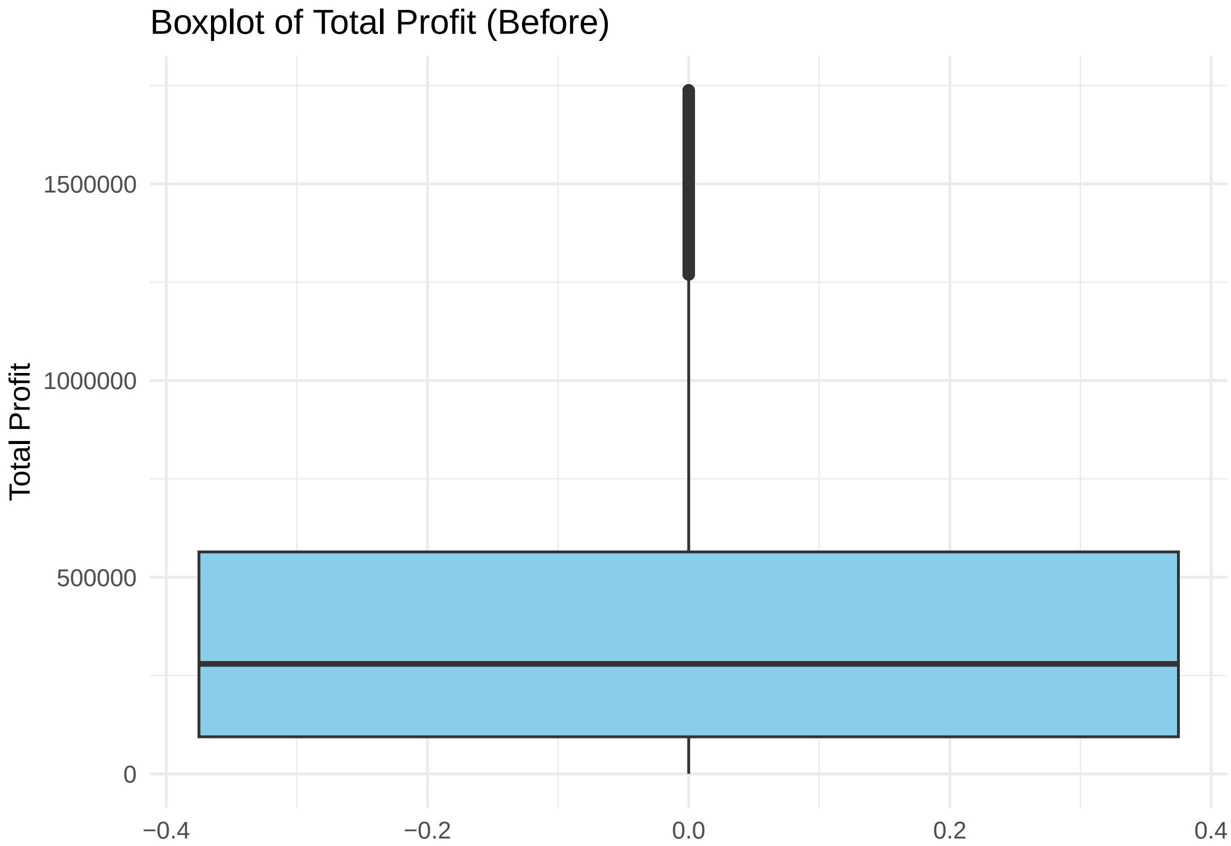
```r
print(boxplot_total_profit) ## has outliers
```



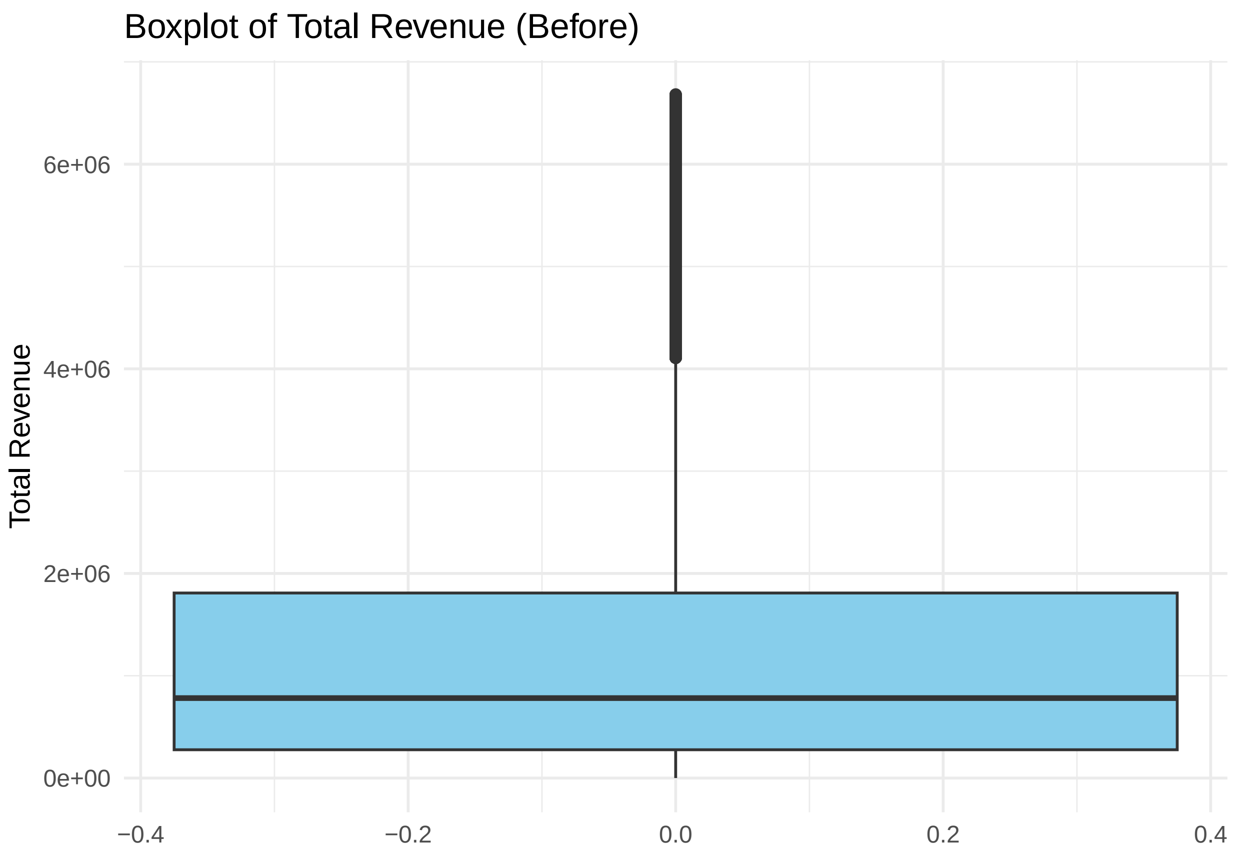
```r
# Subset the data for the specified columns
selected_columns <- c("Sales Channel","Region","Order Priority","Item Type")
selected_data <- data %>% select(all_of(selected_columns))

# Melting the data to long format for visualization
melted_data <- selected_data %>%
  tidyr::gather(key = "Variable", value = "Value")  # Reshape to long format

# Plotting bar charts
bar_chart_1 <- ggplot(melted_data, aes(x = as.factor(Value))) +
  geom_bar(fill = "skyblue", position = "dodge") +
