

Một số chủ đề trong mô hình hóa và phân tích dữ liệu

BÁO CÁO CUỐI KỲ



# Phân Tích Dữ Liệu Bán Hàng BigMart

Sinh viên: Vũ Mạnh Đức - 20002050

Nguyễn Phan Anh - 20002030

Vương Thùy Dương - 20002038

Giảng viên: TS. Phạm Đình Tùng

# NỘI DUNG



1



2



3



4

GIỚI THIỆU CHỦ ĐỀ

KHAI PHÁ & XỬ LÝ  
DỮ LIỆU

XÂY DỰNG & ĐÁNH GIÁ  
MÔ HÌNH

KẾT LUẬN VẤN ĐỀ &  
ĐỊNH HƯỚNG



# 1. GIỚI THIỆU CHỦ ĐỀ

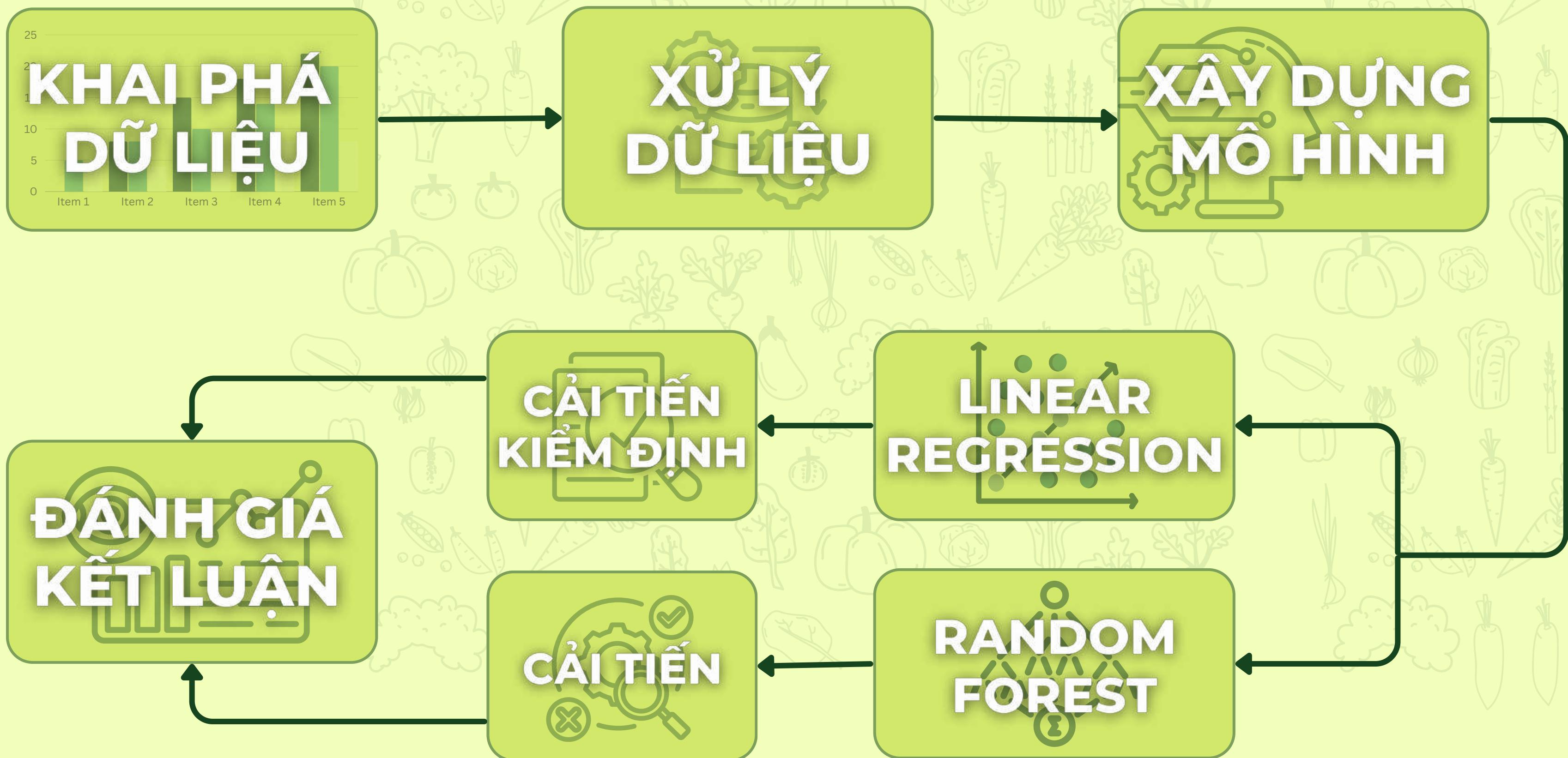


# GIỚI THIỆU CHỦ ĐỀ

- **Mô tả đề tài:** Các nhà KHDN tại hệ thống Big Mart đã tập hợp dữ liệu bán hàng năm 2013 của **1559 sản phẩm tại 10 cửa hàng** ở các thành phố khác nhau. Một số thuộc tính nhất định của từng sản phẩm và của mỗi cửa hàng đã được xác định.
- **Nhiệm vụ:** Phân tích và dự đoán doanh số bán hàng cho mỗi một sản phẩm tại một cửa hàng cụ thể



# FRAMEWORK





## 2. KHAI PHÁ & XỬ LÝ DỮ LIỆU

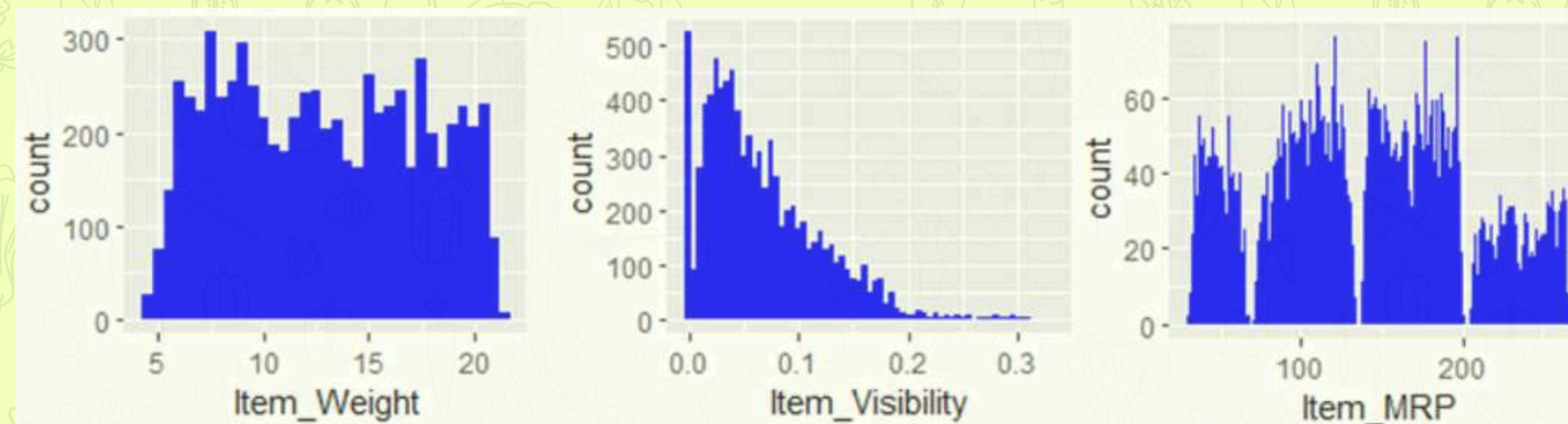


# KHAI PHÁ DỮ LIỆU:

Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type	Item_Outlet_Sales
<chr>	<dbl>	<chr>	<dbl>	<chr>	<dbl>	<chr>	<int>	<chr>	<chr>	<chr>	<dbl>
FDA15	9.300	Low Fat	0.016047301	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1	3735.1380
DRC01	5.920	Regular	0.019278216	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2	443.4228
FDN15	17.500	Low Fat	0.016760075	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1	2097.2700
FDX07	19.200	Regular	0.000000000	Fruits and Vegetables	182.0950	OUT010	1998		Tier 3	Grocery Store	732.3800
NCD19	8.930	Low Fat	0.000000000	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1	994.7052
FDP36	10.395	Regular	0.000000000	Baking Goods	51.4008	OUT018	2009	Medium	Tier 3	Supermarket Type2	556.6088
FDO10	13.650	Regular	0.012741089	Snack Foods	57.6588	OUT013	1987	High	Tier 3	Supermarket Type1	343.5528

