



## 1.3.3\_dplyr Exercises Solutions

### Business Examples using the Superstore Dataset

```
# use a separate cell to load the library to accommodate for the eval=FALSE
library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(readr)

file_path <- "data\\superstore.csv"
superstore <- read_csv(file_path, show_col_types = FALSE)
```

### 1. Filter the Superstore dataset for "Furniture" sales in the "West" region.

```
furniture_sales_west <- superstore %>%
  filter(Category == "Furniture", Region == "West")

print(furniture_sales_west)
```

# A tibble: 707 × 21

	`Row ID`	`Order ID`	`Order Date`	`Ship Date`	`Ship Mode`	`Customer ID`
	<dbl>	<chr>	<chr>	<chr>	<chr>	<chr>
1	6	CA-2014-115812	6/9/2014	6/14/2014	Standard Class	BH-11710
2	11	CA-2014-115812	6/9/2014	6/14/2014	Standard Class	BH-11710
3	25	CA-2015-106320	9/25/2015	9/30/2015	Standard Class	EB-13870
4	66	CA-2015-135545	11/24/2015	11/30/2015	Standard Class	KM-16720
5	105	US-2015-156867	11/13/2015	11/17/2015	Standard Class	LC-16870
6	118	CA-2015-110457	3/2/2015	3/6/2015	Standard Class	DK-13090
7	129	US-2016-125969	11/6/2016	11/10/2016	Second Class	LS-16975
8	130	US-2016-125969	11/6/2016	11/10/2016	Second Class	LS-16975
9	140	CA-2016-145583	10/13/2016	10/19/2016	Standard Class	LC-16885
10	158	CA-2014-104269	3/1/2014	3/6/2014	Second Class	DB-13060

```
# i 697 more rows
# i 15 more variables: `Customer Name` <chr>, Segment <chr>, Country <chr>,
#   City <chr>, State <chr>, `Postal Code` <dbl>, Region <chr>,
#   `Product ID` <chr>, Category <chr>, `Sub-Category` <chr>,
#   `Product Name` <chr>, Sales <dbl>, Quantity <dbl>, Discount <dbl>,
#   Profit <dbl>
```

## 2. Select the “Customer Name”, “State”, and “Sales” columns and arrange them in descending order of sales.

```
furniture_sales_west_selected <- furniture_sales_west %>%
  select(`Customer Name`, `State`, `Sales`) %>%
  arrange(desc(Sales))

print(furniture_sales_west_selected)
```

```
# A tibble: 707 × 3
  `Customer Name`      State      Sales
  <chr>              <chr>    <dbl>
1 Nora Preis          California 3611.
2 Maribeth Schnelling California 3407.
3 Ellis Ballard        Washington 3394.
4 Scot Coram           California 2887.
5 Sean Christensen     California 2804.
6 Lindsay Castell       California 2677.
7 Sarah Foster         Washington 2666.
8 Ted Trevino           Washington 2405.
9 Paul Prost           Washington 2349.
10 Karen Ferguson      California 2054.
# i 697 more rows
```

## 3. Calculate the average profit per sub-category and arrange the results in ascending order.

Here, we are considering the entire Superstore dataset, not just the “Furniture” category or “West” region:

```
average_profit_per_sub_category <- superstore %>%
  group_by(`Sub-Category`) %>%
  summarise(Average_Profit = mean(Profit)) %>%
  arrange(Average_Profit)

print(average_profit_per_sub_category)
```

```
# A tibble: 17 × 2
  `Sub-Category` Average_Profit
  <chr>          <dbl>
1 Tables        -55.6
2 Bookcases     -15.2
3 Supplies      -6.26
```

4 Fasteners	4.38
5 Art	8.20
6 Furnishings	13.6
7 Labels	15.2
8 Binders	19.8
9 Paper	24.9
10 Storage	25.2
11 Envelopes	27.4
12 Machines	29.4
13 Appliances	38.9
14 Chairs	43.4
15 Phones	50.1
16 Accessories	54.1
17 Copiers	818.

