

Analyzing Furniture Sales Data by Trager Bos

#Intro: In this analysis, we will explore a comprehensive dataset of furniture sales to uncover insights about sales trends, revenue distribution, and various pricing factors. To begin, we will install the necessary libraries and packages, such as tidyverse, knitr, and scales, which provide essential data manipulation and visualization tools. Following the setup, we will import the dataset from a CSV file and perform data cleaning to handle missing values. A summary of the cleaned dataset will be generated, including column names, dimensions, and a statistical overview. Finally, we will create visualizations to illustrate total sales and revenue by season, store type, and brand, along with scatterplots that depict the relationship between price and cost. This analysis aims to provide valuable insights for better-understanding furniture sales dynamics and identifying growth opportunities.

Install appropriate libraries & packages

```
install.packages("tidyverse")

##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpzhXlbh/downloaded_packages

install.packages("knitr")

##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpzhXlbh/downloaded_packages

install.packages("scales")

##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpzhXlbh/downloaded_packages

library(tidyverse)
library(knitr)
library(readr)
library(scales)

##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':  
##  
##      discard  
  
## The following object is masked from 'package:readr':  
##  
##      col_factor
```

Import data set

```
furniture_dataset <- read_csv("~/Desktop/Data sets/Furniture.csv")
```

```
## Rows: 2500 Columns: 15  
## — Column specification
```

```
## Delimiter: ","  
## chr (7): category, material, color, location, season, store_type, brand  
## dbl (8): price, cost, sales, profit_margin, inventory,  
discount_percentage, ...  
##  
## 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.
```

Clean data

```
furniture_dataset_clean <- furniture_dataset %>% na.omit()
```

Brief summary

```
colnames(furniture_dataset_clean) #List of column names
```

```
## [1] "price"          "cost"           "sales"  
## [4] "profit_margin" "inventory"       "discount_percentage"  
## [7] "delivery_days"  "category"        "material"  
## [10] "color"          "location"        "season"  
## [13] "store_type"     "brand"           "revenue"
```

```
ncol(furniture_dataset_clean) #How many columns are in data frame?
```

```
## [1] 15
```

```
nrow(furniture_dataset_clean) #How many rows are in data frame?
```

```
## [1] 2500
```

```
dim(furniture_dataset_clean) #Dimensions of the data frame?
```

```
## [1] 2500 15
```

```
head(furniture_dataset_clean) #See the first 6 rows of data frame.
```

```
## # A tibble: 6 × 15
##   price cost sales profit_margin inventory discount_percentage
##   <dbl> <dbl> <dbl>          <dbl>      <dbl>          <dbl>
## 1  219. 182.    40          16.9      105          27.8
9
## 2  478. 385.     7          19.4      192          26.9
6
## 3  379. 277.    32          27.1      59          21.9
2
## 4  319. 282.    48          11.8      45          11.0
2
## 5  120.  69.7    19          42.0      35           3.18
9
## 6  120.  65.4     6          45.6     185          20.7
8
## # i 8 more variables: category <chr>, material <chr>, color <chr>,
## # location <chr>, season <chr>, store_type <chr>, brand <chr>, revenue
<dbl>
```

`str(furniture_dataset_clean)` *#See list of columns and data types (numeric, character, etc)*

```
## tibble [2,500 × 15] (S3: tbl_df/tbl/data.frame)
##  $ price           : num [1:2500] 219 478 379 319 120 ...
##  $ cost            : num [1:2500] 181.6 385 276.7 281.8 69.7 ...
##  $ sales           : num [1:2500] 40 7 32 48 19 6 20 27 43 37 ...
##  $ profit_margin   : num [1:2500] 16.9 19.4 27.1 11.8 42 ...
##  $ inventory       : num [1:2500] 105 192 59 45 35 185 165 10 11 169
...
##  $ discount_percentage: num [1:2500] 27.8 26.94 21.95 11.01 3.18 ...
##  $ delivery_days      : num [1:2500] 9 6 2 2 9 8 9 4 4 2 ...
##  $ category          : chr [1:2500] "Bed" "Chair" "Table" "Table" ...
##  $ material          : chr [1:2500] "Plastic" "Glass" "Metal" "Glass" ...
##  $ color             : chr [1:2500] "Red" "Blue" "Black" "Green" ...
##  $ location          : chr [1:2500] "Rural" "Rural" "Suburban" "Rural"
...
##  $ season            : chr [1:2500] "Spring" "Summer" "Fall" "Summer" ...
##  $ store_type        : chr [1:2500] "Online" "Online" "Online" "Retail"
...
##  $ brand             : chr [1:2500] "BrandA" "BrandD" "BrandD" "BrandD"
...
##  $ revenue           : num [1:2500] 3949 -3521 14286 12261 -4588 ...
```

`summary(furniture_dataset_clean)` *#Statistical summary of data. Mainly for numerics*

```
##      price          cost          sales          profit_margin
##  Min.   : 50.7      Min.   : 26.51      Min.   : 1.00      Min.   :10.02
##  1st Qu.:159.1      1st Qu.:106.40      1st Qu.:13.00      1st Qu.:20.23
```

```
## Median :277.6    Median :189.34    Median :25.00    Median :30.30
## Mean    :274.5    Mean    :191.93    Mean    :24.92    Mean    :30.21
## 3rd Qu.:387.4    3rd Qu.:263.19    3rd Qu.:37.00    3rd Qu.:40.13
## Max.    :499.9    Max.    :447.02    Max.    :49.00    Max.    :50.00
## inventory discount_percentage delivery_days category
## Min.     : 0.00    Min.     : 0.005556    Min.     :1.000    Length:2500
## 1st Qu.: 50.00    1st Qu.: 7.760214    1st Qu.:3.000    Class :character
## Median : 94.00    Median :14.915143    Median :5.000    Mode  :character
## Mean    : 97.72    Mean    :14.947616    Mean     :4.894
## 3rd Qu.:147.00    3rd Qu.:22.292661    3rd Qu.:7.000
## Max.    :199.00    Max.    :29.991229    Max.     :9.000
## material color location season
## Length:2500 Length:2500 Length:2500 Length:2500
## Class :character Class :character Class :character Class :character
## Mode :character Mode :character Mode :character Mode :character
##
##
## store_type brand revenue
## Length:2500 Length:2500 Min. : -14215
## Class :character Class :character 1st Qu.: 1217
## Mode :character Mode :character Median : 5523
## Mean : 5927
## 3rd Qu.: 10234
## Max. : 32922
```