  facet_wrap(~Variable, scales = "free_x") +
  labs(title = "Count of Categories in Selected Columns", x = "Categories", y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Displaying bar chart
print(bar_chart_1)
```



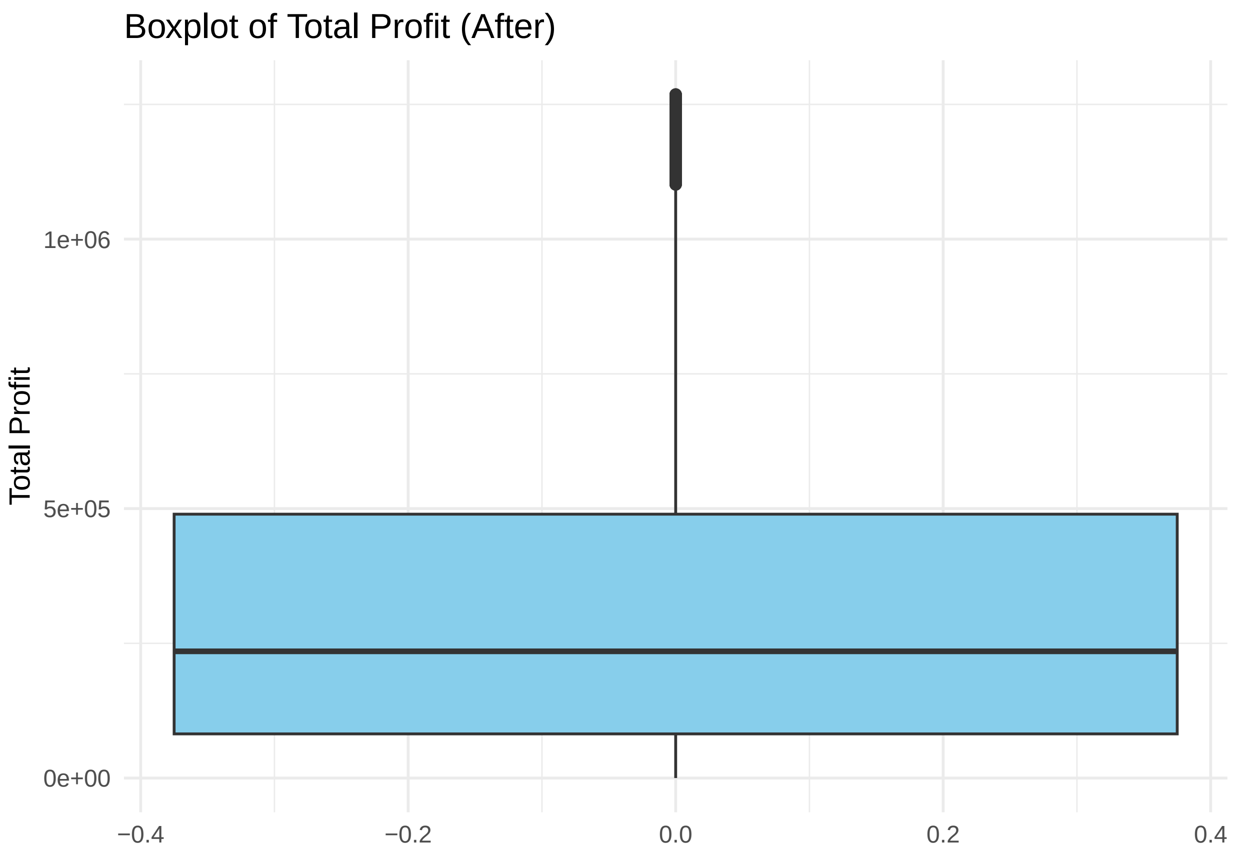
```
## Droping Values
# Detect outliers in 'Total Profit' and 'Total Revenue' columns
outliers_profit <- boxplot.stats(data$`Total Profit`)$out
outliers_revenue <- boxplot.stats(data$`Total Revenue`)$out

# Filter out rows without outliers
cleaned_data <- data %>%
  filter(!(`Total Profit` %in% outliers_profit) & !(`Total Revenue` %in% outliers_revenue))

