

SHETH L.U.J AND SIR M.V COLLEGE  
PRACTICAL NO 5  
SUBJECT - Data Analysis with SAS / SPSS / R

AIM - Sorting data using arrange() in R.

# Load packages

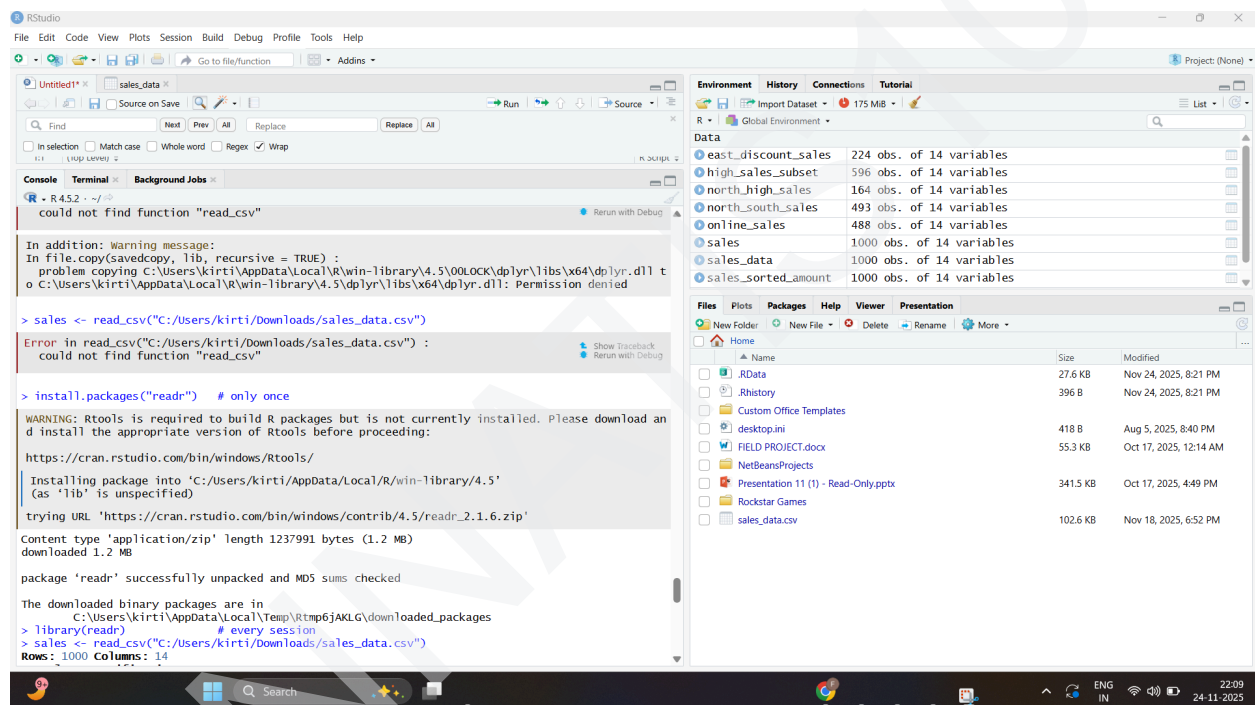
install.packages("dplyr", dependencies = FALSE) # optional (you can remove if already installed)

library(dplyr)

library(readr)

# Load the CSV (change path if your file is in a different folder)

sales <- read\_csv("C:/Users/kirti/Downloads/sales\_data.csv")



INPUT

# Example 1: Sort by a single variable (Sales\_Amount ascending)

# -----

# Sort sales by Sales\_Amount from smallest to largest

sales\_sorted\_amount <- sales |>

arrange(Sales\_Amount)

cat("\nTop 5 lowest Sales\_Amount:\n")

print(head(sales\_sorted\_amount, 5))

NAME- UNNATI RATHOD

ROLL NO - S109

SHETH L.U.J AND SIR M.V COLLEGE  
PRACTICAL NO 5  
SUBJECT - Data Analysis with SAS / SPSS / R

OUTPUT

The screenshot shows the RStudio interface. The script editor contains the following R code:

```
1 # Example 1: Sort by a single variable (Sales_Amount ascending)
2 # -----
3 # Sort sales by Sales_Amount from smallest to largest
4 sales_sorted_amount <- sales |>
5   arrange(Sales_Amount)
6
7 cat("\nTop 5 lowest Sales_Amount:\n")
8 print(head(sales_sorted_amount, 5))
```