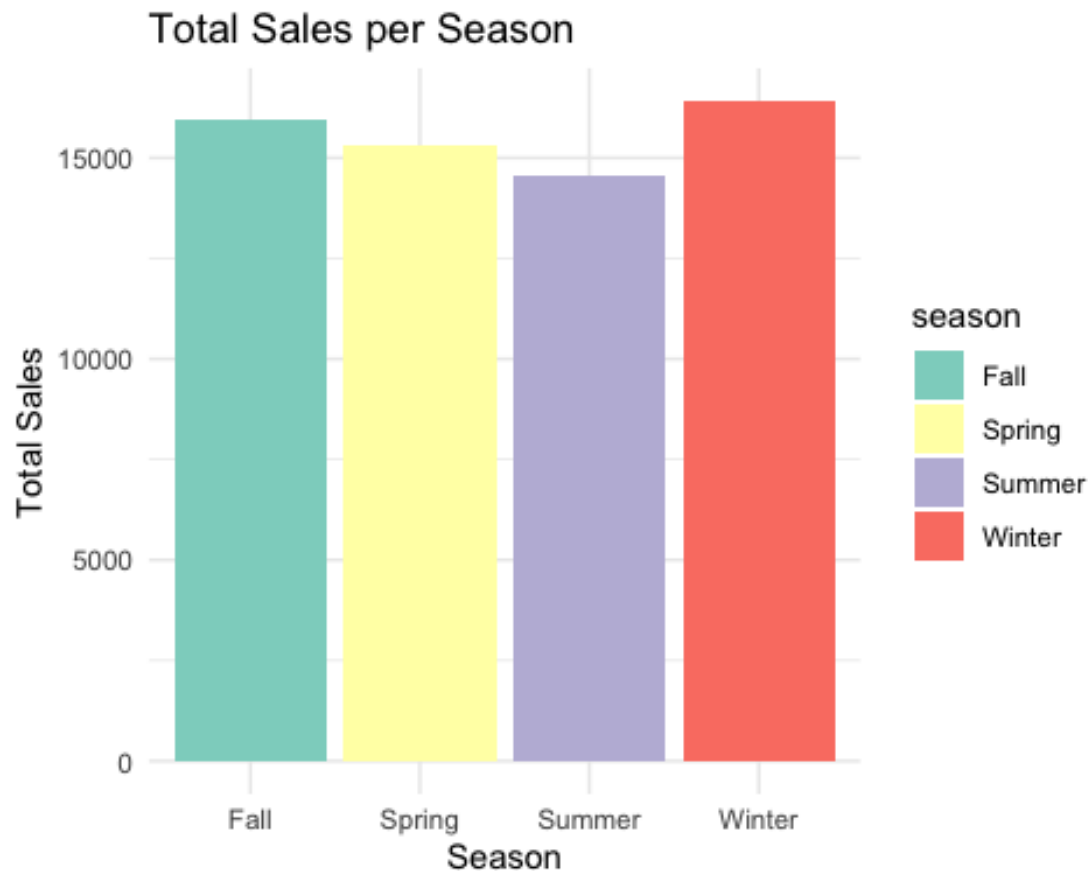
```
names(furniture_dataset_clean)
```

```
## [1] "price"          "cost"           "sales"
## [4] "profit_margin" "inventory"       "discount_percentage"
## [7] "delivery_days"  "category"        "material"
## [10] "color"          "location"        "season"
## [13] "store_type"     "brand"           "revenue"
```

Visualizations

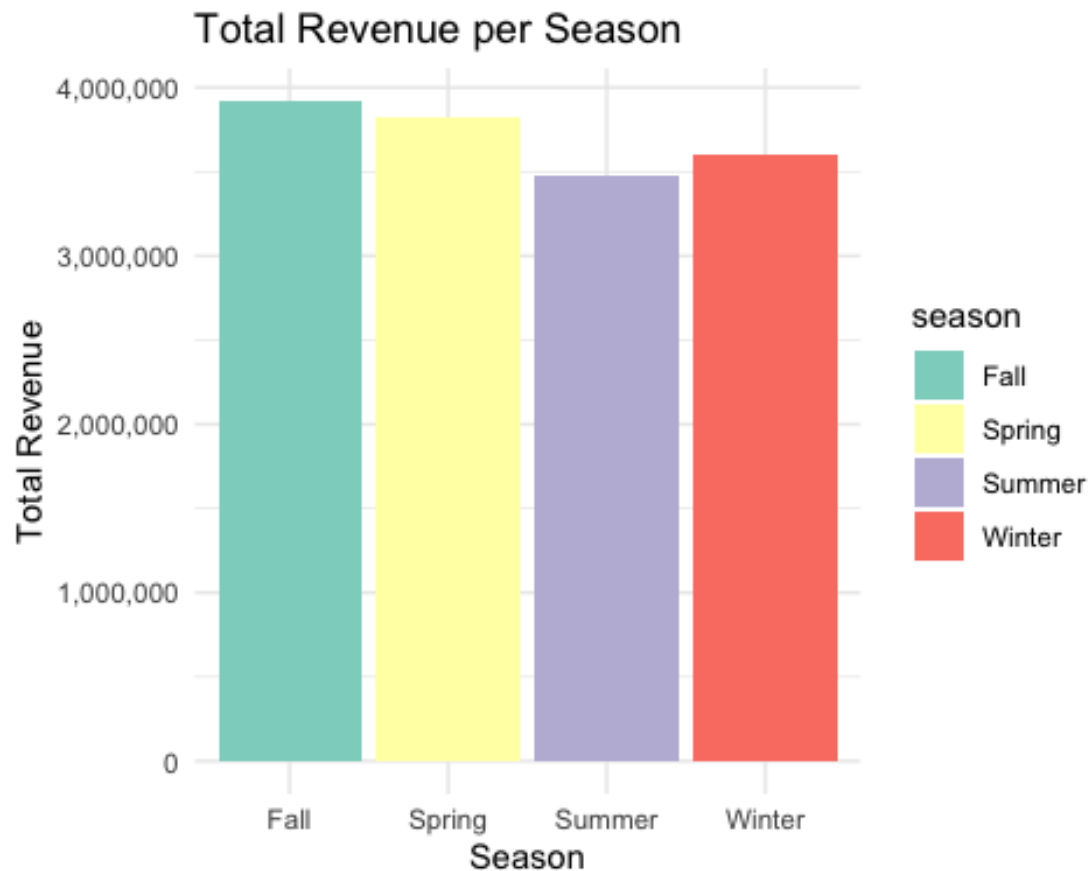
```
# Sales per season
sales_per_season <- furniture_dataset_clean %>%
  group_by(season) %>%
  summarize(total_sales = sum(sales))

ggplot(sales_per_season, aes(x = season, y = total_sales, fill = season)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Sales per Season", x = "Season", y = "Total Sales") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
```