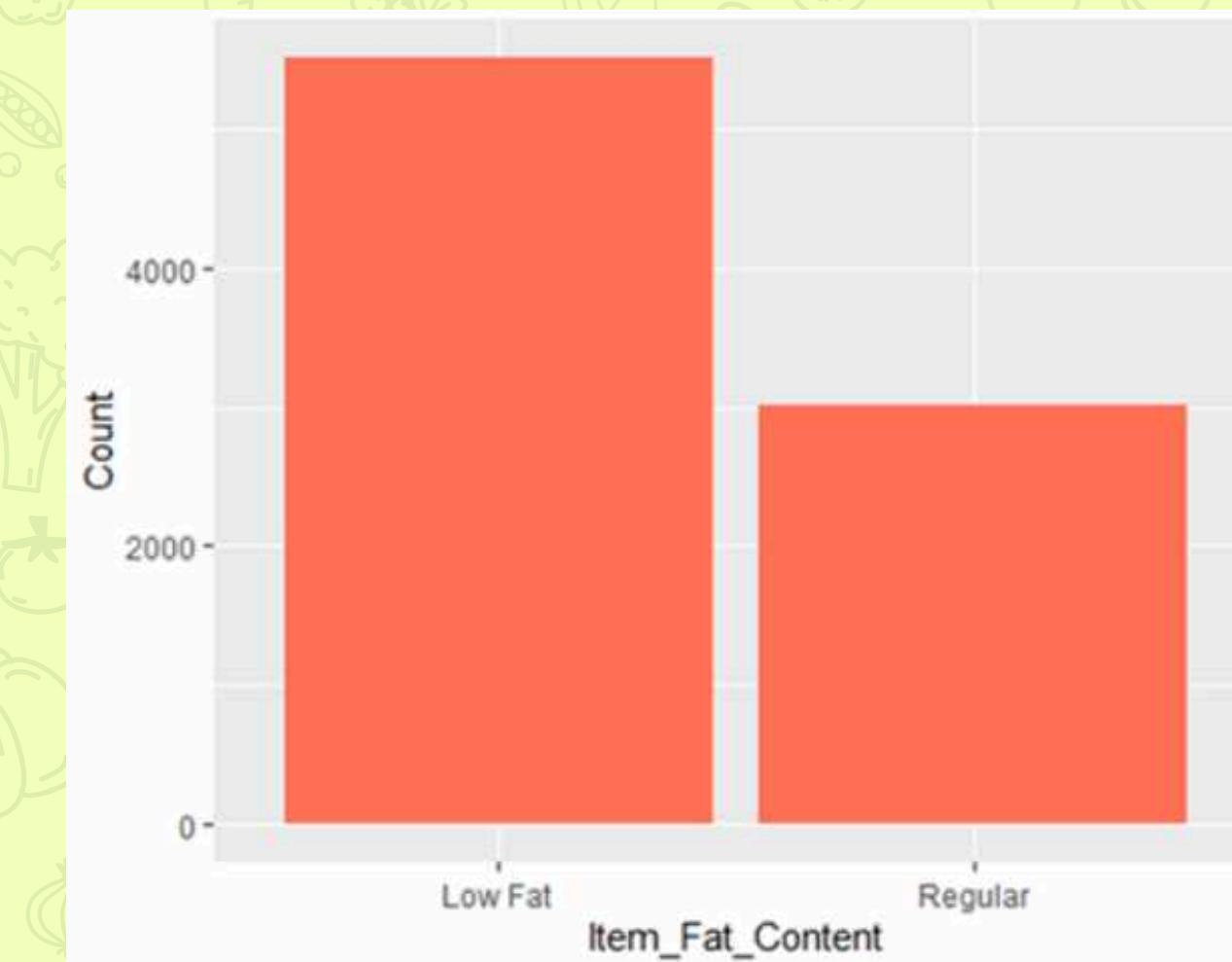
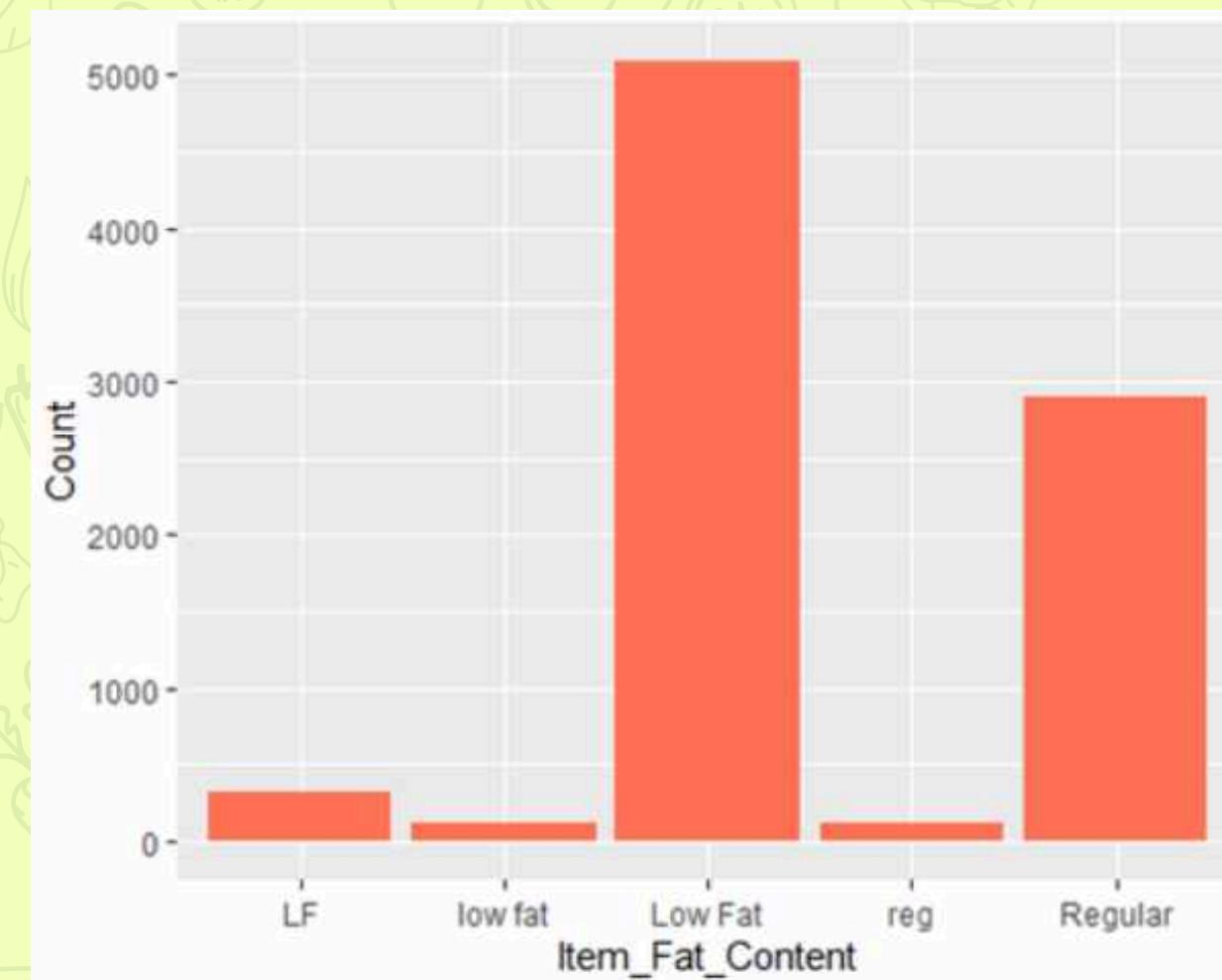
- Dữ liệu BigMart được thu thập từ 2013, có tất cả 12 biến:
  - 11 biến độc lập:
    - 3 biến định lượng
    - 7 biến định tính
  - 1 biến phụ thuộc (**Item\_Outlet\_Sales**)

