EXPT NO:6	IMPLEMENTATION OF MULTIVARIATE DISPLAYS
DATE: 24.01.2026	

PRE-LAB QUESTIONS

1. Why are multivariate displays important in AI analytics?
   Multivariate displays help analyze **multiple variables at the same time**, which is important in AI because models depend on many features together. They help in:

- Finding **hidden patterns and relationships**
- Detecting **feature importance**
- Identifying **outliers and anomalies**
- Understanding **data distribution before training**

2. How do parallel coordinates differ from scatter plots?
   - **Scatter Plot:** compares **only 2 variables** at a time (X vs Y).
   - **Parallel Coordinates:** compares **many variables together** by drawing each record as a line across multiple axes.

   Parallel coordinates are better when you want to see **multi-feature behavior** in one view.

3. What challenges exist in interpreting multivariate plots?
   Some common challenges are:

- **Overplotting / clutter** (too many lines or points)
- Hard to track patterns when variables are many
- Difficult to interpret without good scaling or normalization
- Requires more skill to read compared to simple plots

4. Where are trellis displays commonly used?
   Trellis displays are used when we want to compare the same chart across categories, such as:

- Sales performance by **region**
- Customer behavior by **segment**
- Machine sensor readings by **machine type**
- Medical data by **age group or gender**
  They are common in **business analytics, healthcare, and ML feature analysis**.

5. How does multivariate visualization aid model evaluation?
   It helps evaluate AI models by:

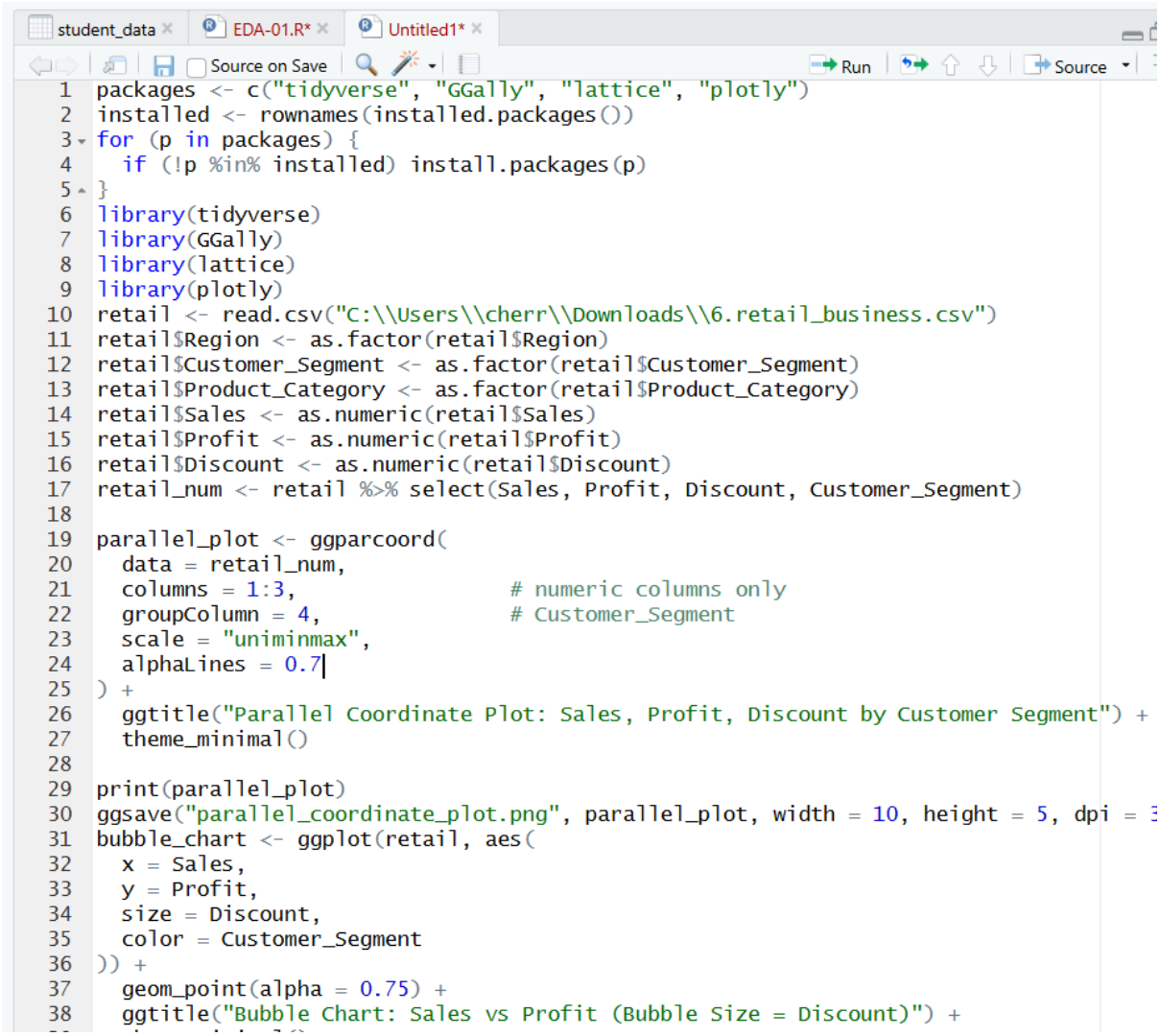
- Showing how features affect predictions
- Identifying **misclassified regions**
- Checking feature interactions that influence accuracy
- Detecting **bias** across groups (region, segment, category)
- Understanding errors and improving model performance