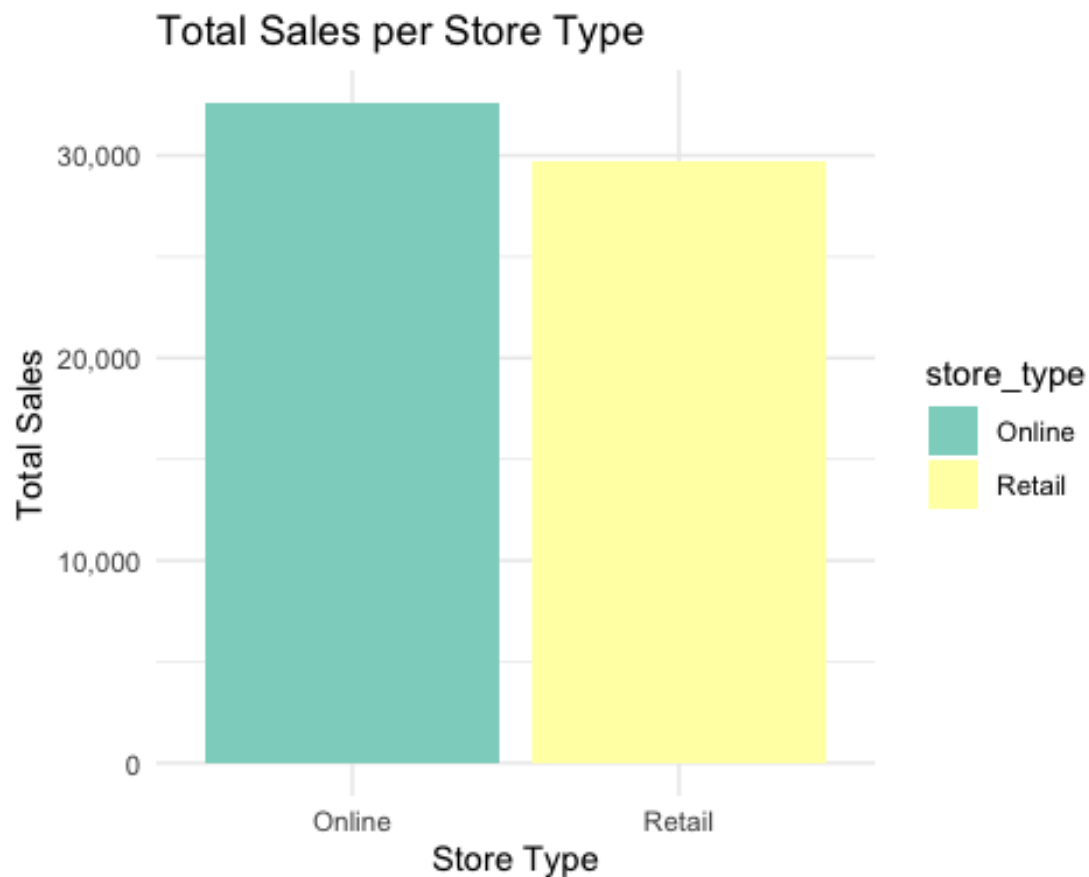
```
# Revenue per season
revenue_per_season <- furniture_dataset_clean %>%
  group_by(season) %>%
  summarize(total_revenue = sum(revenue))

ggplot(revenue_per_season, aes(x = season, y = total_revenue, fill = season))
+
  geom_bar(stat = "identity") +
  labs(title = "Total Revenue per Season", x = "Season", y = "Total Revenue")
+
  theme_minimal() +
  scale_y_continuous(labels = label_comma()) +
  scale_fill_brewer(palette = "Set3")
```



```
# Store type sales
sales_per_store_type <- furniture_dataset_clean %>%
  group_by(store_type) %>%
  summarize(total_sales = sum(sales))

ggplot(sales_per_store_type, aes(x = store_type, y = total_sales, fill =
store_type)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Sales per Store Type", x = "Store Type", y = "Total
Sales") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  scale_fill_brewer(palette = "Set3")
```



```
# Sales by Location
sales_per_location_store_type <- furniture_dataset_clean %>%
  group_by(location, store_type) %>%
  summarize(total_sales = sum(sales))

## `summarise()` has grouped output by 'location'. You can override using the
## `.groups` argument.

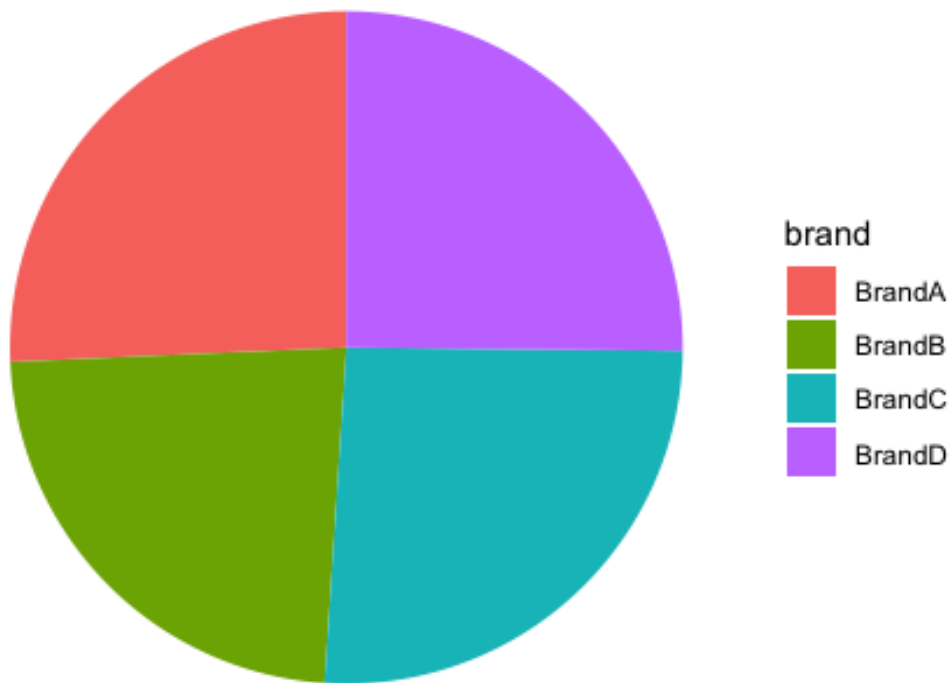
ggplot(sales_per_location_store_type, aes(x = location, y = total_sales, fill
= store_type)) +
  geom_bar(stat = "identity") + # No position = "dodge" for stacking
  labs(title = "Total Sales by Location and Store Type", x = "Location", y =
"Total Sales") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma())
```



```
# Sales per brand
sales_per_brand <- furniture_dataset_clean %>%
  group_by(brand) %>%
  summarize(total_sales = sum(sales))

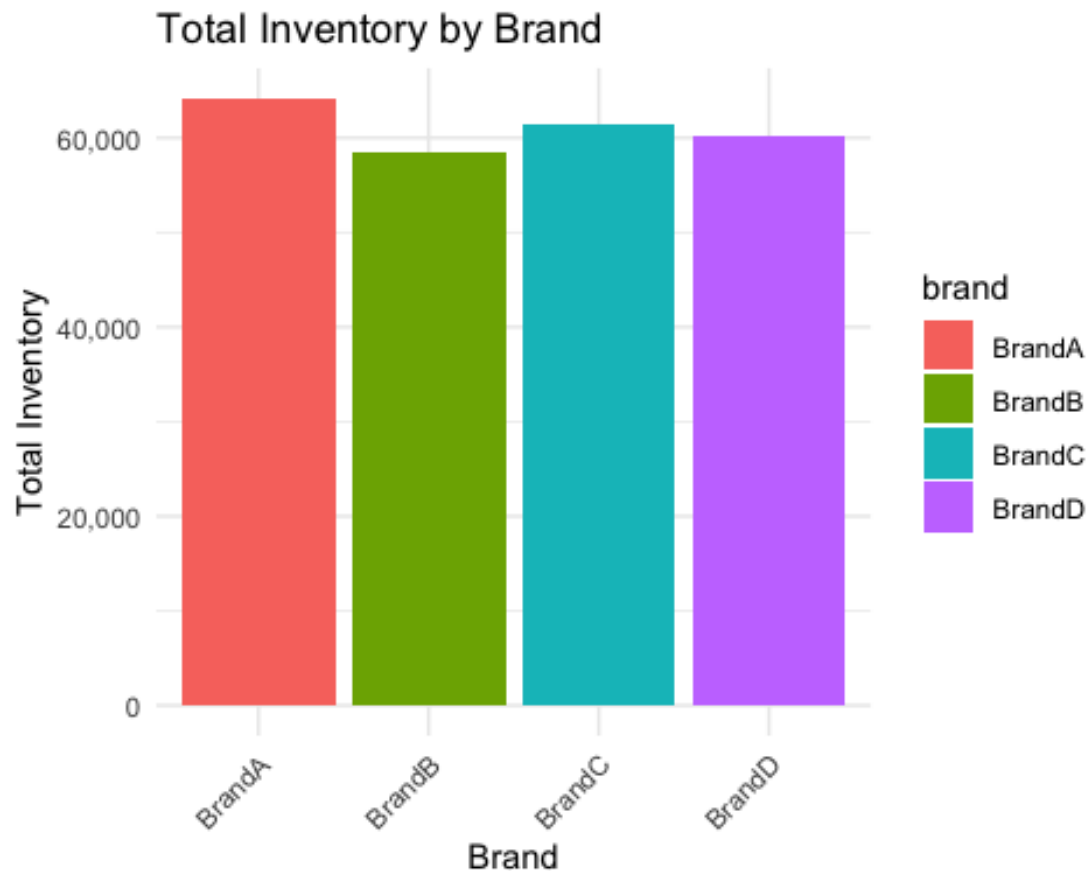
ggplot(sales_per_brand, aes(x = "", y = total_sales, fill = brand)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") + # Convert bar chart to pie chart
  labs(title = "Sales Distribution by Brand") +
  theme_void() +
  scale_y_continuous(labels = scales::label_comma())
```