## 4. Identify Highest Selling Products in Furniture Category

**Question:** What are the top 5 highest selling products in the Furniture category?

```
library(dplyr)

# Filter for Furniture category
furniture_sales <- superstore %>%
  filter(`Category` == "Furniture")

# Find top 5 products by Sales in Furniture category
top_furniture_products <- furniture_sales %>%
  group_by(`Product Name`) %>%
  summarise(Total_Sales = sum(`Sales`)) %>%
  arrange(desc(Total_Sales)) %>%
  slice_head(n = 5)

print(top_furniture_products)
```

```
# A tibble: 5 × 2
  `Product Name`                Total_Sales
  <chr>                  <dbl>
1 HON 5400 Series Task Chairs for Big and Tall      21871.
2 Riverside Palais Royal Lawyers Bookcase, Royale Cherry Finish 15611.
3 Bretford Rectangular Conference Table Tops        12995.
4 Global Troy Executive Leather Low-Back Tilter      12975.
5 SAFCO Arco Folding Chair                          11573.
```

## 5. Yearly Sales Trend for Technology in the East Region

**Question:** What is the yearly sales trend for Technology products in the East region?

To solve this question we need to use `lubridate` package.

**lubridate** is an R package that makes it easier to work with dates and times. It provides functions to parse various date-time formats, extract parts of dates or times, and perform calculations with date-time objects. With **lubridate**, you can add or subtract time periods from dates, handle time zones, and efficiently compare dates. Its functions are designed to simplify common use cases for date-time data and integrate well with the **tidyverse** set of packages. It is a go-to choice in R for date-time manipulation due to its user-friendly and intuitive syntax.

```
library(lubridate)
```

Attaching package: 'lubridate'

The following objects are masked from 'package:base':

date, intersect, setdiff, union

```
# Filter for Technology category and East region
tech_sales_east <- superstore %>%
  filter(`Category` == "Technology", `Region` == "East")

# Convert Order Date to Date type and extract the year
tech_sales_east <- tech_sales_east %>%
  mutate(Year = year(mdy(`Order Date`)))

# Calculate yearly sales trend for Technology in East region
yearly_sales_trend <- tech_sales_east %>%
  group_by(Year) %>%
  summarise(Total_Sales = sum(`Sales`)) %>%
  arrange(Year)

print(yearly_sales_trend)
```

```
# A tibble: 4 × 2
  Year Total_Sales
  <dbl>      <dbl>
1  2014      45479.
2  2015      59859.
3  2016      72497.
4  2017      87138.
```

## 6. Average Discount Impact on Profits Across Segments

**Question:** How does the average discount given affect profits across different customer segments?

```
# Calculate average discount and profit per segment
discount_impact <- superstore %>%
  group_by(`Segment`) %>%
  summarise(Average_Discount = mean(`Discount`),
            Average_Profit = mean(`Profit`))
```

```
print(discount_impact)
```

```
# A tibble: 3 × 3
```

	Segment	Average_Discount	Average_Profit
	<chr>	<dbl>	<dbl>
1	Consumer	0.158	25.9
2	Corporate	0.158	30.4
3	Home Office	0.147	33.8

## 7. State-wise Profitability for Office Supplies

**Question:** Which states are the most and least profitable for Office Supplies?

```
# Filter for Office Supplies category
office_supplies_sales <- superstore %>%
  filter(Category == "Office Supplies")

# Calculate profit by state for Office Supplies
state_profitability <- office_supplies_sales %>%
  group_by(State) %>%
  summarise(Total_Profit = sum(Profit)) %>%
  arrange(desc(Total_Profit))

# View the most profitable states
most_profitable_states <- head(state_profitability, n = 5)
print(most_profitable_states)
```

```
# A tibble: 5 × 2
```

	State	Total_Profit
	<chr>	<dbl>
1	California	37755.
2	New York	25994.
3	Michigan	15005.
4	Washington	11190.
5	Georgia	9801.

```
# View the least profitable states
least_profitable_states <- tail(state_profitability, n = 5)

print(least_profitable_states)
```

```
# A tibble: 5 × 2
```

	State	Total_Profit
	<chr>	<dbl>
1	Florida	-1675.
2	Tennessee	-3199.
3	Pennsylvania	-5172.

```
4 Illinois      -8354.
5 Texas        -18585.
```

```
# Piping it all together
superstore %>%
  filter(`Category` == "Office Supplies") %>%
  group_by(`State`) %>%
  summarise(Total_Profit = sum(`Profit`)) %>%
  arrange(desc(Total_Profit))%>% head(state_profitability, n = 5)
```

```
# A tibble: 5 × 2
  State      Total_Profit
  <chr>      <dbl>
1 California 37755.
2 New York   25994.
3 Michigan   15005.
4 Washington 11190.
5 Georgia    9801.
```