# PHÂN TÍCH ĐƠN BIẾN :



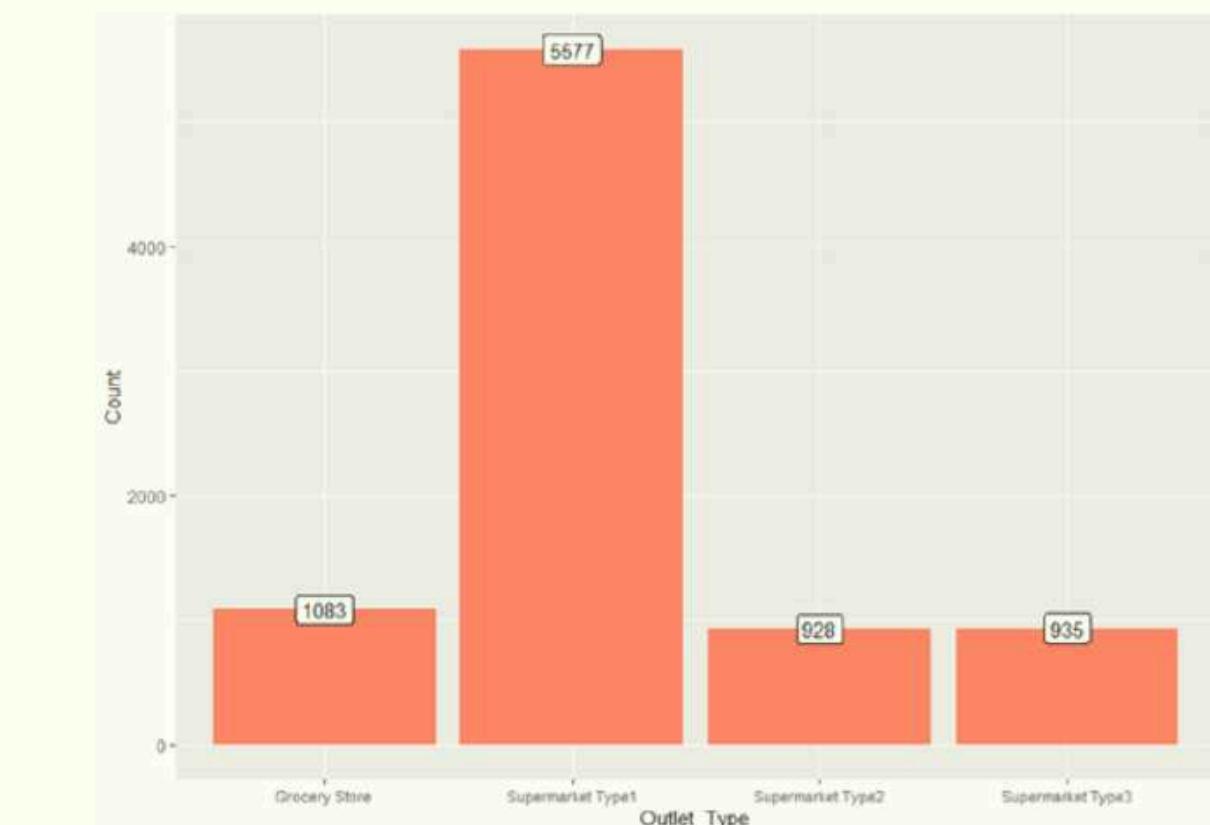
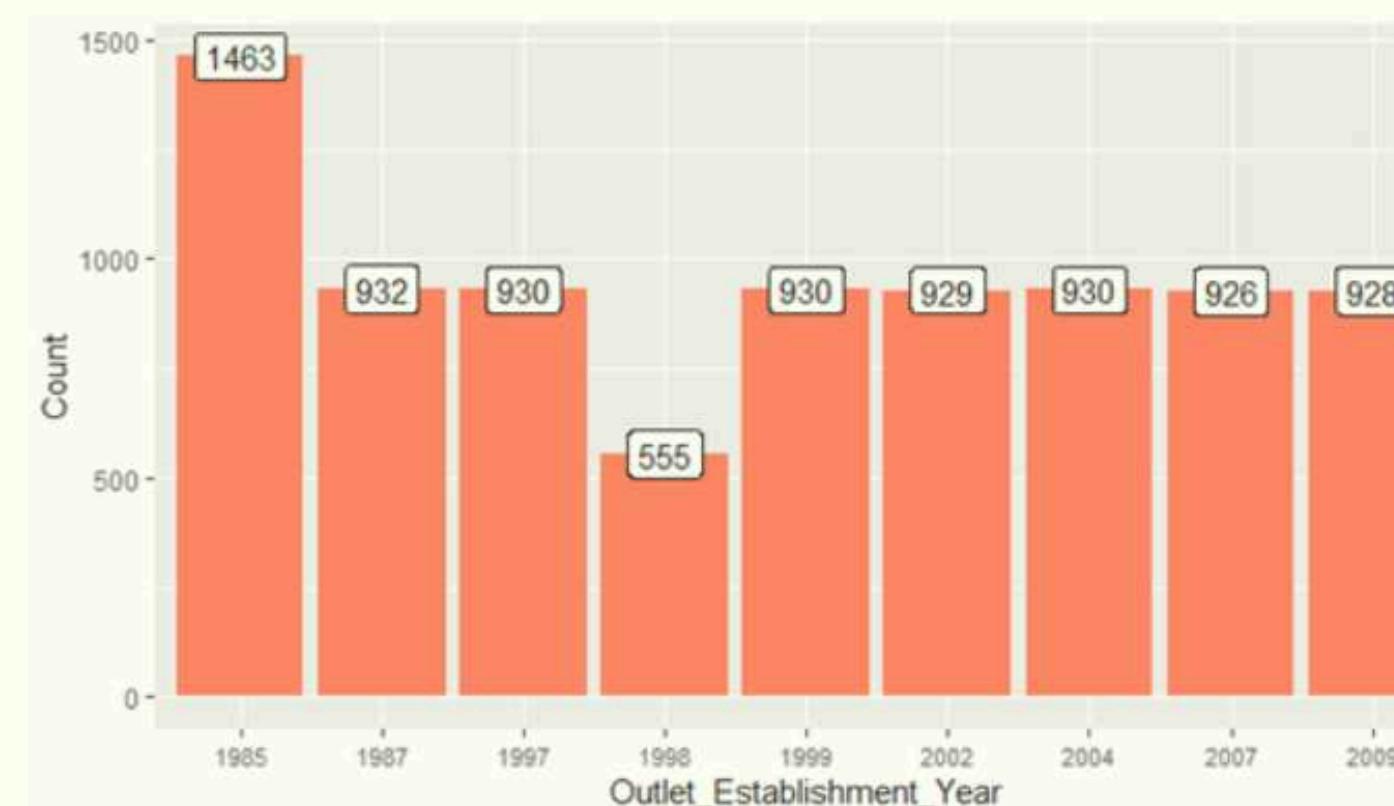
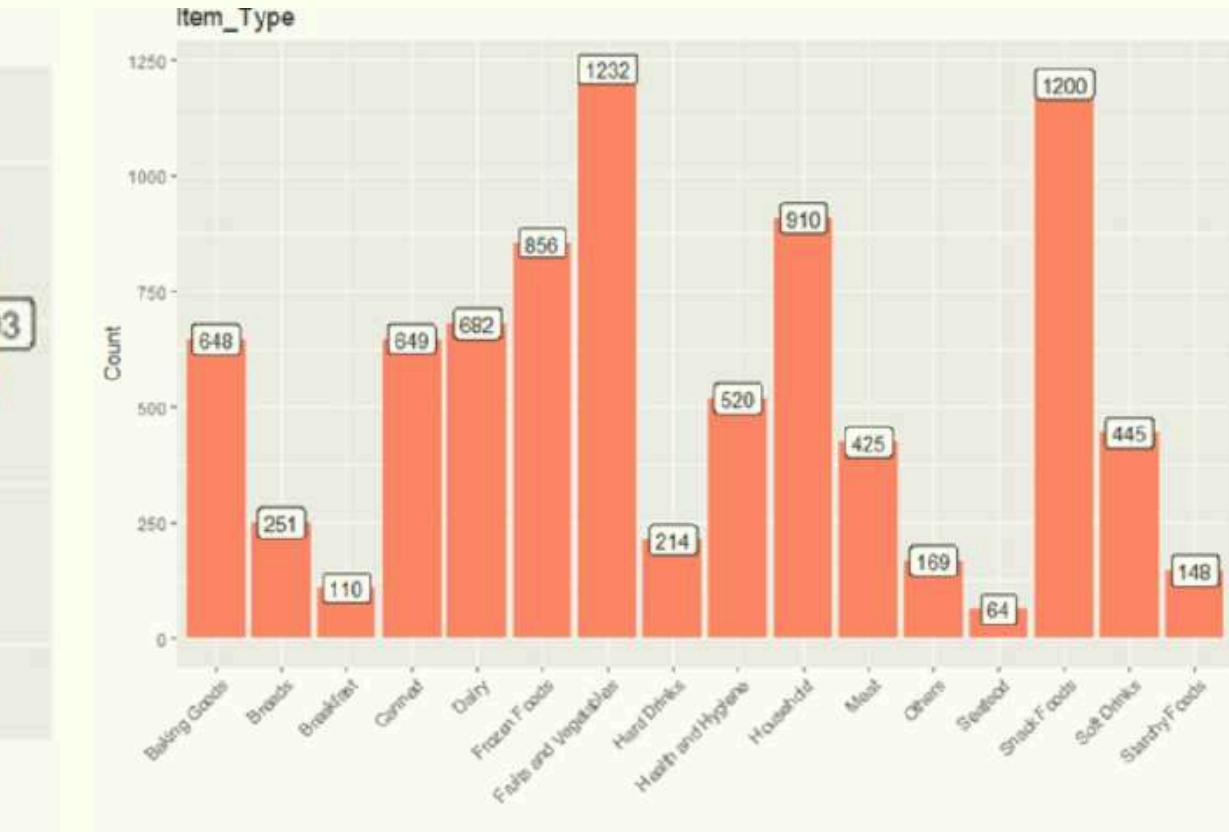
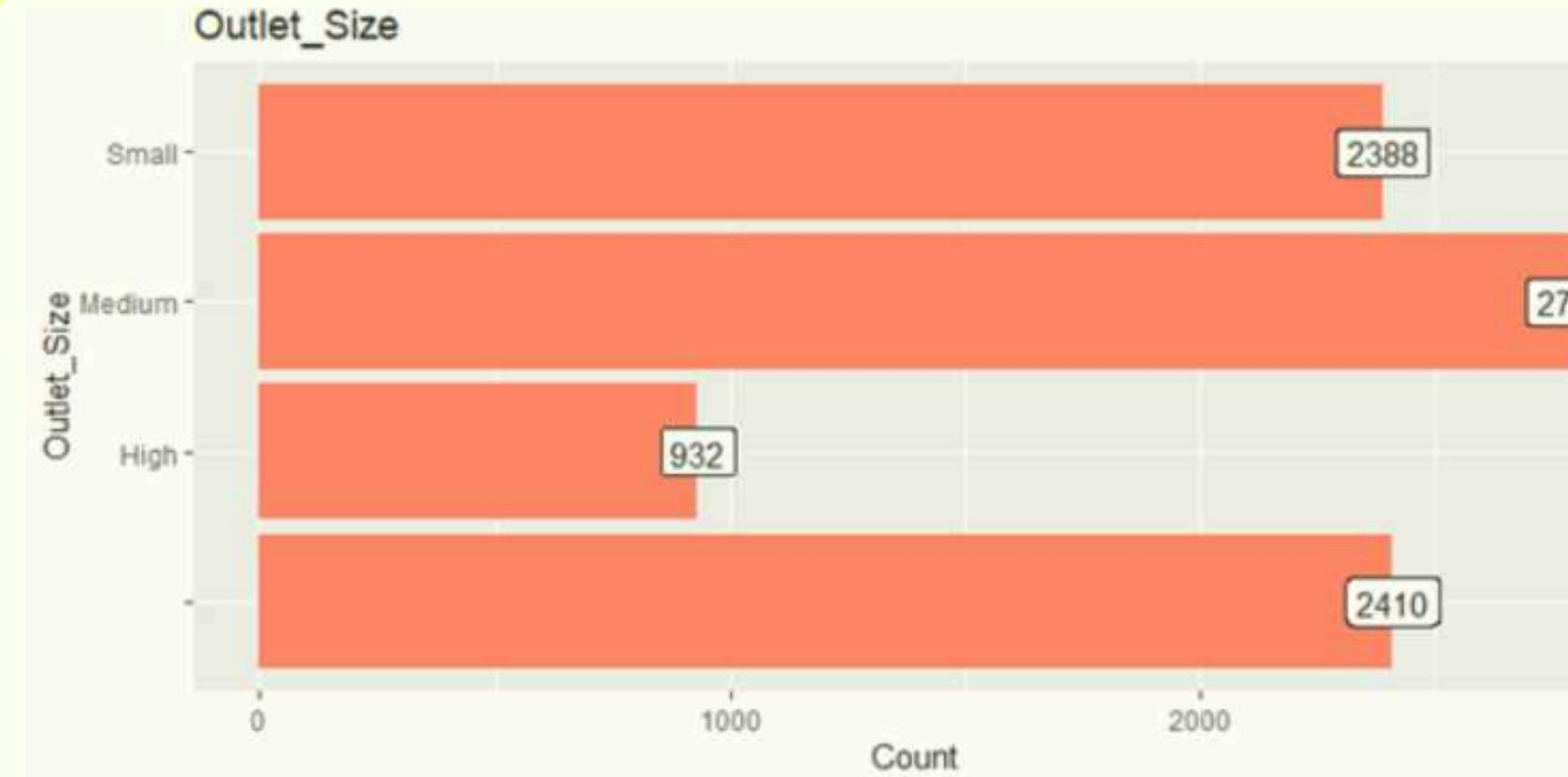
- Sử dụng histogram để xem phân phối của 3 biến định tính:
  - Item\_Weight có phân phối khá đều
  - Item\_MRP có phân phối đa đỉnh
  - Item\_Visibility có độ lệch nghiêng hẳn sang trái