Sales Distribution by Brand



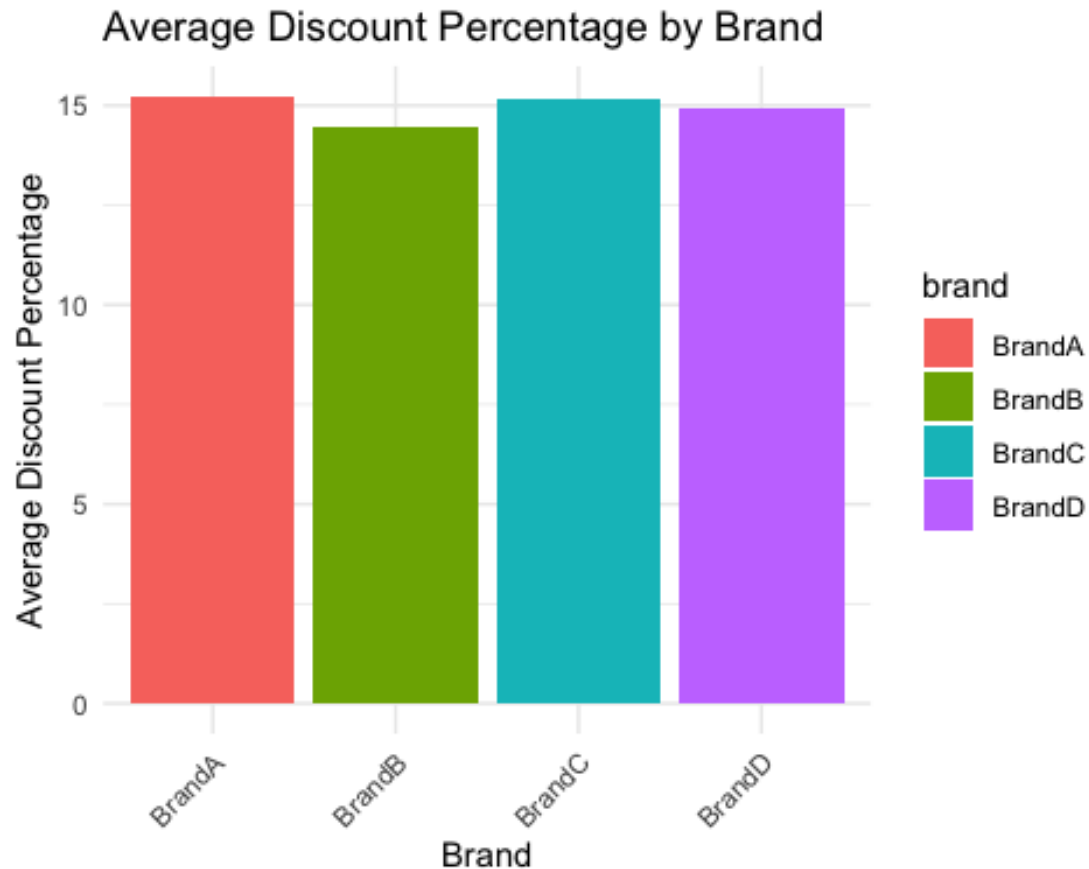
```
# inventory by brand
inventory_per_brand <- furniture_dataset_clean %>%
  group_by(brand) %>%
  summarize(total_inventory = sum(inventory))

ggplot(inventory_per_brand, aes(x = brand, y = total_inventory, fill =
brand)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Inventory by Brand", x = "Brand", y = "Total
Inventory") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



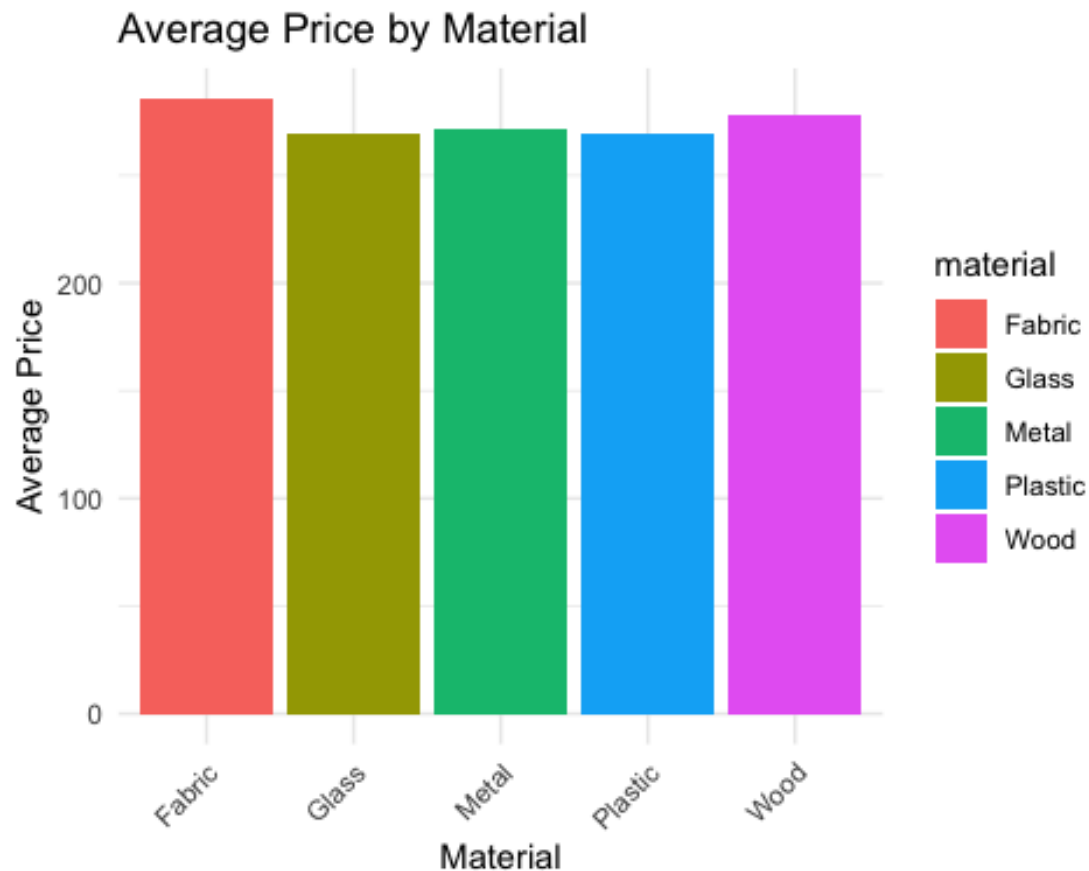
```
# Discount by brand
discount_per_brand <- furniture_dataset_clean %>%
  group_by(brand) %>%
  summarize(average_discount_percentage = mean(discount_percentage))