## 8. Create a new column that represents the sales as a percentage of total sales within each category.

For this, we will first calculate the total sales within the "Furniture" category and then compute the percentage for each sale.

```
library(dplyr)

modified_superstore <- superstore %>%
  group_by(Category) %>%
  mutate(TotalSales = sum(Sales), # Calculate total sales per category
         SalesPercentage = (Sales / TotalSales) * 100) %>%
  ungroup() # Remove grouping

View(modified_superstore)
```

The code adds a column that shows sales as a percentage of the total sales within each category using **dplyr** from the **tidyverse**.

When you use **mutate()** on a grouped data frame in **dplyr**, it will preserve the grouping. However, this does not mean the data is permanently grouped within the resultant data frame. Instead, the grouping metadata is kept so that further **dplyr** operations can take advantage of it.

The **ungroup()** function is used to explicitly remove this grouping metadata from the data frame. While it's not always necessary to include **ungroup()** at the end of your operation, it can be a good practice when you want to ensure that the returned data frame doesn't carry any group structure, especially before performing operations that should not be influenced by the previous grouping.

If you don't call `ungroup()`, the returned data frame will still contain the grouping information, which might affect how subsequent functions work with the data frame. For example, if you pass this data frame to another `summarise()` call without removing the grouping, it will perform the summarisation by the existing groups.

However, when you just print the data frame or write it to a file, it will appear ungrouped. The grouping only affects how `dplyr` handles the data for further operations. If no further grouped operations are intended, then whether you ungroup the data or not may not have any visible impact on the output, but it's generally a good practice to ungroup a data frame if you've finished with group-specific operations to avoid unexpected results in subsequent analyses.

## 9. Correlation between Sales and Profit in Sub-Categories

**Question:** Is there a correlation between sales and profit within each sub-category?

```
# Calculate correlation between Sales and Profit for each Sub-Category
sales_profit_correlation <- superstore %>%
  group_by(`Sub-Category`) %>%
  summarise(Correlation = cor(`Sales`, `Profit`))

print(sales_profit_correlation)
```

```
# A tibble: 17 × 2
  `Sub-Category` Correlation
  <chr>          <dbl>
1 Accessories    0.884
2 Appliances     0.751
3 Art            0.862
4 Binders        0.652
5 Bookcases      0.132
6 Chairs         0.550
7 Copiers        0.966
8 Envelopes      0.981
9 Fasteners      0.679
10 Furnishings   0.468
11 Labels        0.991
12 Machines     -0.00322
13 Paper         0.986
14 Phones        0.657
15 Storage       0.582
16 Supplies     -0.221
17 Tables       -0.0270
```

Before running the above examples, make sure to have the necessary libraries loaded (`dplyr`, `lubridate` for date operations) and that your data is properly formatted (e.g., `Order Date` is in the correct date format). The examples provided assume that the data types and formats are correct and that the Superstore dataset is named `superstore` in your R environment.

## More Business Questions

Presented below are additional business-oriented questions that require a combination of data manipulation and analysis skills.

These questions require that you apply **dplyr** verbs creatively and think critically about business strategies and outcomes.

### 1- Customer Segmentation Analysis

**Question:** How do sales and profits differ among different customer segments (Consumer, Corporate, Home Office)?

Which segment is the most profitable, and why might that be the case?

#### Solution and Explanation:

To analyze sales and profits across different customer segments, we group the data by the **Segment** column and then summarize it to calculate total sales, total profit, and average profit for each segment. This provides insights into which customer segment is most profitable and hints at possible reasons behind the differences in profitability.

```
# Load necessary library
library(dplyr)

# Analyze sales and profits by customer segment
customer_segment_analysis <- superstore %>%
  group_by(Segment) %>%
  summarise(Total_Sales = sum(Sales), Total_Profit = sum(Profit),
            Average_Profit = mean(Profit)) %>%
  mutate(
    Total_Sales = round(Total_Sales, 2),
    Total_Profit = round(Total_Profit, 2),
    Average_Profit = round(Average_Profit, 2)
  ) %>%
  arrange(desc(Total_Profit))

# View the results
print(customer_segment_analysis)
```

# A tibble: 3 × 4

Segment	Total_Sales	Total_Profit	Average_Profit
<chr>	<dbl>	<dbl>	<dbl>
1 Consumer	1162133.	134339.	25.9
2 Corporate	706161.	91986.	30.4
3 Home Office	429653.	60299.	33.8

### 2- Product Performance Evaluation



**Question:** Identify the top 5 and bottom 5 products based on profit margins. Discuss potential reasons for their performance and suggest strategies for improving sales of the less profitable products.