# PHÂN TÍCH ĐƠN BIẾN:

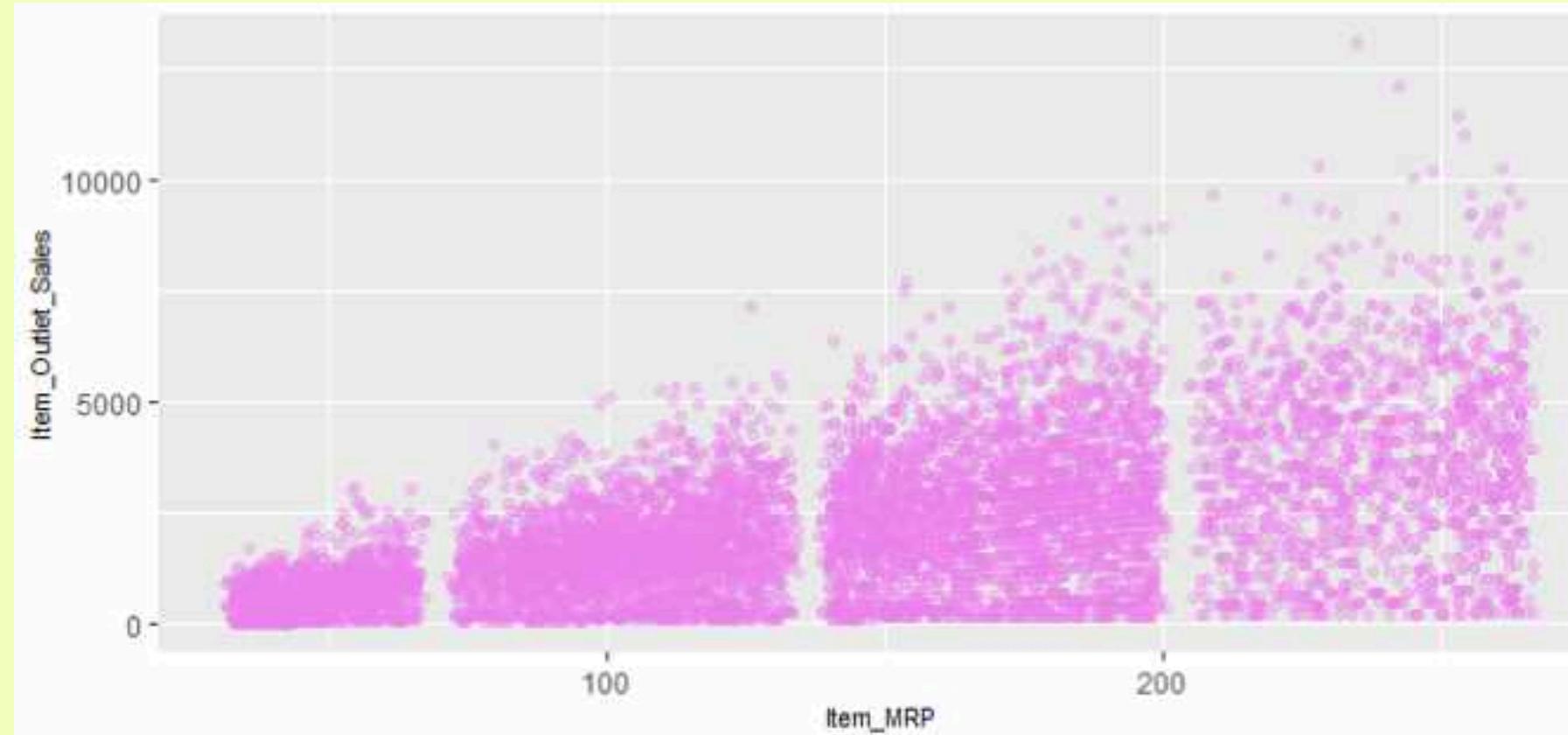
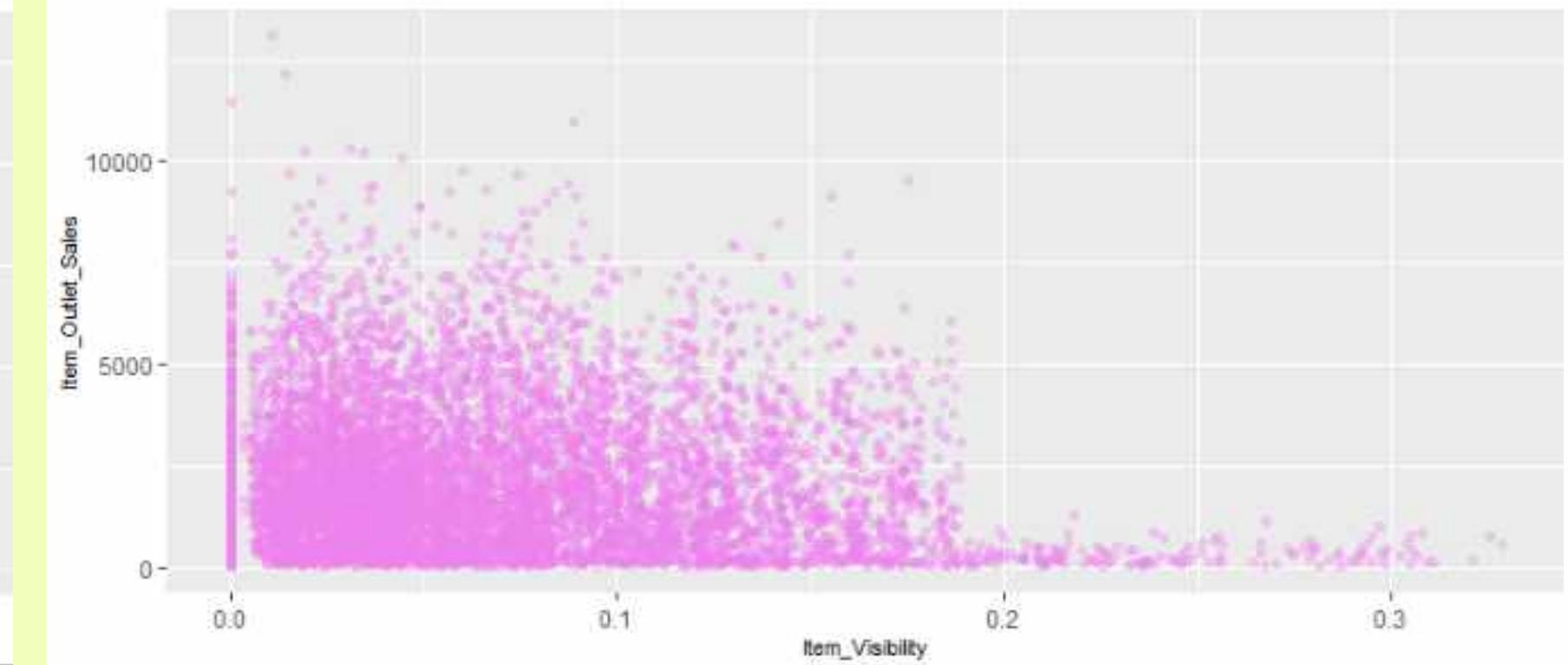
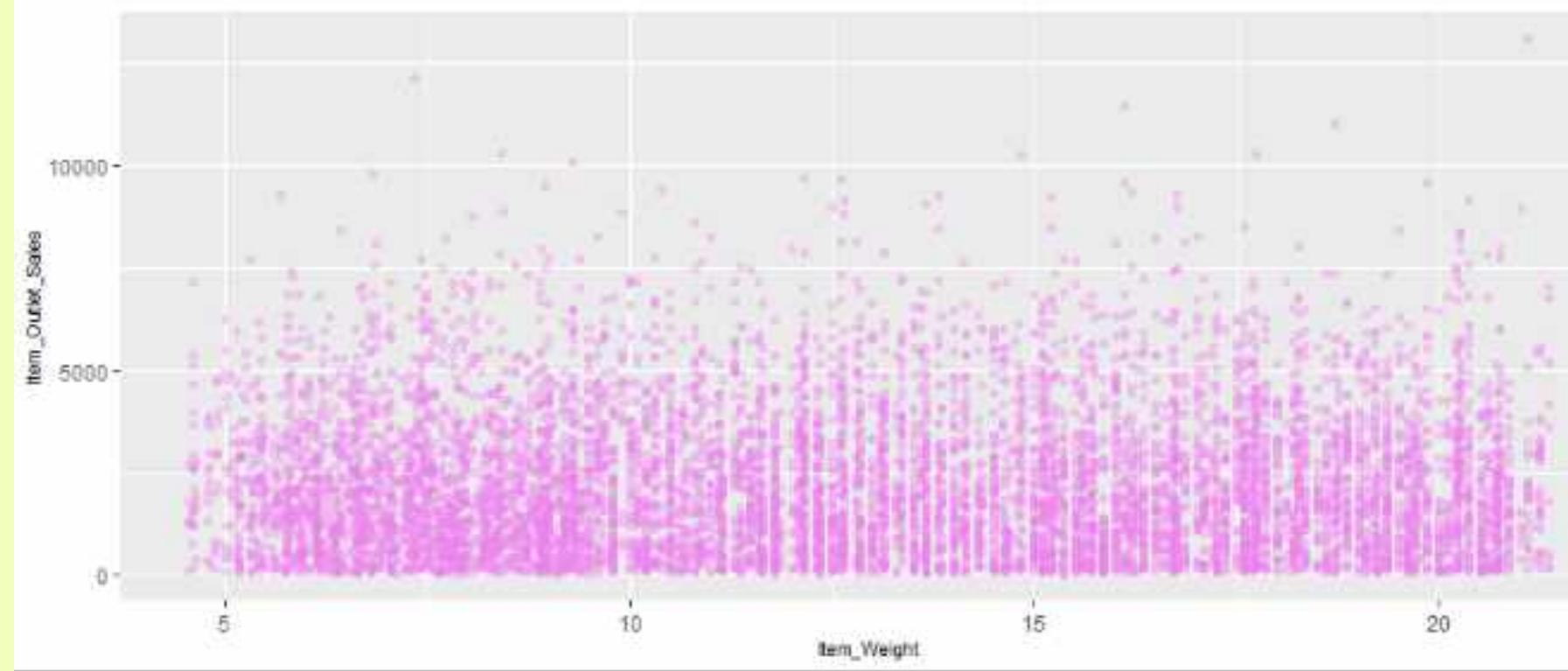