ggplot(discount_per_brand, aes(x = brand, y = average_discount_percentage,
fill = brand)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Discount Percentage by Brand", x = "Brand", y =
"Average Discount Percentage") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



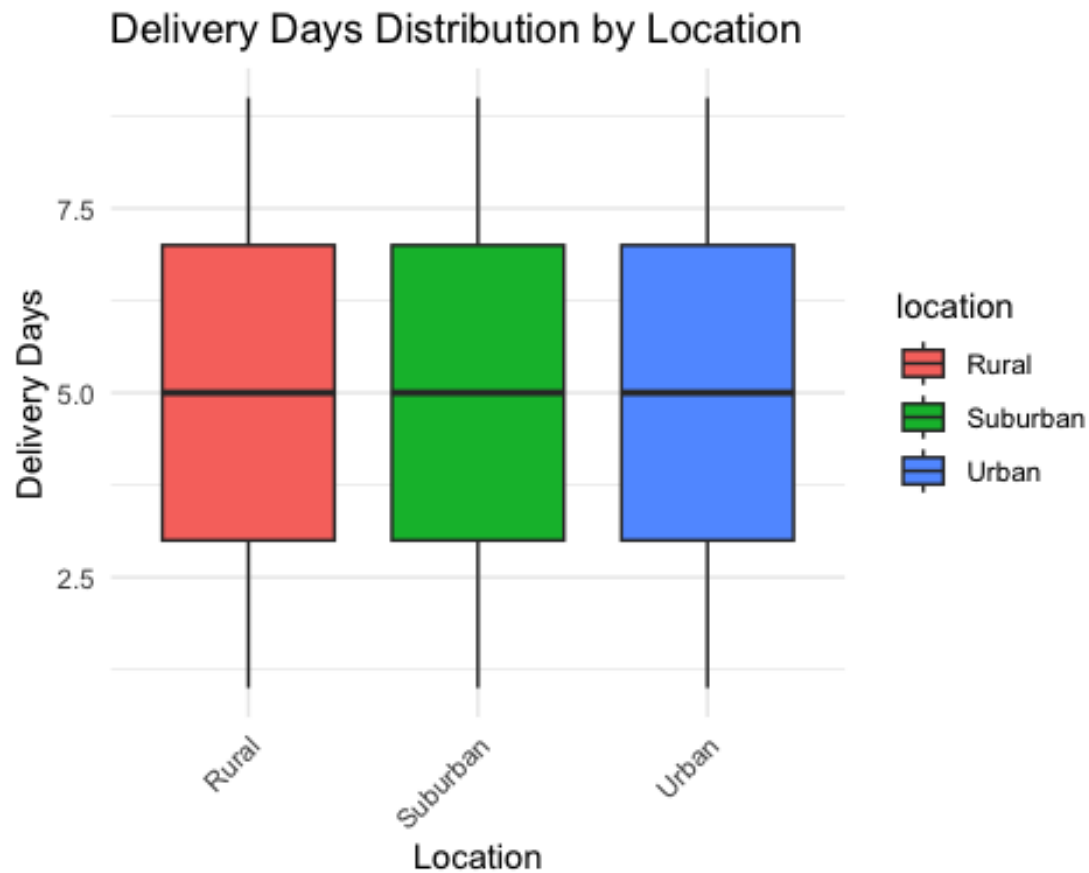
```
# Price by material
price_per_material <- furniture_dataset_clean %>%
  group_by(material) %>%
  summarize(average_price = mean(price))

ggplot(price_per_material, aes(x = material, y = average_price, fill =
material)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Price by Material", x = "Material", y = "Average
Price") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

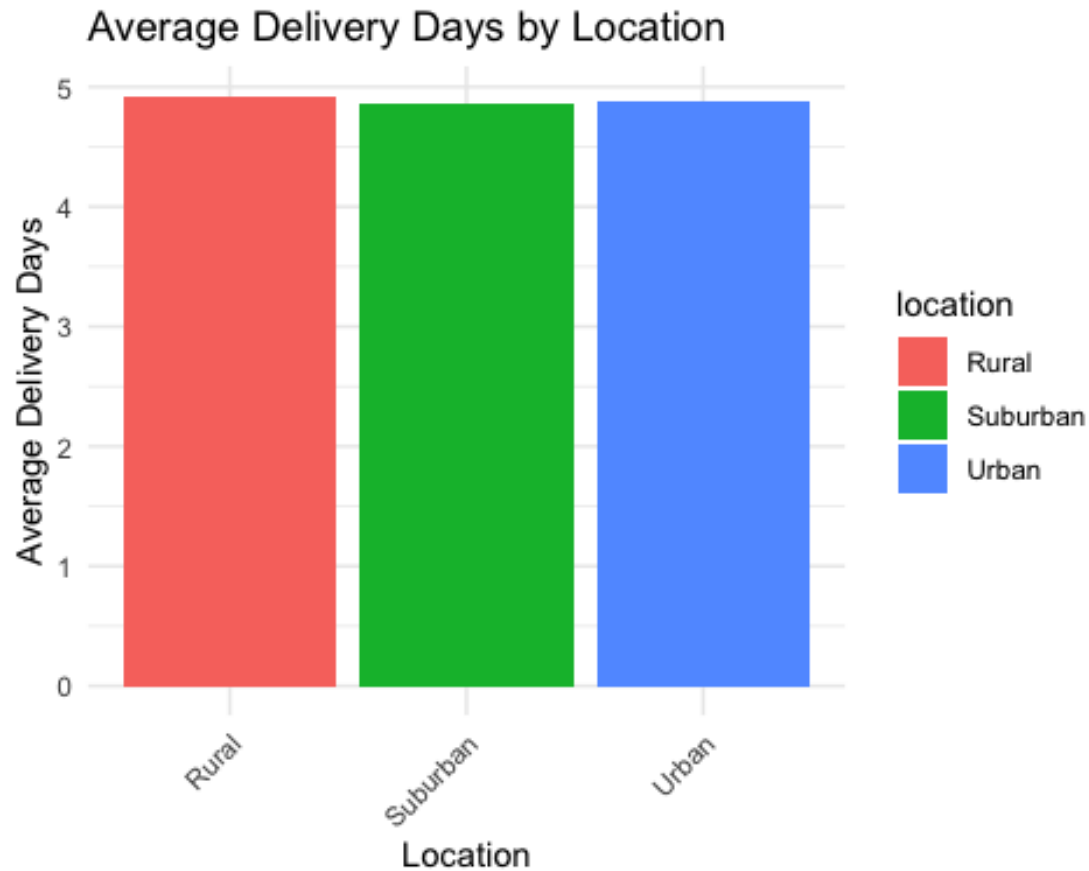