### Solution and Explanation:

To identify products by profitability, we first calculate the profit margin for each product. Then, we rank these products to find the top 5 and bottom 5 based on their profit margins. This analysis can reveal which products are performing well and which are not, providing a basis for strategic adjustments.

```
# Calculate profit margin and identify top and bottom 5 products
product_performance_evaluation <- superstore %>%
  mutate(Profit_Margin = Profit / Sales) %>%
  arrange(desc(Profit_Margin)) %>%
  mutate(Rank = row_number()) %>%
  filter(Rank <= 5 | Rank > (n() - 5))

# View the results
print(product_performance_evaluation)
```

# A tibble: 10 × 23

	`Row ID` <dbl>	`Order ID` <chr>	`Order Date` <chr>	`Ship Date` <chr>	`Ship Mode` <chr>	`Customer ID` <chr>
1	62	CA-2016-111682	6/17/2016	6/18/2016	First Class	TB-21055
2	247	CA-2014-131926	6/1/2014	6/6/2014	Second Class	DW-13480
3	319	CA-2014-164973	11/4/2014	11/9/2014	Standard Class	NM-18445
4	323	CA-2014-156601	9/19/2014	9/24/2014	Standard Class	FA-14230
5	382	CA-2016-134775	10/28/2016	10/29/2016	First Class	AS-10285
6	9087	CA-2016-143406	9/26/2016	9/30/2016	Standard Class	LR-17035
7	262	US-2017-155299	6/8/2017	6/12/2017	Standard Class	DL-13600
8	677	US-2017-119438	3/18/2017	3/23/2017	Standard Class	CD-11980
9	8767	CA-2015-107083	11/21/2015	11/27/2015	Standard Class	BB-11545
10	9165	CA-2015-164007	6/8/2015	6/12/2015	Standard Class	MG-17695

# i 17 more variables: `Customer Name` <chr>, Segment <chr>, Country <chr>,  
 # City <chr>, State <chr>, `Postal Code` <dbl>, Region <chr>,  
 # `Product ID` <chr>, Category <chr>, `Sub-Category` <chr>,  
 # `Product Name` <chr>, Sales <dbl>, Quantity <dbl>, Discount <dbl>,  
 # Profit <dbl>, Profit\_Margin <dbl>, Rank <int>

## 3- Regional Sales Analysis

**Question:** Compare the total sales and profits across different regions. Are there regions that underperform? What strategies could be implemented to boost sales in those regions?

### Solution and Explanation:

This analysis involves grouping the data by region and summarizing it to calculate total sales and profits for each region. By comparing these figures, we can identify which regions might be underperforming and consider strategies to improve their performance.

```
# Analyze sales and profits by region
regional_sales_analysis <- superstore %>%
  group_by(Region) %>%
  summarise(Total_Sales = sum(Sales), `Total Profit` = sum(Profit)) %>%
  arrange(desc(`Total Profit`))

# View the results
print(regional_sales_analysis)
```

# A tibble: 4 × 3

	Region	Total_Sales	`Total Profit`
	<chr>	<dbl>	<dbl>
1	West	725472.	108425.
2	East	678781.	91523.
3	South	392454.	46969.
4	Central	501240.	39706.

## 4- Time Series Analysis

**Question:** How have sales and profits trended over time? Is there seasonality in sales? Identify the peak and low sales periods throughout the year.