OBJECTIVE : To implement advanced multivariate displays for complex data analysis.

SCENARIO A retail analytics firm studies sales, profit, customer segment, and region to optimize business strategy.

IN-LAB TASKS (Using R Language) • Create parallel coordinate plots • Generate bubble charts • Implement trellis displays by region

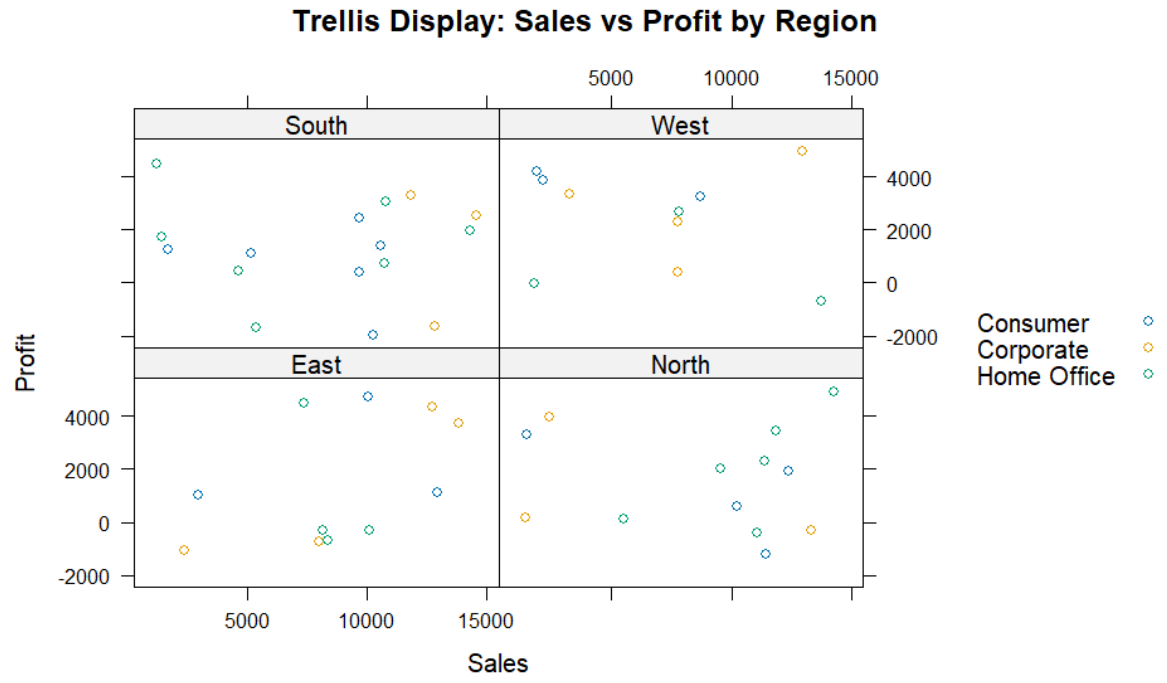
CODE:

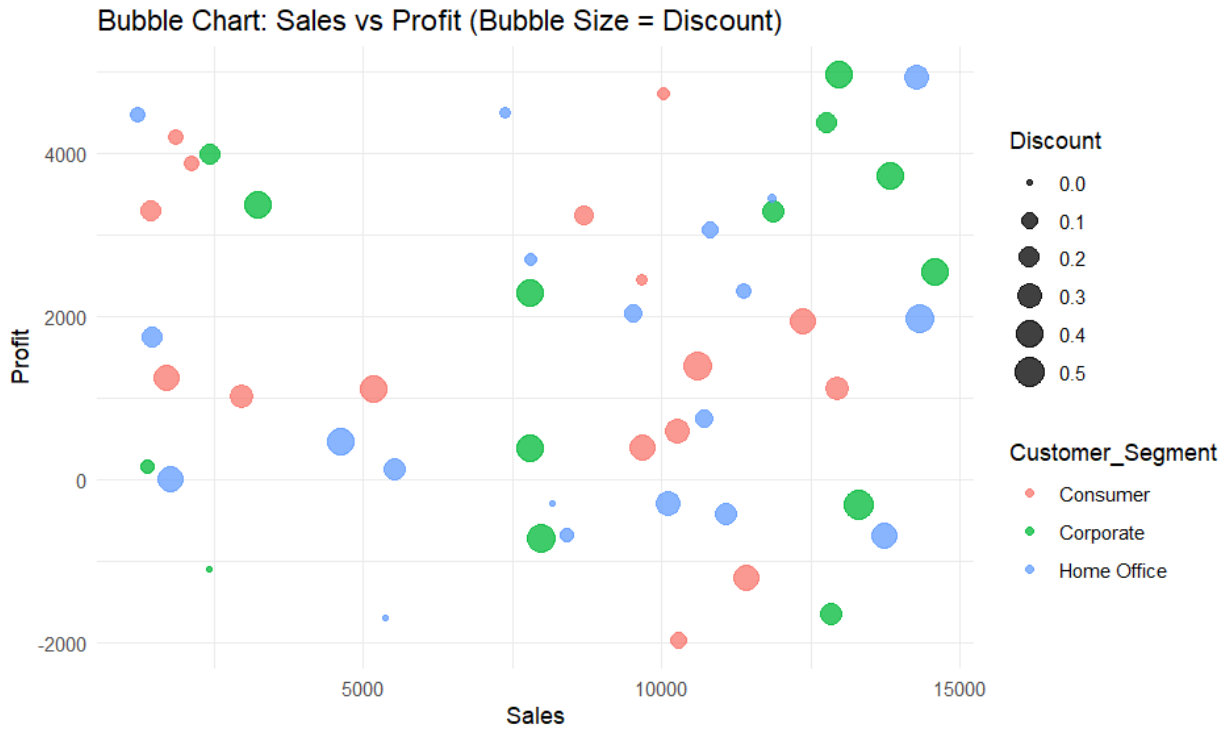


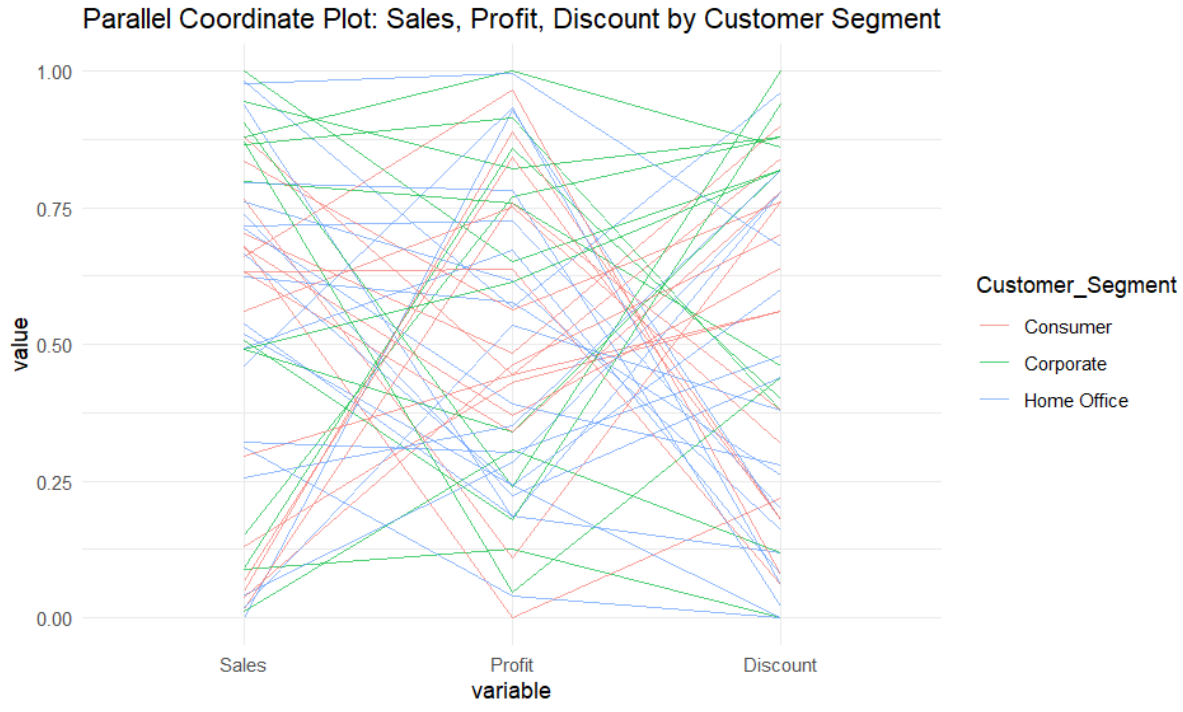
```r
packages <- c("tidyverse", "GGally", "lattice", "plotly")
installed <- rownames(installed.packages())
for (p in packages) {
  if (!p %in% installed) install.packages(p)
}
library(tidyverse)
library(GGally)
library(lattice)
library(plotly)
retail <- read.csv("C:\\Users\\cherr\\Downloads\\6.retail_business.csv")
retail$Region <- as.factor(retail$Region)
retail$Customer_Segment <- as.factor(retail$Customer_Segment)
retail$Product_Category <- as.factor(retail$Product_Category)
retail$Sales <- as.numeric(retail$Sales)
retail$Profit <- as.numeric(retail$Profit)
retail$Discount <- as.numeric(retail$Discount)
retail_num <- retail %>% select(Sales, Profit, Discount, Customer_Segment)

parallel_plot <- ggparcoord(
  data = retail_num,
  columns = 1:3,                    # numeric columns only
  groupColumn = 4,                  # Customer_Segment
  scale = "uniminmax",
  alphaLines = 0.7
) +
  ggtitle("Parallel Coordinate Plot: Sales, Profit, Discount by Customer Segment") +
  theme_minimal()

print(parallel_plot)
ggsave("parallel_coordinate_plot.png", parallel_plot, width = 10, height = 5, dpi = 3