```
#Average delivery days by Location
delivery_days_per_location <- furniture_dataset_clean %>%
  group_by(location) %>%
  summarize(average_delivery_days = mean(delivery_days))

ggplot(furniture_dataset_clean, aes(x = location, y = delivery_days, fill =
location)) +
  geom_boxplot() +
  labs(title = "Delivery Days Distribution by Location", x = "Location", y =
"Delivery Days") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

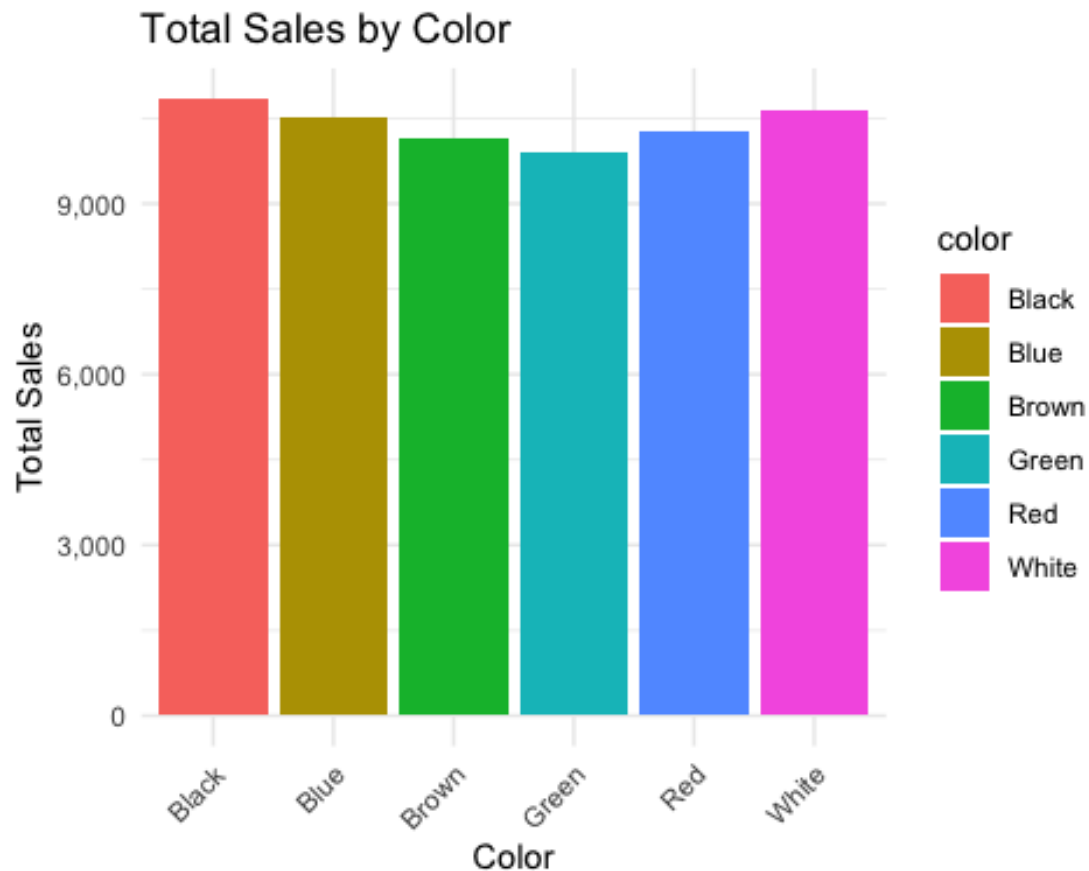


```
ggplot(delivery_days_per_location, aes(x = location, y =  
average_delivery_days, fill = location)) +  
  geom_bar(stat = "identity") +  
  labs(title = "Average Delivery Days by Location", x = "Location", y =  
"Average Delivery Days") +  
  theme_minimal() +  
  scale_y_continuous(labels = scales::label_comma()) +  
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



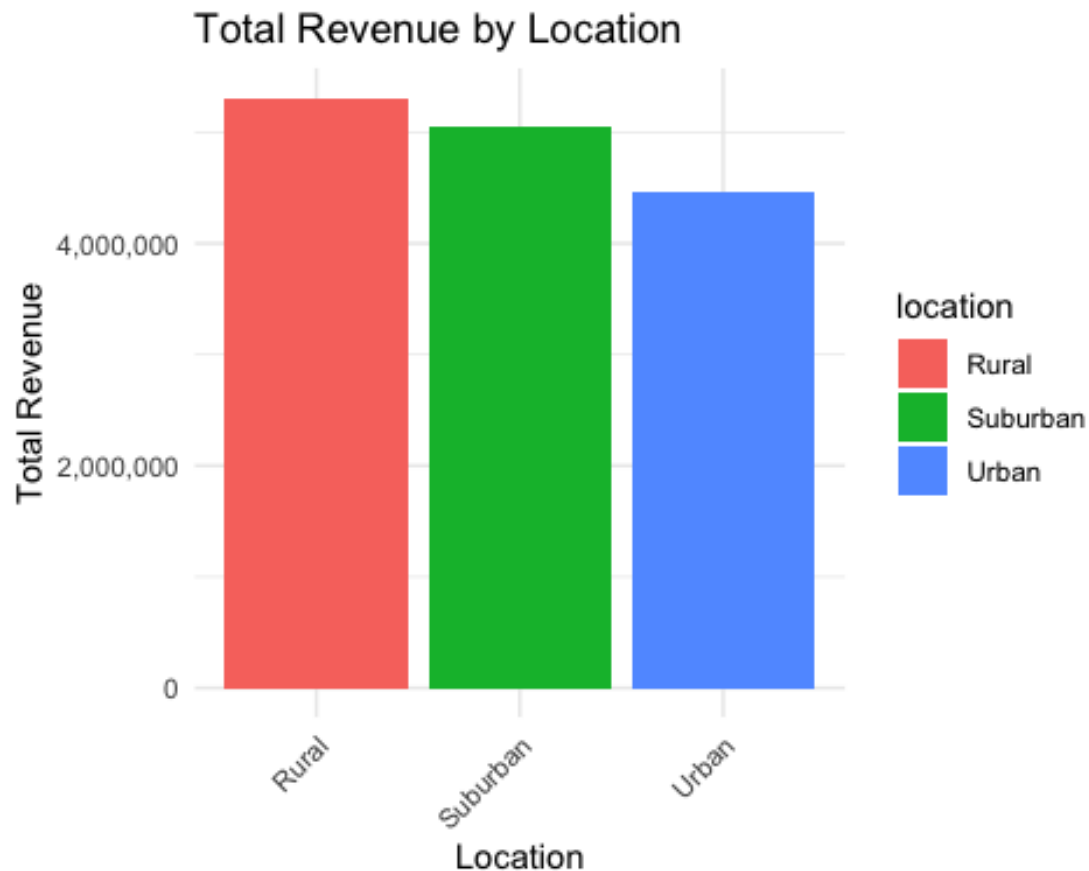
```
# Sales by color
sales_per_color <- furniture_dataset_clean %>%
  group_by(color) %>%
  summarize(total_sales = sum(sales))