### Solution and Explanation:

For this analysis, we first need to convert the **Order Date** to a Date type. We then group the data by month and year to analyze trends and seasonality in sales and profits over time.

```
# Convert Order Date to Date type and perform time series analysis
time_series_analysis <- superstore %>%
  mutate(`Order Date` = as.Date(`Order Date`, format = "%m/%d/%Y")) %>%
  arrange(`Order Date`) %>%
  group_by(Month = format(`Order Date`, "%Y-%m")) %>%
  summarise(Total_Sales = sum(Sales), `Total Profit` = sum(Profit))

print(time_series_analysis)
```

# A tibble: 48 × 3

	Month	Total_Sales	`Total Profit`
	<chr>	<dbl>	<dbl>
1	2014-01	14237.	2450.
2	2014-02	4520.	862.
3	2014-03	55691.	499.
4	2014-04	28295.	3489.
5	2014-05	23648.	2739.
6	2014-06	34595.	4977.
7	2014-07	33946.	-841.
8	2014-08	27909.	5318.
9	2014-09	81777.	8328.

```
10 2014-10      31453.      3448.
# i 38 more rows
```

[View\(time\\_series\\_analysis\)](#)

## 5- Discount Impact Analysis

**Question:** Analyze the impact of discounts on sales volume and profitability. Is there a discount threshold that maximizes profit without significantly hurting sales?

### Solution and Explanation:

To explore the relationship between discounts, sales, and profitability, we group the data by discount level. Then, we calculate the average sales and profit for each discount level to understand how discounts affect profitability and sales volume.

```
# Analyze the impact of discounts on sales and profit
discount_impact_analysis <- superstore %>%
  group_by(Discount) %>%
  summarise(Average_Sales = mean(Sales), Average_Profit = mean(Profit)) %>%
  arrange(Discount)

# View the results
print(discount_impact_analysis)
```

# A tibble: 12 × 3

	Discount	Average_Sales	Average_Profit
	<dbl>	<dbl>	<dbl>
1	0	227.	66.9
2	0.1	578.	96.1
3	0.15	530.	27.3
4	0.2	209.	24.7
5	0.3	455.	-45.7
6	0.32	537.	-88.6
7	0.4	565.	-112.
8	0.45	499.	-227.
9	0.5	893.	-311.
10	0.6	48.2	-43.1
11	0.7	97.2	-95.9
12	0.8	56.5	-102.

## 6- Shipping Mode Analysis

**Question:** Examine how different shipping modes (Standard Class, Second Class, First Class, Same Day) affect sales and profits. Is there a correlation between ship mode and product category or segment?

### Solution and Explanation:

This analysis looks at sales and profits across different shipping modes and explores potential correlations with product categories or customer segments. Understanding these relationships can help in optimizing shipping strategies for better profitability.

```
# Analyze sales and profits by shipping mode
shipping_mode_analysis <- superstore %>%
  group_by(`Ship Mode`) %>%
  summarise(Total_Sales = sum(Sales), `Total Profit` = sum(Profit)) %>%
  arrange(desc(`Total Profit`))

# View the results
print(shipping_mode_analysis)
```

# A tibble: 4 × 3

	`Ship Mode`	Total_Sales	`Total Profit`
	<chr>	<dbl>	<dbl>
1	Standard Class	1358216.	164089.
2	Second Class	459940.	57673.
3	First Class	351428.	48970.
4	Same Day	128363.	15892.

## 7- Product Category Analysis

**Question:** Within each product category (Furniture, Office Supplies, Technology), identify the most and least profitable sub-categories. Discuss possible reasons and suggest adjustments to product offerings or marketing strategies.

### Solution and Explanation:

By breaking down sales and profits within each product category to the sub-category level, we can identify high and low performers. This detailed view provides insights for targeted strategies to enhance product offerings and marketing efforts.

```
# Analyze profitability by product category and sub-category
product_category_analysis <- superstore %>%
  group_by(Category, `Sub-Category`) %>%
  summarise(Total_Sales = sum(Sales), `Total Profit` = sum(Profit)) %>%
  arrange(Category, desc(`Total Profit`))
```

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

```
# View the results
print(product_category_analysis)
```

# A tibble: 17 × 4

# Groups: Category [3]

	Category	`Sub-Category`	Total_Sales	`Total Profit`
	<chr>	<chr>	<dbl>	<dbl>
1	Furniture	Chairs	329181.	26810.
2	Furniture	Furnishings	91705.	13059.
3	Furniture	Bookcases	114880.	-3473.
4	Furniture	Tables	206966.	-17725.
5	Office Supplies	Paper	78479.	34054.
6	Office Supplies	Binders	203413.	30222.
7	Office Supplies	Storage	223844.	21279.
8	Office Supplies	Appliances	107532.	18138.
9	Office Supplies	Envelopes	16476.	6964.
10	Office Supplies	Art	27119.	6528.
11	Office Supplies	Labels	12501.	5553.
12	Office Supplies	Fasteners	3024.	950.
13	Office Supplies	Supplies	46674.	-1189.
14	Technology	Copiers	149528.	55618.
15	Technology	Phones	330007.	44516.
16	Technology	Accessories	167380.	41937.
17	Technology	Machines	189239.	3385.