The console output shows the execution of the code:

```
R 4.5.2 ~ /
> arrange(sales_amount)
> # Example 1: Sort by a single variable (Sales_Amount ascending)
> # -----
> # Sort sales by Sales_Amount from smallest to largest
> sales_sorted_amount <- sales |>
+   arrange(Sales_Amount)
>
> cat("\nTop 5 lowest Sales_Amount:\n")
Top 5 lowest Sales_Amount:
> print(head(sales_sorted_amount, 5))
# A tibble: 5 x 14
  Product_ID Sale_Date Sales_Rep Region Sales_Amount Quantity_Sold Product_Category Unit_Cost
  <dbl> <date> <chr> <chr> <dbl> <dbl> <chr> <dbl>
1 1070 2023-12-13 Alice West 100. 8 Clothing 3762.
2 1064 2023-12-19 Alice West 106. 35 Clothing 4900.
3 1078 2023-11-16 David East 113. 8 Furniture 3460.
4 1067 2023-06-29 Eve South 115. 48 Food 4319.
5 1087 2023-09-08 Charlie North 120. 20 Clothing 193.
# 6 more variables: Unit_Price <dbl>, Customer_Type <chr>, Discount <dbl>,
# Payment_Method <chr>, Sales_Channel <chr>, Region_and_Sales_Rep <chr>
> |
```

The Environment pane on the right shows the following data objects:

Object	Size
east_discount_sales	224 obs. of 14 variables
high_sales_subset	596 obs. of 14 variables
north_high_sales	164 obs. of 14 variables
north_south_sales	493 obs. of 14 variables
online_sales	488 obs. of 14 variables
sales	1000 obs. of 14 variables
sales_data	1000 obs. of 14 variables
sales_sorted_amount	1000 obs. of 14 variables

INPUT

```
# Example 2: Sort by a single variable (Quantity_Sold descending)
# -----
# Find the largest quantity orders first
sales_sorted_qty_desc <- sales |>
  arrange(desc(Quantity_Sold))

cat("\nTop 5 largest Quantity_Sold:\n")
print(head(sales_sorted_qty_desc, 5))
```

OUTPUT

NAME- UNNATI RATHOD  
ROLL NO - S109

SHETH L.U.J AND SIR M.V COLLEGE  
PRACTICAL NO 5  
SUBJECT - Data Analysis with SAS / SPSS / R

The screenshot shows the RStudio environment with a script editor, console, and environment pane. The script editor contains R code for sorting data by quantity sold. The console shows the output of the code, including a table of the top 5 largest quantity orders.

```
# Example 2: Sort by a single variable (Quantity_Sold descending)
# Find the largest quantity orders first
sales_sorted_qty_desc <- sales |>
  arrange(desc(Quantity_Sold))
cat("\nTop 5 largest Quantity_Sold:\n")
print(head(sales_sorted_qty_desc, 5))
```

Console Output:

```
R 4.5.2 ~ /~
3 1078 2023-11-16 David East 113. 8 Furniture 3460.
4 1067 2023-06-29 Eve South 115. 48 Food 4319.
5 1087 2023-09-08 Charlie North 120. 20 Clothing 193.
# 6 more variables: Unit_Price <dbl>, Customer_Type <chr>, Discount <dbl>,
# Payment_Method <chr>, Sales_Channel <chr>, Region_and_Sales_Rep <chr>
> # Example 2: Sort by a single variable (Quantity_Sold descending)
> # Find the largest quantity orders first
> sales_sorted_qty_desc <- sales |>
+   arrange(desc(Quantity_Sold))
>
> cat("\nTop 5 largest Quantity_Sold:\n")
Top 5 largest Quantity_Sold:
> print(head(sales_sorted_qty_desc, 5))
# A tibble: 5 x 14
  Product_ID Sale_Date Sales_Rep Region Sales_Amount Quantity_Sold Product_Category Unit_Cost
  <dbl> <date> <chr> <chr> <dbl> <dbl> <chr> <dbl>
1 1036 2023-10-20 David North 6500. 49 Clothing 1247.
2 1006 2023-09-16 Bob North 8996. 49 Food 2844.
3 1078 2023-04-04 David East 9631. 49 Furniture 1834.
4 1078 2023-02-08 Alice South 9813. 49 Furniture 3027.
5 1032 2023-06-08 Eve West 220. 49 Electronics 4812.
# 6 more variables: Unit_Price <dbl>, Customer_Type <chr>, Discount <dbl>,
# Payment_Method <chr>, Sales_Channel <chr>, Region_and_Sales_Rep <chr>
>
```