bubble_chart <- ggplot(retail, aes(
  x = Sales,
  y = Profit,
  size = Discount,
  color = Customer_Segment
)) +
  geom_point(alpha = 0.75) +
  ggtitle("Bubble Chart: Sales vs Profit (Bubble Size = Discount)") +
```

```r
print(parallel_plot)
ggsave("parallel_coordinate_plot.png", parallel_plot, width = 10, height = 5, dpi = 3
bubble_chart <- ggplot(retail, aes(
  x = Sales,
  y = Profit,
  size = Discount,
  color = Customer_Segment
)) +
  geom_point(alpha = 0.75) +
  ggtitle("Bubble Chart: Sales vs Profit (Bubble Size = Discount)") +
  theme_minimal()

print(bubble_chart)

ggsave("bubble_chart.png", bubble_chart, width = 8, height = 5, dpi = 300)
trellis_plot <- xyplot(
  Profit ~ Sales | Region,
  data = retail,
  groups = Customer_Segment,
  auto.key = TRUE,
  type = "p",
  main = "Trellis Display: Sales vs Profit by Region",
  xlab = "Sales",
  ylab = "Profit"
)

print(trellis_plot)

png("trellis_display_region.png", width = 1000, height = 600)
print(trellis_plot)
dev.off()

cat("\n✅ All plots generated successfully!\n")
cat("Saved plots:\n")
cat("1) parallel_coordinate_plot.png\n")
cat("2) bubble_chart.png\n")
cat("3) trellis_display_region.png\n")