- Item\_Fat\_Content khi biểu diễn có thể thấy:
  - “LF”, “low fat”, “Fow Fat” là cùng 1 loại
  - “reg” và “Regular” cũng cùng 1 loại
- Ta sẽ xử lý được invalid values ngay lập tức

# PHÂN TÍCH ĐƠN BIẾN:



# PHÂN TÍCH ĐA BIỂN:



# XỬ LÝ MISSING VALUES:

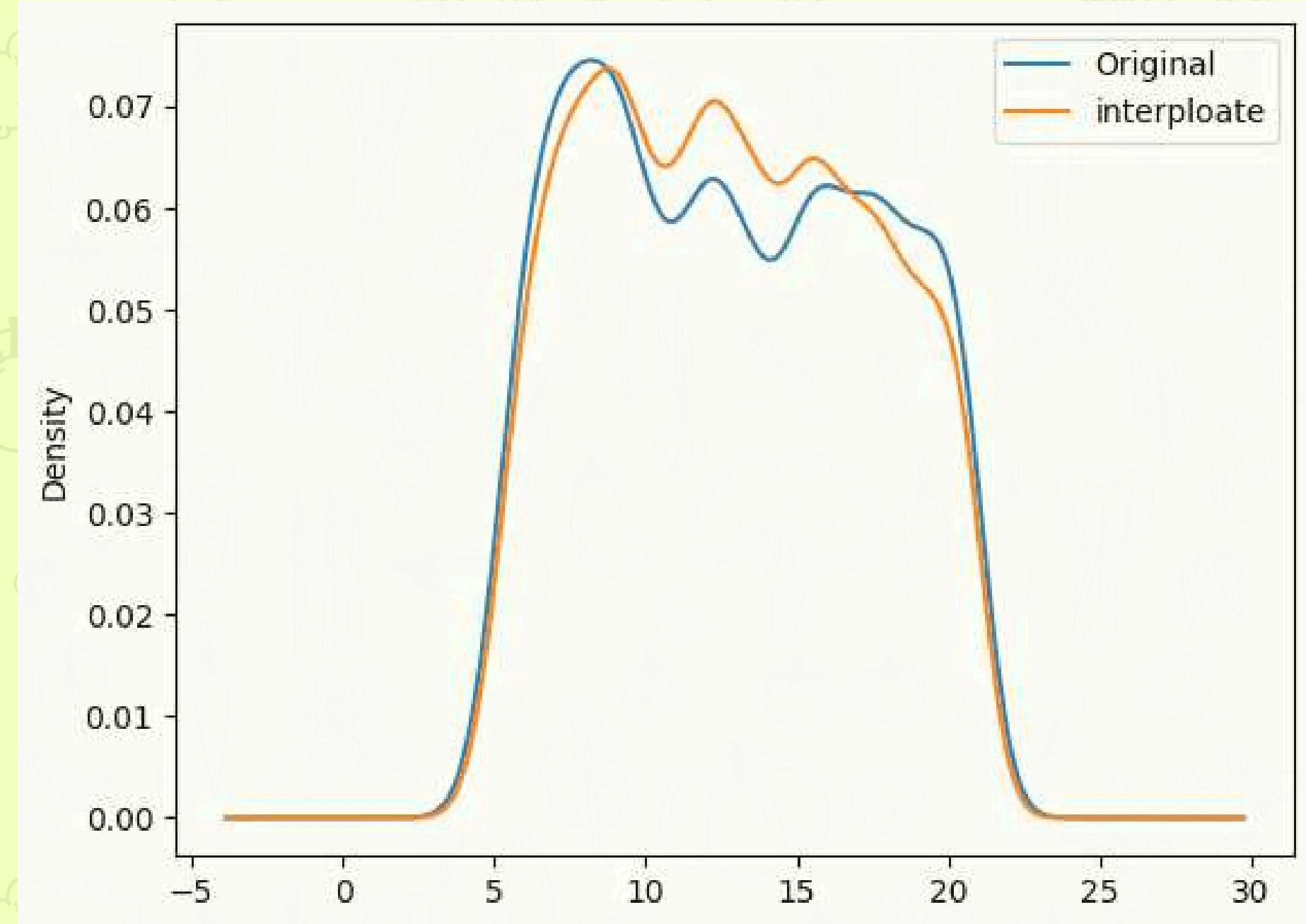
- Sau khi thống kê các giá trị thiếu trong dữ liệu Bigmart:
  - Item\_Weight: 1463 null
  - Outlet\_Size: 2410 null

```
[ ] bigmart_train.isnull().sum()
```

```
Item_Identifier           0  
Item_Weight               1463  
Item_Fat_Content          0  
Item_Visibility            0  
Item_Type                  0  
Item_MRP                   0  
Outlet_Identifier           0  
Outlet_Establishment_Year      0  
Outlet_Size                2410  
Outlet_Location_Type         0  
Outlet_Type                  0  
Item_Outlet_Sales             0  
dtype: int64
```

# XỬ LÝ MISSING VALUES:

- Xử lý missing values biến Item\_Weight
  - Item\_Weight là **biến định lượng**
  - Phương pháp **nội suy tuyến tính**:
    - sử dụng một đường thẳng để ước lượng giá trị thiếu dựa trên các giá trị xung quanh
  - Đánh giá bằng biểu đồ phân phối Kernel Density Estimate (**KDE**):
    - 2 đường cong khá **tương đồng** nên không ảnh hưởng tới dữ liệu



# XỬ LÝ MISSING VALUES:

- Xử lý missing value biến Outlet\_Size:

- Sử dụng 2 biến Outlet\_Size (kích cỡ mặt bằng) và Outlet\_Type (loại cửa hàng)

```
[ ] bigmart_train['Outlet_Size'].value_counts()
```

```
→ Outlet_Size  
Medium    2793  
Small     2388  
High      932  
Name: count, dtype: int64
```

```
[ ] bigmart_train['Outlet_Type'].value_counts()
```

```
→ Outlet_Type  
Supermarket Type1    5577  
Grocery Store        1083  
Supermarket Type3    935  
Supermarket Type2    928  
Name: count, dtype: int64
```

- Tạo bảng pivot để tính **mode** của Outlet\_Size cho mỗi Outlet\_Type

Outlet_Type	Grocery Store	Supermarket Type1	Supermarket Type2	Supermarket Type3
Outlet_Size	Small	Small	Medium	Medium

- Các giá trị Outlet\_Size thiếu được điền bằng mode tương ứng với Outlet\_Type của chúng

# XỬ LÝ INVALID VALUES:

- Kiểm tra các giá trị trong biến Item\_Visibility nhận thấy có 526 giá trị 0.0 trong khi doanh số bán hàng vẫn cao

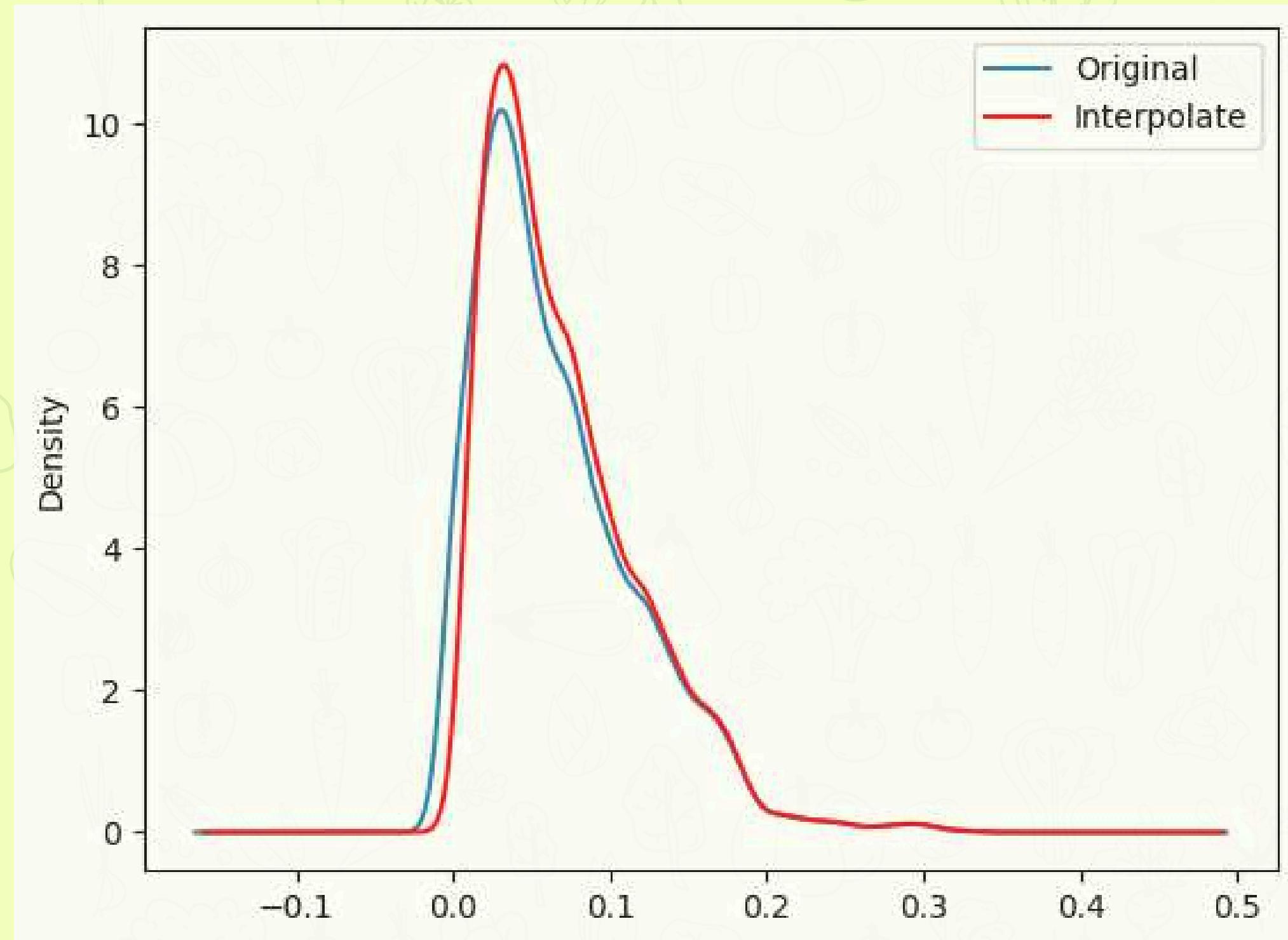
```
bigmart_train['Item_Visibility'].value_counts()
```