## 8- Customer Loyalty and Order Size

**Question:** Who are the top 10 most loyal customers by number of orders and by total sales? Analyze their purchasing patterns to determine what drives their loyalty.

### Solution and Explanation:

Identifying and analyzing the top 10 most loyal customers by both order frequency and sales volume offers valuable insights into customer loyalty drivers. This understanding can inform strategies to enhance customer retention and increase order size.

```
# Identify top 10 loyal customers by order frequency and total sales
customer_loyalty_analysis <- superstore %>%
  group_by(`Customer ID`, `Customer Name`) %>%
  summarise(Total_Orders = n(), Total_Sales = sum(Sales)) %>%
  arrange(desc(Total_Orders), desc(Total_Sales)) %>%
  slice_head(n = 10)
```

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

```
# View the results
print(customer_loyalty_analysis)
```

# A tibble: 793 × 4

# Groups: Customer ID [793]

`Customer ID`	`Customer Name`	Total_Orders	Total_Sales
<chr>	<chr>	<int>	<dbl>

1	AA-10315	Alex Avila	11	5564.
2	AA-10375	Allen Arnold	15	1056.
3	AA-10480	Andrew Allen	12	1791.
4	AA-10645	Anna Andreadi	18	5087.
5	AB-10015	Aaron Bergman	6	886.
6	AB-10060	Adam Bellavance	18	7756.
7	AB-10105	Adrian Barton	20	14474.
8	AB-10150	Aimee Bixby	12	967.
9	AB-10165	Alan Barnes	14	1114.
10	AB-10255	Alejandro Ballentine	14	915.

# i 783 more rows

## 9- Market Penetration and Expansion

**Question:** Identify cities and states with low sales volume. What could be potential reasons for low penetration in these areas? Propose strategies for market expansion.

### Solution and Explanation:

Focusing on areas with low sales volume helps in pinpointing under-served markets. Analyzing these areas can reveal opportunities for market penetration and expansion strategies.

```
# Identify areas with low sales volume
market_penetration_analysis <- superstore %>%
  group_by(State, City) %>%
  summarise(Total_Sales = sum(Sales)) %>%
  arrange(Total_Sales) %>%
  slice_head(n = 10)
```

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

```
# View the results
print(market_penetration_analysis)
```

```
# A tibble: 350 × 3
# Groups:   State [49]
  State City      Total_Sales
  <chr> <chr>      <dbl>
1 Alabama Tuscaloosa      176.
2 Alabama Hoover         526.
3 Alabama Auburn      1767.
4 Alabama Florence     1997.
5 Alabama Huntsville   2484.
6 Alabama Decatur      3375.
7 Alabama Montgomery   3723.
8 Alabama Mobile      5463.
9 Arizona Bullhead City    22.3
```

```
10 Arizona Sierra Vista      76.1
# i 340 more rows
```

---

## 10- Profit Efficiency by State or Region

**Question:** Which states or regions have the highest and lowest profit margins? Analyze factors that might contribute to these disparities and suggest ways to increase efficiency.

### Solution and Explanation:

This analysis ranks states or regions by profit margin to highlight areas of operational efficiency and inefficiency. Understanding these disparities can guide strategic decisions to improve profit margins across different markets.

```
# Calculate and compare profit efficiency by state or region
profit_efficiency_analysis <- superstore %>%
  group_by(Region) %>%
  summarise(Total_Sales = sum(Sales), `Total Profit` = sum(Profit),
            Profit_Margin = `Total Profit` / Total_Sales) %>%
  arrange(desc(Profit_Margin))

# View the results
print(profit_efficiency_analysis)
```

# A tibble: 4 × 4

	Region	Total_Sales	`Total Profit`	Profit_Margin
	<chr>	<dbl>	<dbl>	<dbl>
1	West	725472.	108425.	0.149
2	East	678781.	91523.	0.135
3	South	392454.	46969.	0.120
4	Central	501240.	39706.	0.0792