ggplot(sales_per_color, aes(x = color, y = total_sales, fill = color)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Sales by Color", x = "Color", y = "Total Sales") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



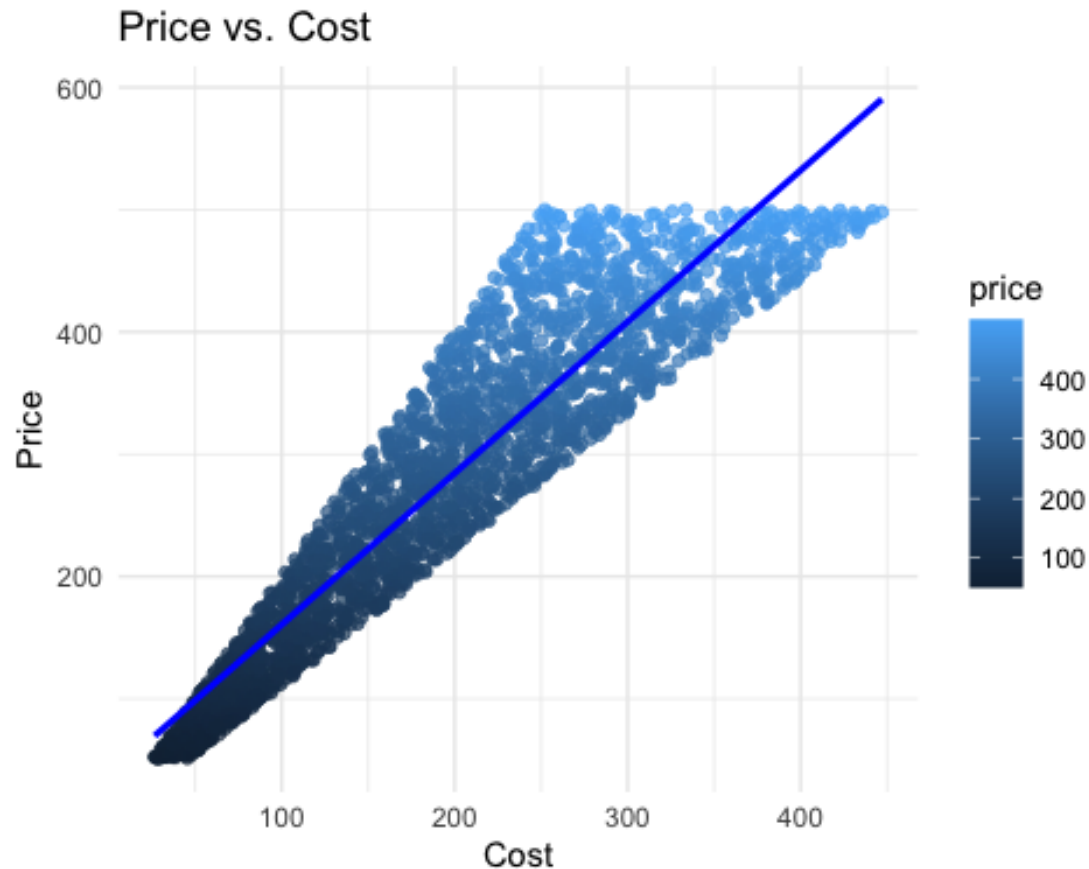
```
# Revenue by Location
revenue_per_location <- furniture_dataset_clean %>%
  group_by(location) %>%
  summarize(total_revenue = sum(revenue))

ggplot(revenue_per_location, aes(x = location, y = total_revenue, fill =
location)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Revenue by Location", x = "Location", y = "Total
Revenue") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# Price vs Cost scatterplot with trend line
ggplot(furniture_dataset_clean, aes(x = cost, y = price)) +
  geom_point(aes(color = price), alpha = 0.7) +
  geom_smooth(method = "lm", color = "blue", se = FALSE) +
  labs(title = "Price vs. Cost", x = "Cost", y = "Price") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  scale_x_continuous(labels = scales::label_comma())

## `geom_smooth()` using formula = 'y ~ x'
```

```
# Price vs Cost by Brand with trend Line
ggplot(furniture_dataset_clean, aes(x = cost, y = price, color = brand)) +
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", aes(group = brand), se = FALSE) +
  labs(title = "Price vs. Cost by Brand", x = "Cost", y = "Price") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  scale_x_continuous(labels = scales::label_comma()) +
  theme(legend.position = "right")

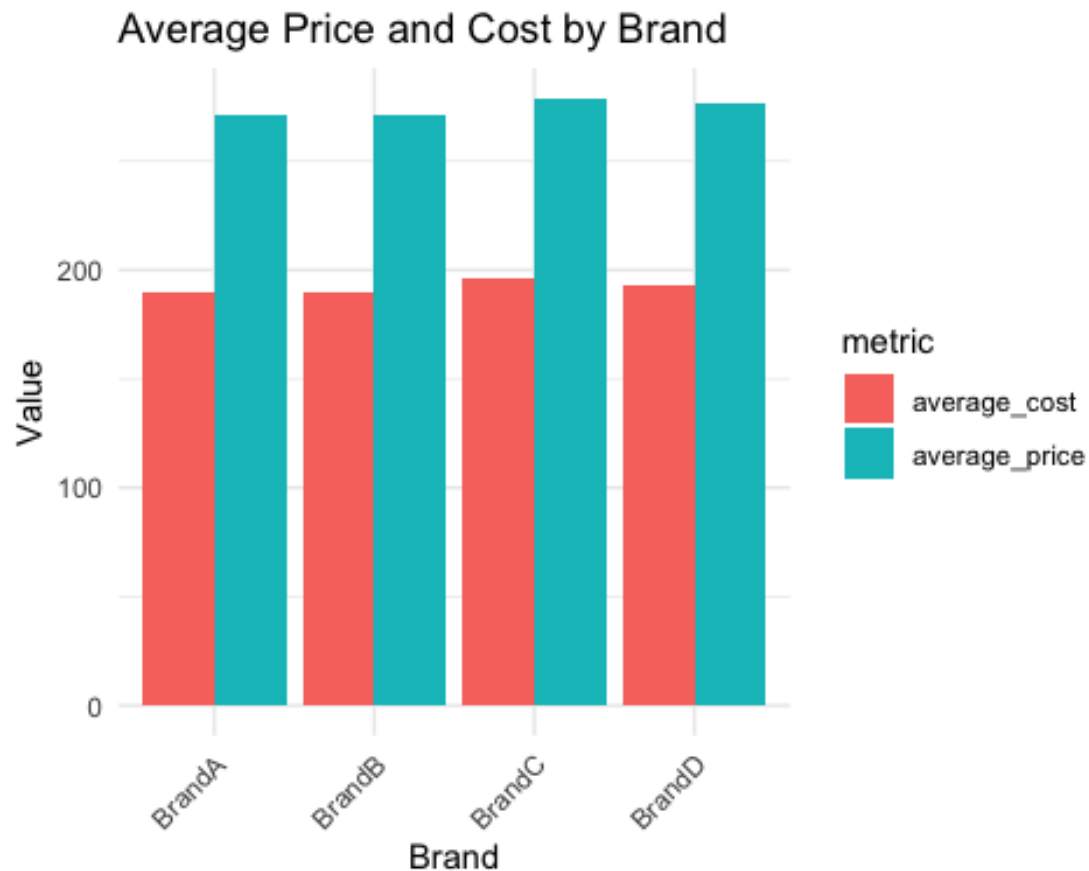
## `geom_smooth()` using formula = 'y ~ x'
```



```
# Average price and cost by brand
averages_by_brand <- furniture_dataset_clean %>%
  group_by(brand) %>%
  summarize(
    average_price = mean(price),
    average_cost = mean(cost)
  )

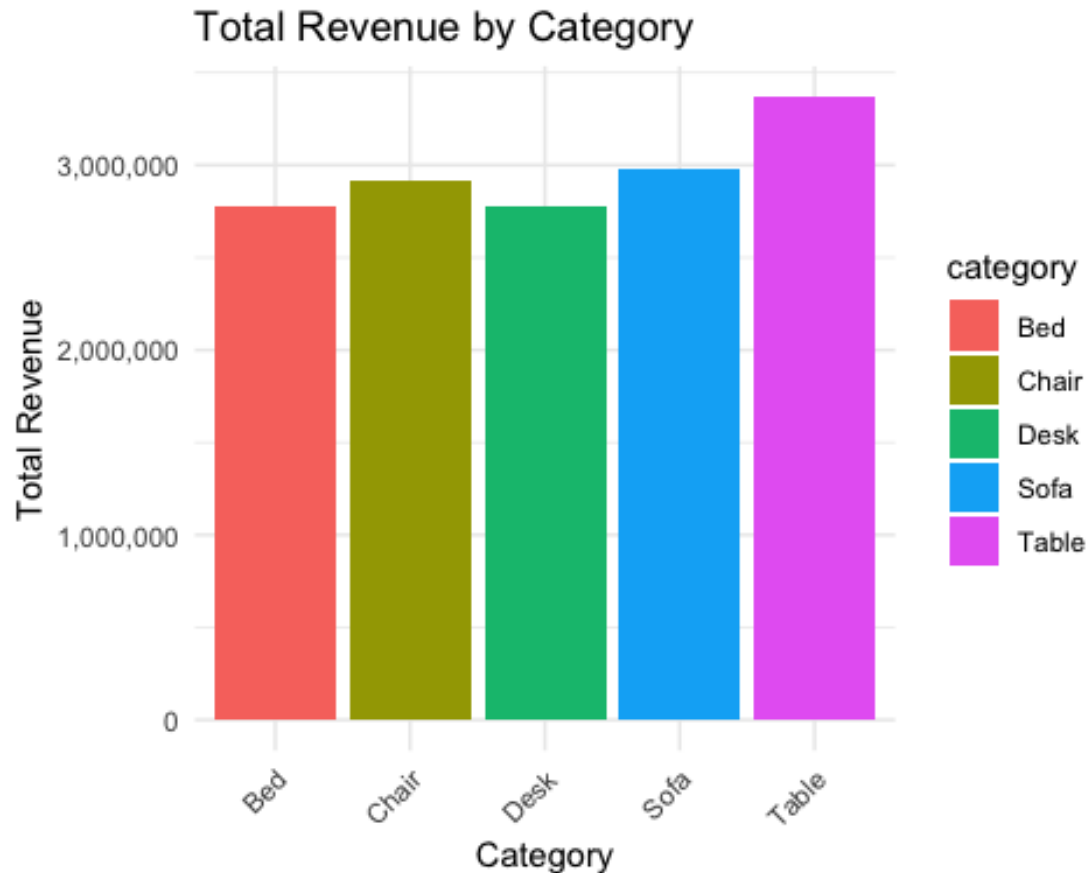
averages_long <- averages_by_brand %>%