Item_Visibility	count
0.000000	526
0.076975	3
0.162462	2
0.076841	2
0.073562	2
...	
0.013957	1
0.110460	1
0.124646	1
0.054142	1
0.044878	1

Name: count, Length: 7880, dtype: int64

# XỬ LÝ INVALID VALUES:

- Xử lý giá trị không hợp lệ:
  - Biến các giá trị 0.0 thành NaN
  - Sử dụng **nội suy tuyến tính** để điền vào missing values.
- Đánh giá bằng biểu đồ phân phối Kernel Density Estimate (KDE):
  - 2 đường cong khá tương đồng nên không ảnh hưởng tới dữ liệu



# BIẾN ĐỔI DỮ LIỆU

- Với biến Outlet\_Establishment\_Year là năm thành lập cửa hàng

Outlet_Identifier	Outlet_Establishment_Year
OUT010	1998
OUT013	1987
OUT017	2007
OUT018	2009
OUT019	1985
OUT027	1985
OUT035	2004
OUT045	2002
OUT046	1997
OUT049	1999



- Biến đổi thành số năm hoạt động = 2013 (thời điểm thu thập DL) - năm thành lập cửa hàng

Outlet_Identifier	Outlet_Establishment_Year
OUT010	15
OUT013	26
OUT017	6
OUT018	4
OUT019	28
OUT027	28
OUT035	9
OUT045	11
OUT046	16
OUT049	14



### 3. XÂY DỰNG & ĐÁNH GIÁ MÔ HÌNH



# LINEAR REGRESSION (ĐA BIẾN):

- Bắt đầu với mô hình tuyến tính đầy đủ các biến
- Kết quả thu được:
  - Có nhiều biến không có ý nghĩa
  - R-squared: 0.5016 tương đối thấp

Call:

```
lm(formula = Item_Outlet_Sales ~ Item_Identifier + Item_Weight +  
  Item_Fat_Content + Item_Visibility + Item_Type + Item_MRP +  
  Outlet_Identifier + Outlet_Establishment_Year + Outlet_Size +  
  Outlet_Location_Type + Outlet_Type, data = data_encoded)
```

Residuals:

Min	1Q	Median	3Q	Max
-4256.8	-753.7	-72.1	629.0	7965.1

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-9.211e+02	1.106e+02	-8.326	< 2e-16 ***
Item_Identifier	-5.403e-03	2.933e-02	-0.184	0.85388
Item_Weight	-4.074e-01	2.818e+00	-0.145	0.88506
Item_Fat_Content	5.108e+01	2.783e+01	1.836	0.06644 .
Item_Visibility	-3.578e+02	2.954e+02	-1.211	0.22579
Item_Type	-4.510e-01	3.140e+00	-0.144	0.88579
Item_MRP	1.557e+01	2.099e-01	74.151	< 2e-16 ***
Outlet_Identifier	5.895e+01	9.360e+00	6.298	3.16e-10 ***
Outlet_Establishment_Year	-1.862e+00	1.755e+00	-1.061	0.28861
Outlet_Size	-1.487e+02	2.480e+01	-5.997	2.10e-09 ***
Outlet_Location_Type	-1.108e+02	3.752e+01	-2.954	0.00314 **
Outlet_Type	8.455e+02	2.667e+01	31.703	< 2e-16 ***

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1205 on 8511 degrees of freedom

Multiple R-squared: 0.5016, Adjusted R-squared: 0.501

F-statistic: 778.8 on 11 and 8511 DF, p-value: < 2.2e-16

# STEP FORWARD:

- Cải thiện mô hình bằng step forward để giảm biến
- Kết quả thu được:
  - Có 3 biến ảnh hưởng tốt tới mô hình:
    - Item\_MRP, Outlet\_Identifier,
    - Item\_Fat\_Content
  - R-squared: 0.5634

Call:

```
lm(formula = Item_Outlet_Sales ~ Item_MRP + Outlet_Identifier +  
    Item_Fat_Content, data = data_encoded)
```

Residuals:

Min	1Q	Median	3Q	Max
-4330.8	-678.2	-89.3	570.2	7929.6

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-1868.7540	56.0003	-33.370	<2e-16 ***
Item_MRP	15.5587	0.1963	79.260	<2e-16 ***
Outlet_Identifier1	1949.7252	60.4961	32.229	<2e-16 ***
Outlet_Identifier2	2022.1576	60.5697	33.386	<2e-16 ***
Outlet_Identifier3	1642.0046	60.5448	27.120	<2e-16 ***
Outlet_Identifier4	16.8150	68.5929	0.245	0.8064
Outlet_Identifier5	3369.8798	60.4600	55.737	<2e-16 ***
Outlet_Identifier6	2063.2470	60.5220	34.091	<2e-16 ***
Outlet_Identifier7	1850.7850	60.5327	30.575	<2e-16 ***
Outlet_Identifier8	1918.3875	60.5208	31.698	<2e-16 ***
Outlet_Identifier9	2016.1679	60.5204	33.314	<2e-16 ***
Item_Fat_Content1	50.3786	25.5804	1.969	0.0489 *