INPUT

# Example 3: Sort by multiple columns (Region then Sales\_Amount descending)

# -----

# Primary sort: Region (alphabetical)

# Secondary sort: Sales\_Amount (highest first)

```
sales_multi_sort <- sales |>
```

```
  arrange(Region, desc(Sales_Amount))
```

```
cat("\nFirst 8 rows sorted by Region then Sales_Amount (desc):\n")
```

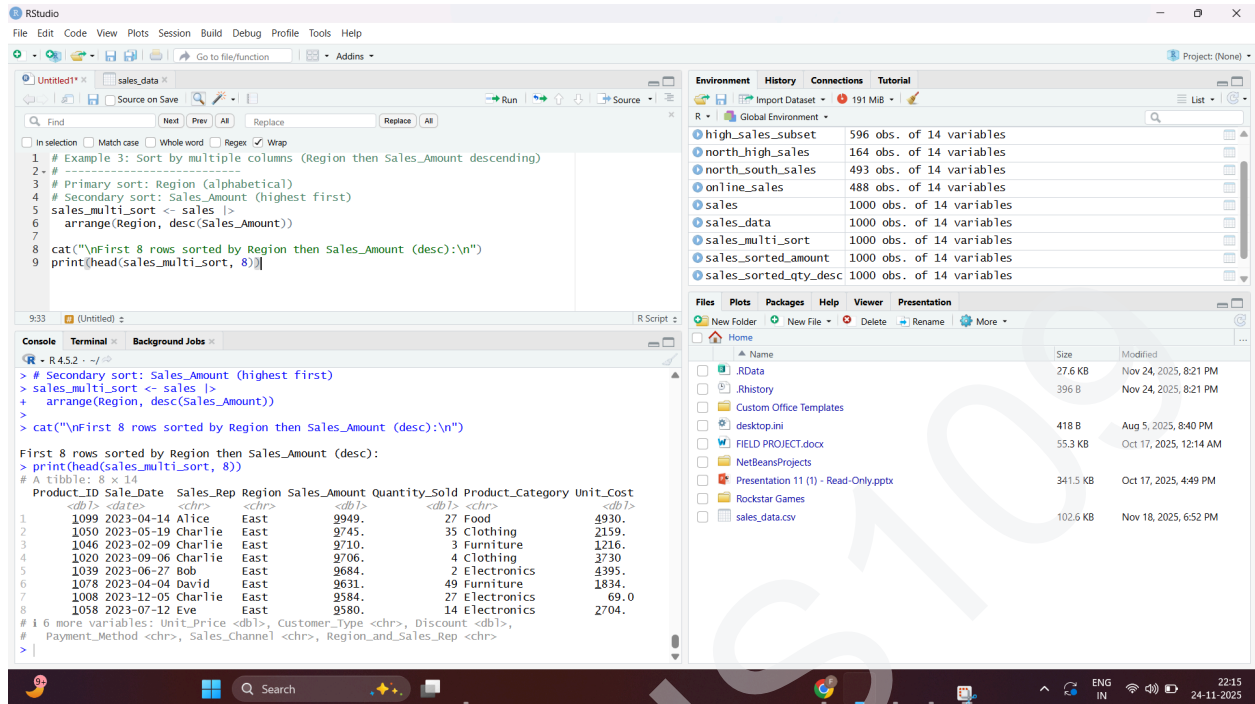
```
print(head(sales_multi_sort, 8))
```

OUTPUT

NAME- UNNATI RATHOD

ROLL NO - S109

SHETH L.U.J AND SIR M.V COLLEGE  
PRACTICAL NO 5  
SUBJECT - Data Analysis with SAS / SPSS /R



```
1 # Example 3: Sort by multiple columns (Region then Sales_Amount descending)
2 # -----
3 # Primary sort: Region (alphabetical)
4 # Secondary sort: Sales_Amount (highest first)
5 sales_multi_sort <- sales |>
6   arrange(Region, desc(Sales_Amount))
7
8 cat("\nFirst 8 rows sorted by Region then Sales_Amount (desc):\n")
9 print(head(sales_multi_sort, 8))
```

```
> # Secondary sort: Sales_Amount (highest first)
> sales_multi_sort <- sales |>
+   arrange(Region, desc(Sales_Amount))
>
> cat("\nFirst 8 rows sorted by Region then Sales_Amount (desc):\n")
First 8 rows sorted by Region then Sales_Amount (desc):
> print(head(sales_multi_sort, 8))
# A tibble: 8 x 14
  Product_ID Sale_Date Sales_Rep Region Sales_Amount Quantity_Sold Product_Category Unit_Cost
  <dbl> <date> <chr> <chr> <dbl> <dbl> <chr> <dbl>
1 1099 2023-04-14 Alice East 9949. 27 Food 4930.
2 1050 2023-05-19 Charlie East 9745. 35 Clothing 2159.
3 1046 2023-02-09 Charlie East 9710. 3 Furniture 1216.
4 1020 2023-09-06 Charlie East 9706. 4 Clothing 3730.
5 1039 2023-06-27 Bob East 9684. 2 Electronics 4395.
6 1078 2023-04-04 David East 9631. 49 Furniture 1834.
7 1008 2023-12-05 Charlie East 9584. 27 Electronics 69.0
8 1058 2023-07-12 Eve East 9580. 14 Electronics 2704.
```