  pivot_longer(cols = c(average_price, average_cost), names_to = "metric",
    values_to = "value")

ggplot(averages_long, aes(x = brand, y = value, fill = metric)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Average Price and Cost by Brand", x = "Brand", y = "Value") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# Revenue by category
revenue_by_category <- furniture_dataset_clean %>%
  group_by(category) %>%
  summarize(total_revenue = sum(revenue))

ggplot(revenue_by_category, aes(x = category, y = total_revenue, fill =
category)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Revenue by Category", x = "Category", y = "Total
Revenue") +
  theme_minimal() +
  scale_y_continuous(labels = scales::label_comma()) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



#Conclusion: Looking at the data, Fall and Winter are the most popular seasons for total sales of furniture; however, for total revenue, Winter and Spring are the most successful seasons. The company's online store is more popular for total sales than its brick-and-mortar counterpart. The company's rural stores make more in-person and online sales than its suburban and urban stores, and the rural store has the highest revenue. Brand C is the most popular brand across the stores, with Brand A closely behind it and Brand B being the least popular. This could likely be due to Brand A and Brand C having the most discounts per brand. Brand A is also the most profitable for the store, as the price revenue analysis is detailed. According to the data, fabric is the most popular material for furniture, and Black and White furniture are the most popular colors. The table is the most popular type of furniture at the store. The delivery days for the location of the stores (Rural, suburban, and urban) are all relatively the same, with urban stores taking slightly longer.