---

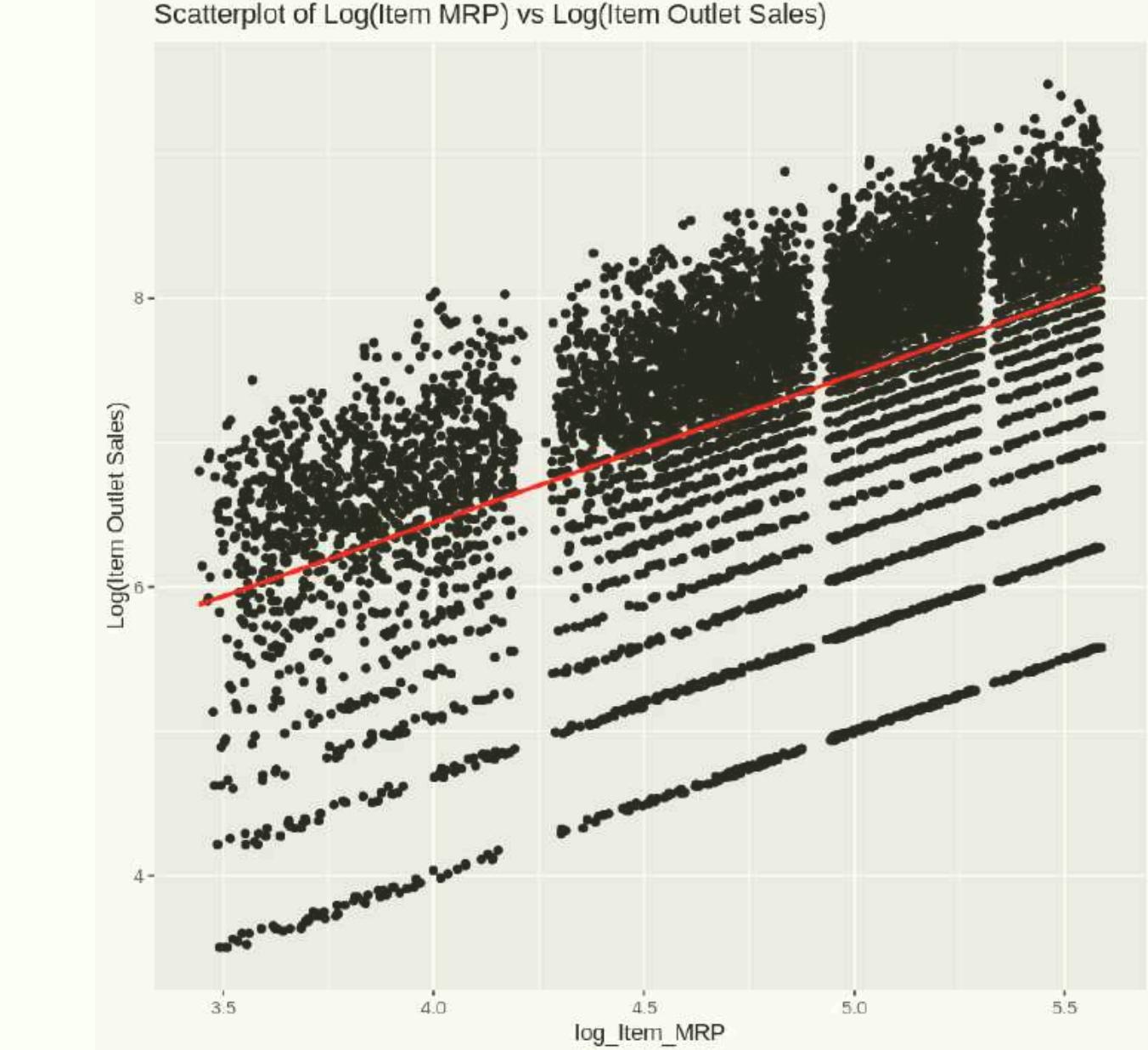
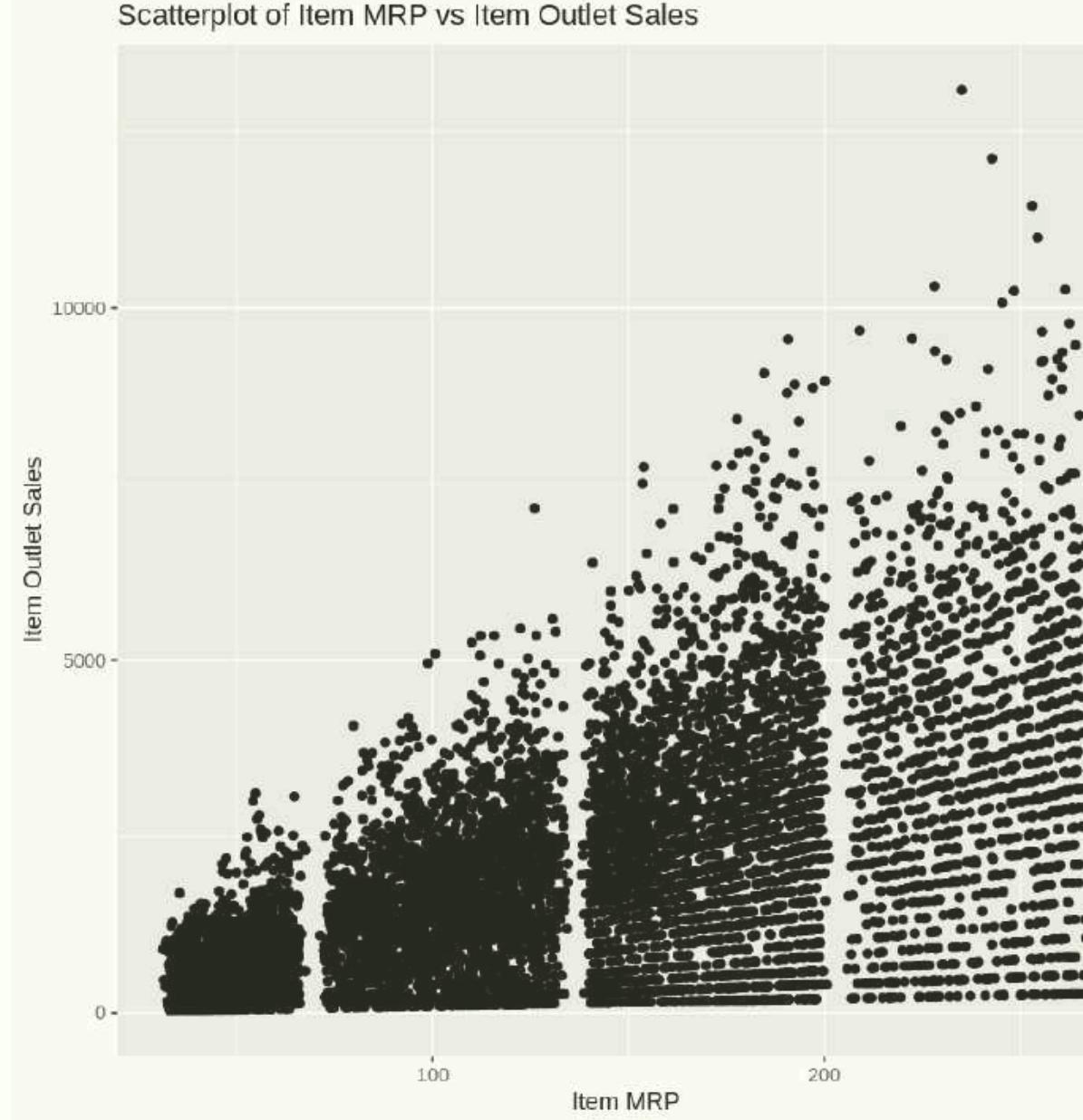
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1128 on 8511 degrees of freedom

Multiple R-squared: 0.5634, Adjusted R-squared: 0.5628

F-statistic: 998.5 on 11 and 8511 DF, p-value: < 2.2e-16

# BIẾN ĐỔI HÀM LOG:



- Cải thiện hiệu suất mô hình bằng cách biến đổi cả 2 biến Item\_Outlet\_Sales và Item\_MRP thành **hàm log**

# BIẾN ĐỔI HÀM LOG:

- Kết quả thu được :
  - Có 2 biến ảnh hưởng tốt tới mô hình:
    - Item\_MRP, Outlet\_Identifier,
  - R-squared: 0.7418 đã tăng lên khá đáng kể

Call:

```
lm(formula = log_Item_Outlet_Sales ~ log_Item_MRP + Outlet_Identifier +  
Item_Fat_Content, data = data_encoded)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.11446	-0.27284	0.05591	0.36337	1.36104

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	0.60037	0.05600	10.721	<2e-16 ***
log_Item_MRP	1.02203	0.01065	95.956	<2e-16 ***
Outlet_Identifier1	1.93316	0.02773	69.714	<2e-16 ***
Outlet_Identifier2	1.99230	0.02776	71.759	<2e-16 ***
Outlet_Identifier3	1.78709	0.02775	64.394	<2e-16 ***
Outlet_Identifier4	0.02470	0.03144	0.786	0.432
Outlet_Identifier5	2.49552	0.02771	90.047	<2e-16 ***
Outlet_Identifier6	2.01201	0.02774	72.525	<2e-16 ***
Outlet_Identifier7	1.91790	0.02775	69.121	<2e-16 ***
Outlet_Identifier8	1.95323	0.02774	70.409	<2e-16 ***
Outlet_Identifier9	1.99319	0.02774	71.850	<2e-16 ***
Item_Fat_Content1	0.01326	0.01173	1.131	0.258

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5172 on 8511 degrees of freedom

Multiple R-squared: 0.7418, Adjusted R-squared: 0.7414

F-statistic: 2223 on 11 and 8511 DF, p-value: < 2.2e-16

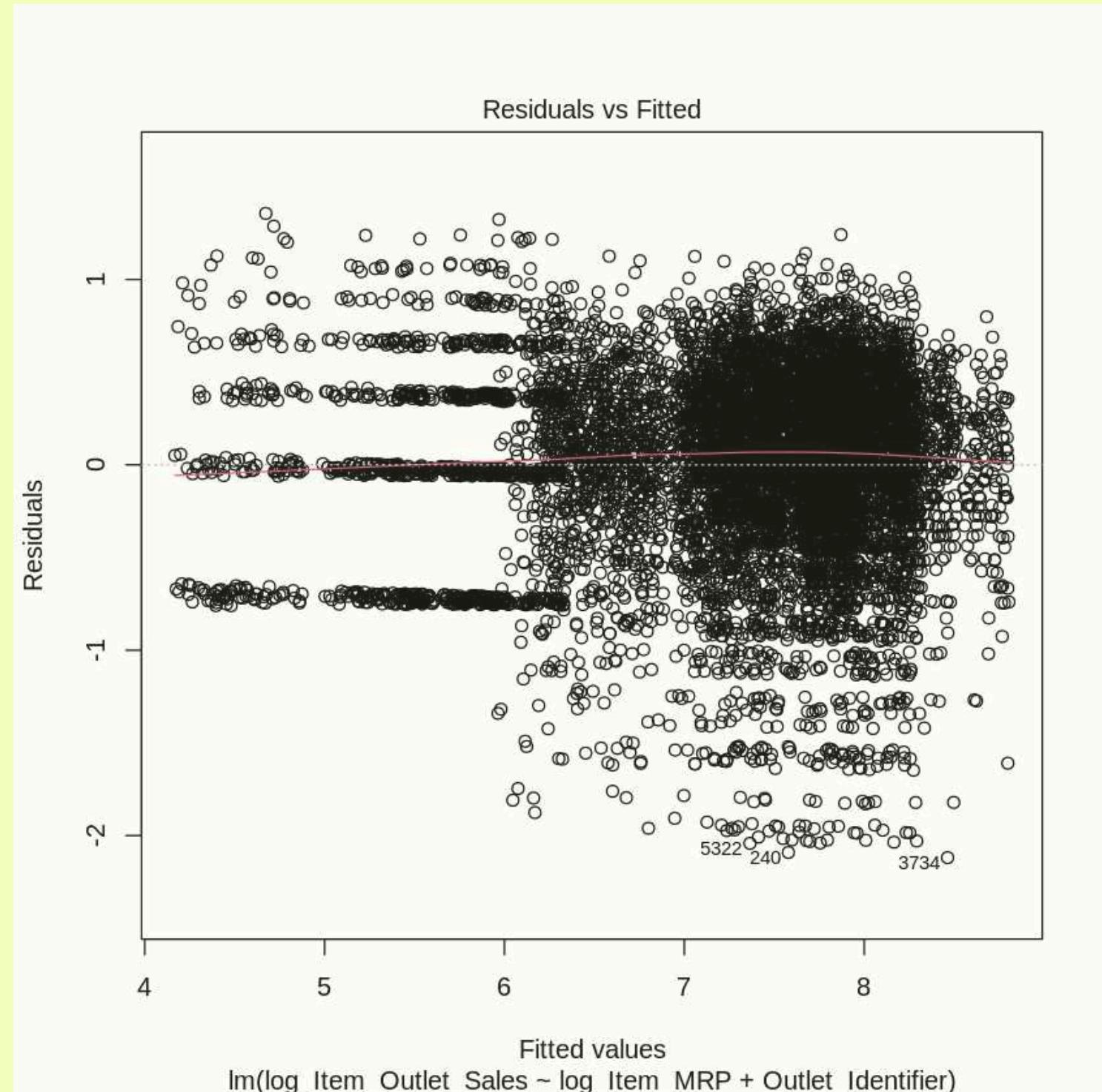
# ĐÁNH GIÁ MÔ HÌNH LOG:

Kiểm tra giả thiết tuyến tính của dữ liệu và giả thiết phần dư có trung bình bằng 0:

- **H<sub>0</sub>**: Trung bình của phần dư (residuals) bằng 0.
- **H<sub>1</sub>**: Trung bình của phần dư khác 0.
- **p-value = 1**: không thể bác bỏ giả thuyết giá trị trung bình của dư thừa là bằng 0