boxplot_profit_after <- ggplot(cleaned_data, aes(y = `Total Profit`)) +
  geom_boxplot(fill = "skyblue") +
  labs(title = "Boxplot of Total Profit (After)", y = "Total Profit") +
  theme_minimal()

# Boxplot for 'Total Revenue' after removing outliers
boxplot_revenue_after <- ggplot(cleaned_data, aes(y = `Total Revenue`)) +
  geom_boxplot(fill = "skyblue") +
  labs(title = "Boxplot of Total Revenue (After)", y = "Total Revenue") +
  theme_minimal()
boxplot_profit_before <- ggplot(data, aes(y = `Total Profit`)) +
  geom_boxplot(fill = "skyblue") +
  labs(title = "Boxplot of Total Profit (Before)", y = "Total Profit") +
  theme_minimal()

# Boxplot for 'Total Revenue' before removing outliers
boxplot_revenue_before <- ggplot(data, aes(y = `Total Revenue`)) +
  geom_boxplot(fill = "skyblue") +
  labs(title = "Boxplot of Total Revenue (Before)", y = "Total Revenue") +
  theme_minimal()

print(boxplot_profit_before)
```



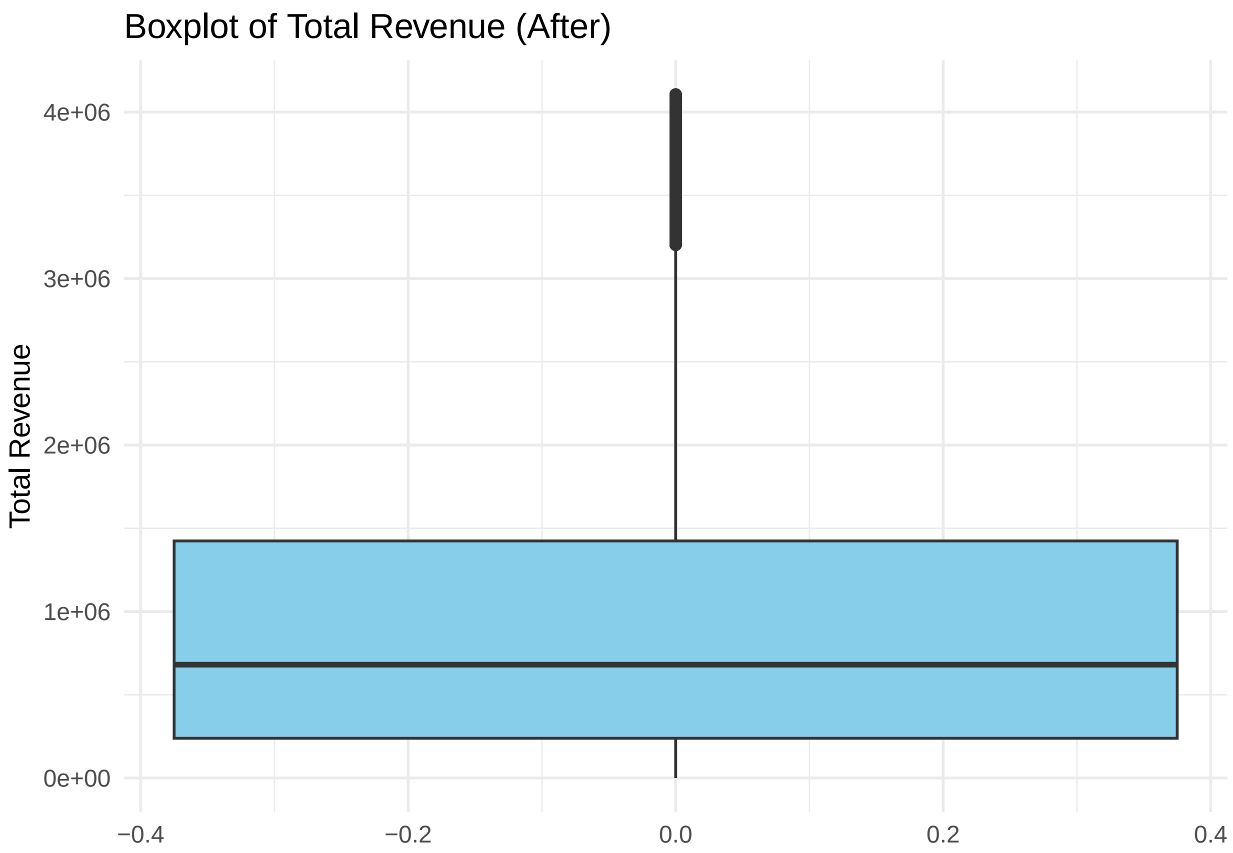
```
print(boxplot_revenue_before)
```



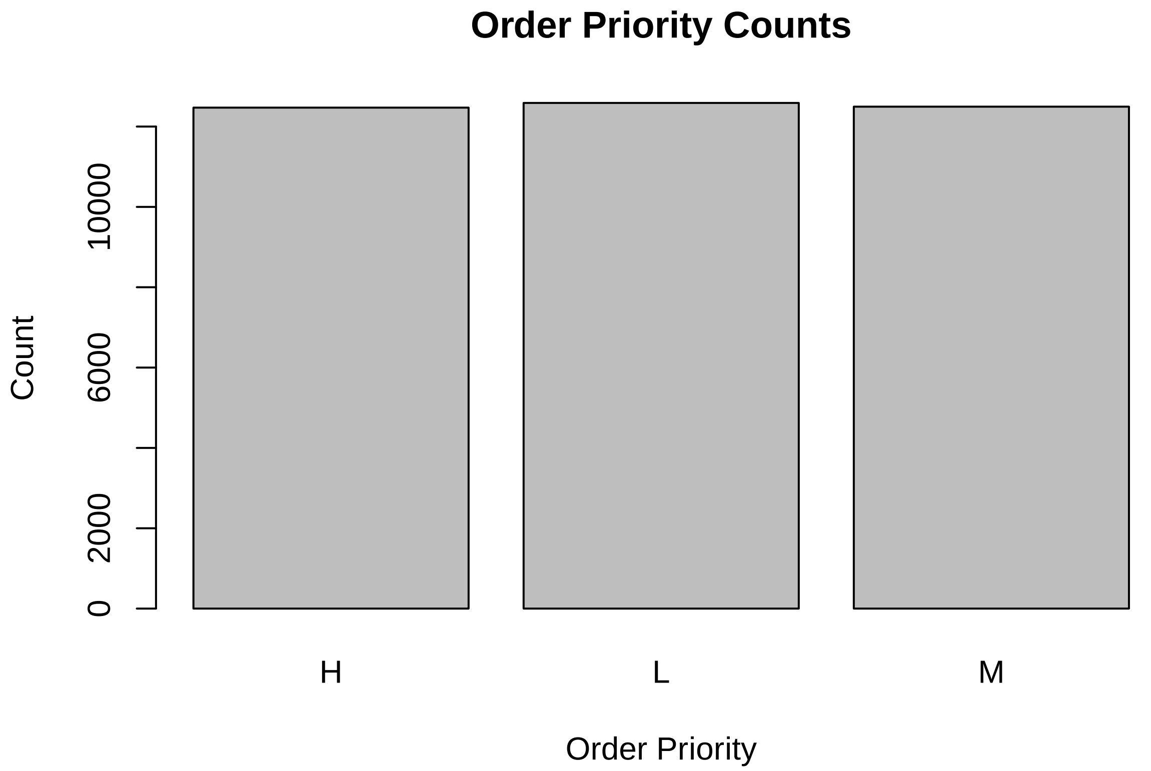
```
print(boxplot_profit_after)
```



```r
print(boxplot_revenue_after)
```



```r
# Filtering rows where Order Priority does not contain 'C'
cleaned_data <- data %>% filter(!grepl("C", `Order Priority`))

# Save cleaned data to a new CSV file
write.csv(cleaned_data, file = "Cleaned_data.csv", row.names = FALSE)

# Calculating value counts for Order Priority
order_priority_counts <- table(cleaned_data$`Order Priority`)

# Creating a bar plot
barplot(order_priority_counts,
        main = "Order Priority Counts",
        xlab = "Order Priority",
        ylab = "Count")
```



```r
# Writing the cleaned data to a CSV file
write.csv(cleaned_data, file = "yours_data1.csv", row.names = FALSE)
```