```

OUTPUT:




Bubble Chart: Sales vs Profit (Bubble Size = Discount)
Profit
4000
2000
0
-2000
5000
10000
15000
Sales
Discount
0.0
0.1
0.2
0.3
0.4
0.5
Customer_Segment
Consumer
Corporate
Home Office


Parallel Coordinate Plot: Sales, Profit, Discount by Customer Segment
value
1.00
0.75
0.50
0.25
0.00
Sales
Profit
Discount
variable
Customer_Segment
Consumer
Corporate
Home Office

POST-LAB QUESTIONS

1. What insights are gained from parallel coordinates?
   Parallel coordinate plots help identify:

- **Patterns across multiple variables** at once
- **High/low performance trends** (ex: high sales but low profit)
- **Clusters of similar records**
- **Outliers** where a line deviates strongly from others

2. How does faceting simplify complex data?
   Faceting splits one big plot into multiple smaller plots based on categories (like region/segment).
   This makes it easier to:

- Compare groups side-by-side
- Detect category-wise differences clearly
- Reduce clutter and improve readability

3. What limitations exist in bubble charts?
   Bubble charts can be difficult because:

- Too many bubbles cause **overlapping**
- Bubble size differences are **hard to judge accurately**
- Small bubbles may become invisible
- Not ideal for very large datasets (becomes messy)

4. How can these displays support AI-driven recommendations?
   These plots help AI systems by:

- Finding which **segments/regions give higher profit**
- Identifying products with **high sales but low profit** for improvement
- Supporting personalized offers using customer segment behavior
- Helping models recommend best strategies based on patterns in data

5. Suggest improvements for large multivariate datasets.
   For large datasets, improve multivariate visualization by:

- Using **filtering and sampling** to reduce clutter
- Applying **normalization/scaling** for fair comparison
- Using **interactive dashboards** (zoom, hover, highlight)
- Using **dimensionality reduction** like PCA/t-SNE for better feature view
- Grouping data using **clustering** before plotting

LEARNING OUTCOME: Students apply multivariate visualization for business intelligence.

ASSESSMENT

Description	Max Marks	Marks Awarded
Pre Lab Exercise	**5**	
In Lab Exercise	**10**	
Post Lab Exercise	**5**	
Viva	**10**	
Total	**30**	
Faculty Signature		