INPUT

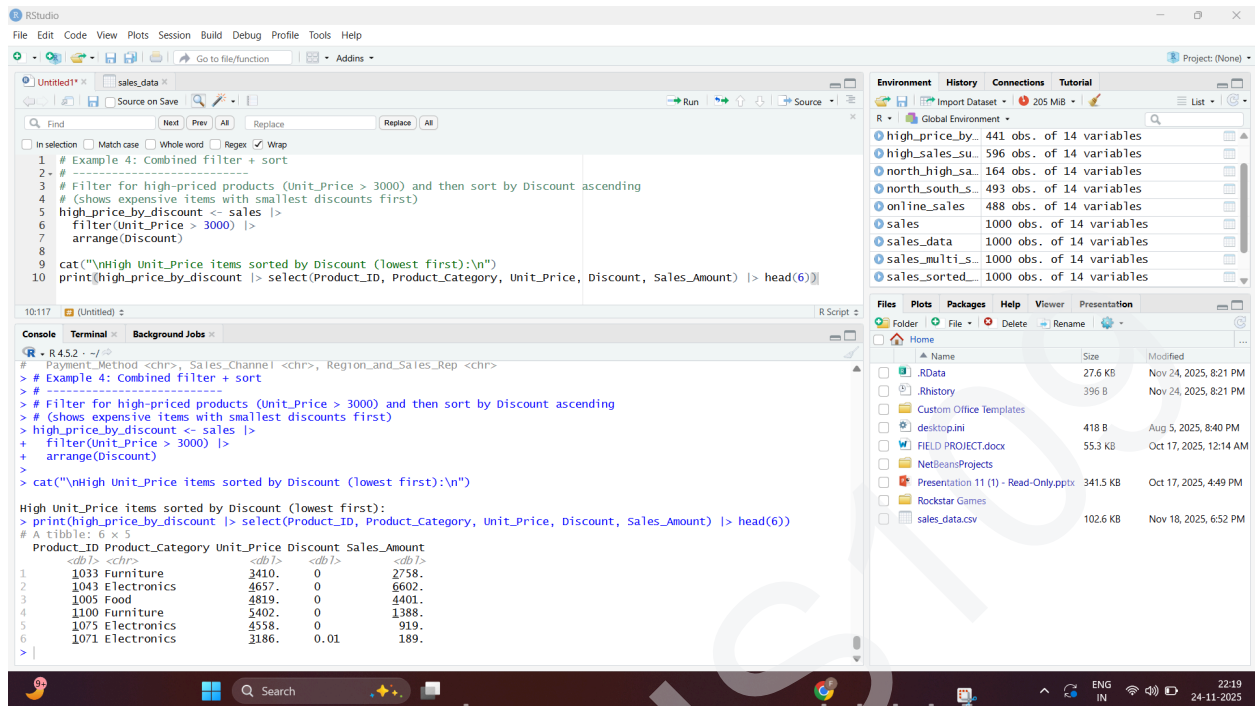
```
# Example 4: Combined filter + sort
# -----
# Filter for high-priced products (Unit_Price > 3000) and then sort by Discount ascending
# (shows expensive items with smallest discounts first)
high_price_by_discount <- sales |>
  filter(Unit_Price > 3000) |>
  arrange(Discount)

cat("\nHigh Unit_Price items sorted by Discount (lowest first):\n")
print(high_price_by_discount |> select(Product_ID, Product_Category, Unit_Price, Discount,
Sales_Amount) |> head(6))
```

OUTPUT

NAME- UNNATI RATHOD  
ROLL NO - S109

SHETH L.U.J AND SIR M.V COLLEGE  
PRACTICAL NO 5  
SUBJECT - Data Analysis with SAS / SPSS / R



The screenshot shows the RStudio interface. The script editor contains R code for filtering and sorting data. The console shows the execution of the code, including the output of a filter and sort operation, and a head command showing the first six rows of the resulting data frame.

```
# Example 4: Combined filter + sort
# Filter for high-priced products (Unit_Price > 3000) and then sort by Discount ascending
# (shows expensive items with smallest discounts first)
high_price_by_discount <- sales |>
  filter(Unit_Price > 3000) |>
  arrange(Discount)

cat("\nHigh Unit_Price items sorted by Discount (lowest first):\n")
print(high_price_by_discount |> select(Product_ID, Product_Category, Unit_Price, Discount, Sales_Amount) |> head(6))
```

High Unit\_Price items sorted by Discount (lowest first):

```
> print(high_price_by_discount |> select(Product_ID, Product_Category, Unit_Price, Discount, Sales_Amount) |> head(6))
# A tibble: 6 x 5
  Product_ID Product_Category Unit_Price Discount Sales_Amount
  <dbl> <chr> <dbl> <dbl> <dbl>
1 1033 Furniture 3410. 0 2758.
2 1043 Electronics 4657. 0 6602.
3 1005 Food 4819. 0 4401.
4 1100 Furniture 5402. 0 1388.
5 1075 Electronics 4558. 0 919.
6 1071 Electronics 3186. 0.01 189.
```

INPUT

# Example 5: Another useful one — top 10 sales by Sales\_Amount overall

# -----

```
top10_sales <- sales |>
```

```
  arrange(desc(Sales_Amount)) |>
```

```
  slice_head(n = 10)
```

```
cat("\nTop 10 sales by Sales_Amount:\n")
```

```
print(top10_sales |> select(Product_ID, Sale_Date, Sales_Rep, Region, Sales_Amount) )
```

OUTPUT

NAME- UNNATI RATHOD

ROLL NO - S109

SHETH L.U.J AND SIR M.V COLLEGE  
PRACTICAL NO 5  
SUBJECT - Data Analysis with SAS / SPSS / R

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for data manipulation:

```
1 # Example 5: Another useful one - top 10 sales by Sales_Amount overall
2 # -----
3 top10_sales <- sales |>
4   arrange(desc(Sales_Amount)) |>
5   slice_head(n = 10)
6
7 cat("\nTop 10 sales by Sales_Amount:\n")
8 print(top10_sales |> select(Product_ID, Sale_Date, Sales_Rep, Region, Sales_Amount) )
```
- Console:** Shows the execution output:

```
> top10_sales <- sales |>
+   arrange(desc(Sales_Amount)) |>
+   slice_head(n = 10)
>
> cat("\nTop 10 sales by Sales_Amount:\n")
Top 10 sales by Sales_Amount:
> print(top10_sales |> select(Product_ID, Sale_Date, Sales_Rep, Region, Sales_Amount) )
# A tibble: 10 x 5
  Product_ID Sale_Date Sales_Rep Region Sales_Amount
  <dbl> <date> <chr> <chr> <dbl>
1 1036 2023-12-10 David North 9989.
2 1050 2023-08-21 David North 9977.
3 1079 2023-08-24 Alice North 9973.
4 1075 2023-08-15 Charlie North 9972.
5 1016 2023-04-23 David South 9962.
6 1063 2023-08-14 Bob North 9957.
7 1099 2023-04-14 Alice East 9949.
8 1089 2023-10-19 Eve West 9933.
9 1015 2023-09-22 David South 9914.
10 1010 2023-10-12 Charlie North 9908.
```
- Environment Pane:** Lists loaded objects:
  - north\_south\_s... 493 obs. of 14 variables
  - online\_sales 488 obs. of 14 variables
  - sales 1000 obs. of 14 variables
  - sales\_data 1000 obs. of 14 variables
  - sales\_multi\_s... 1000 obs. of 14 variables
  - sales\_sorted\_... 1000 obs. of 14 variables
  - sales\_sorted\_... 1000 obs. of 14 variables
  - special\_custo... 752 obs. of 14 variables
  - top10\_sales 10 obs. of 14 variables
- Files Pane:** Shows the file explorer with files like .RData, .Rhistory, Custom Office Templates, desktop.ini, FIELD PROJECT.docx, NetBeansProjects, Presentation 11 (1) - Read-Only.pptx, Rockstar Games, and sales\_data.csv.

NAME- UNNATI RATHOD  
ROLL NO - S109