```
One Sample t-test

data: model_step_log$residuals
t = -1.2842e-15, df = 8522, p-value = 1
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
-0.01097522 0.01097522
sample estimates:
mean of x
-7.190364e-18
```



# ĐÁNH GIÁ MÔ HÌNH LOG:

Kiểm tra giả thiết phần dư có phân phối chuẩn:

- H<sub>0</sub>: Dữ liệu có phân phối chuẩn
- H<sub>1</sub>: Dữ liệu không có phân phối chuẩn

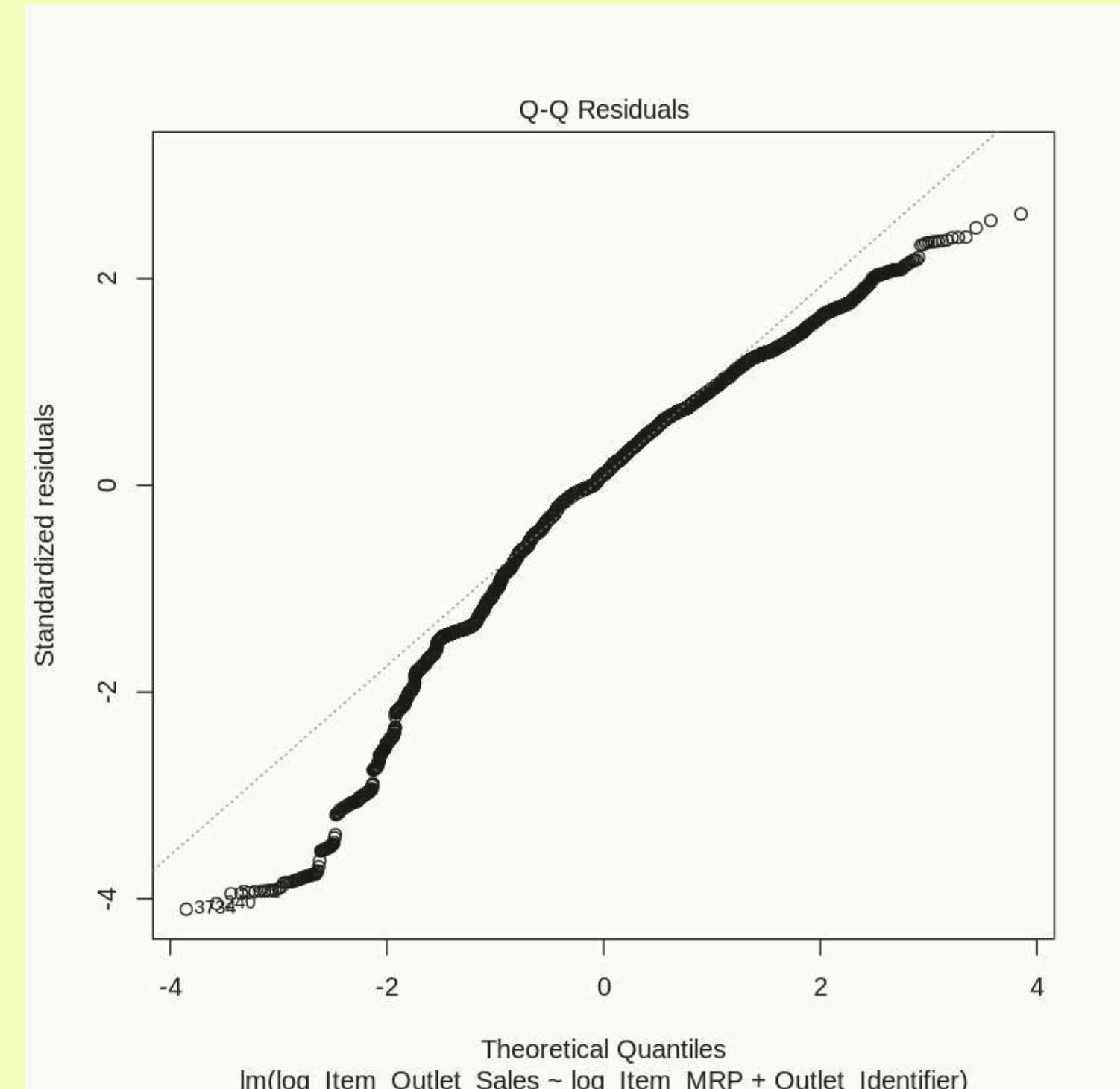
Theo quan sát và đánh giá:

- **p-value = 2.2e-16**: phần dư từ mô hình **không** tuân theo phân phối chuẩn

```
shapiro.test(sample(model_step_log$residuals, 5000))

Shapiro-Wilk normality test

data: sample(model_step_log$residuals, 5000)
W = 0.96324, p-value < 2.2e-16
```



# ĐÁNH GIÁ MÔ HÌNH LOG:

Kiểm tra giả thiết phương sai không đồng nhất:

- H<sub>0</sub>: phương sai không đổi
- H<sub>1</sub>: Phương sai thay đổi

Theo quan sát và đánh giá:

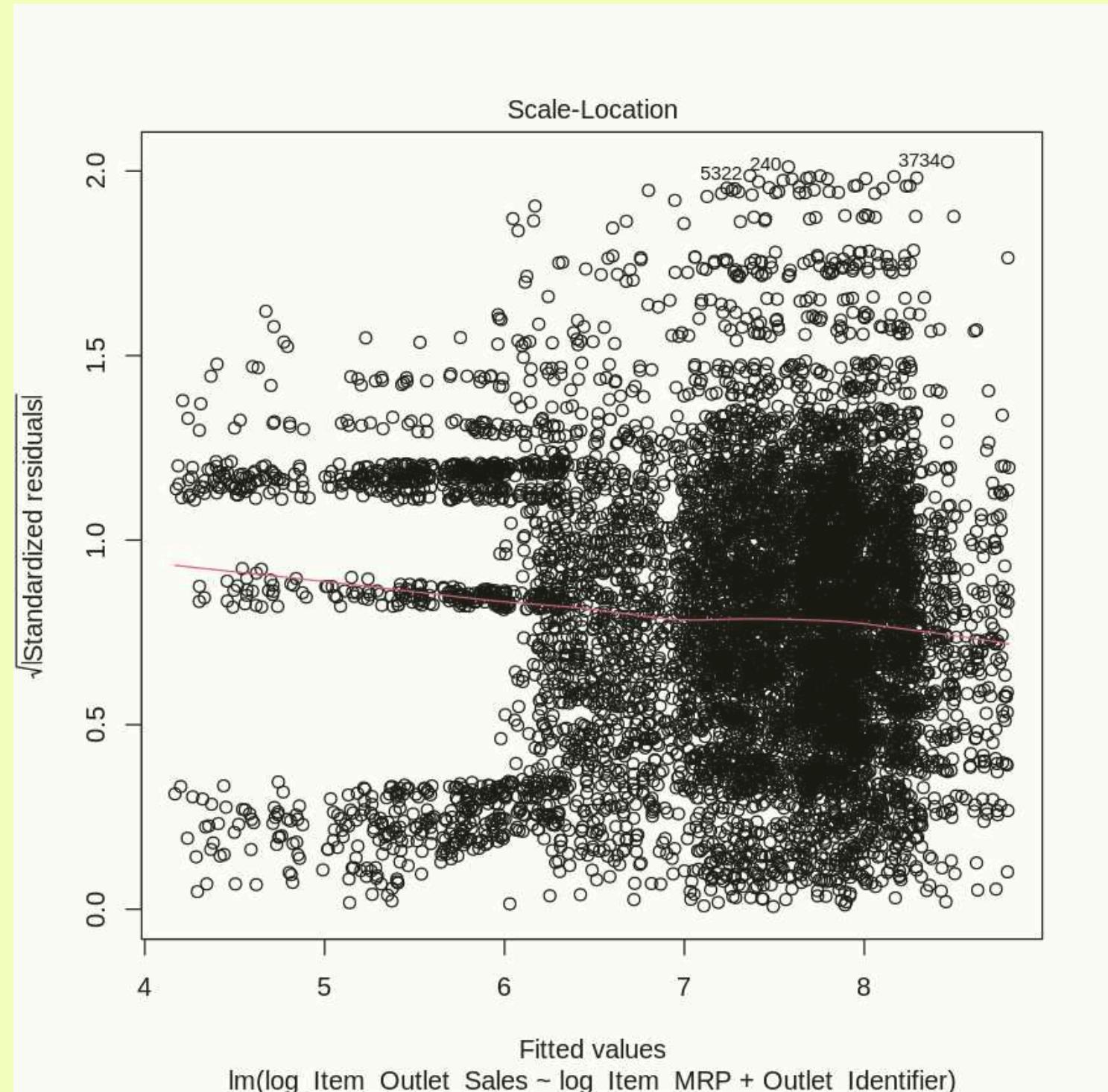
- **p-value = 4.9773e-08**: phương sai của residual thay đổi khi giá trị của biến độc lập thay đổi

```
ncvTest(model_step_log)
```

Non-constant Variance Score Test

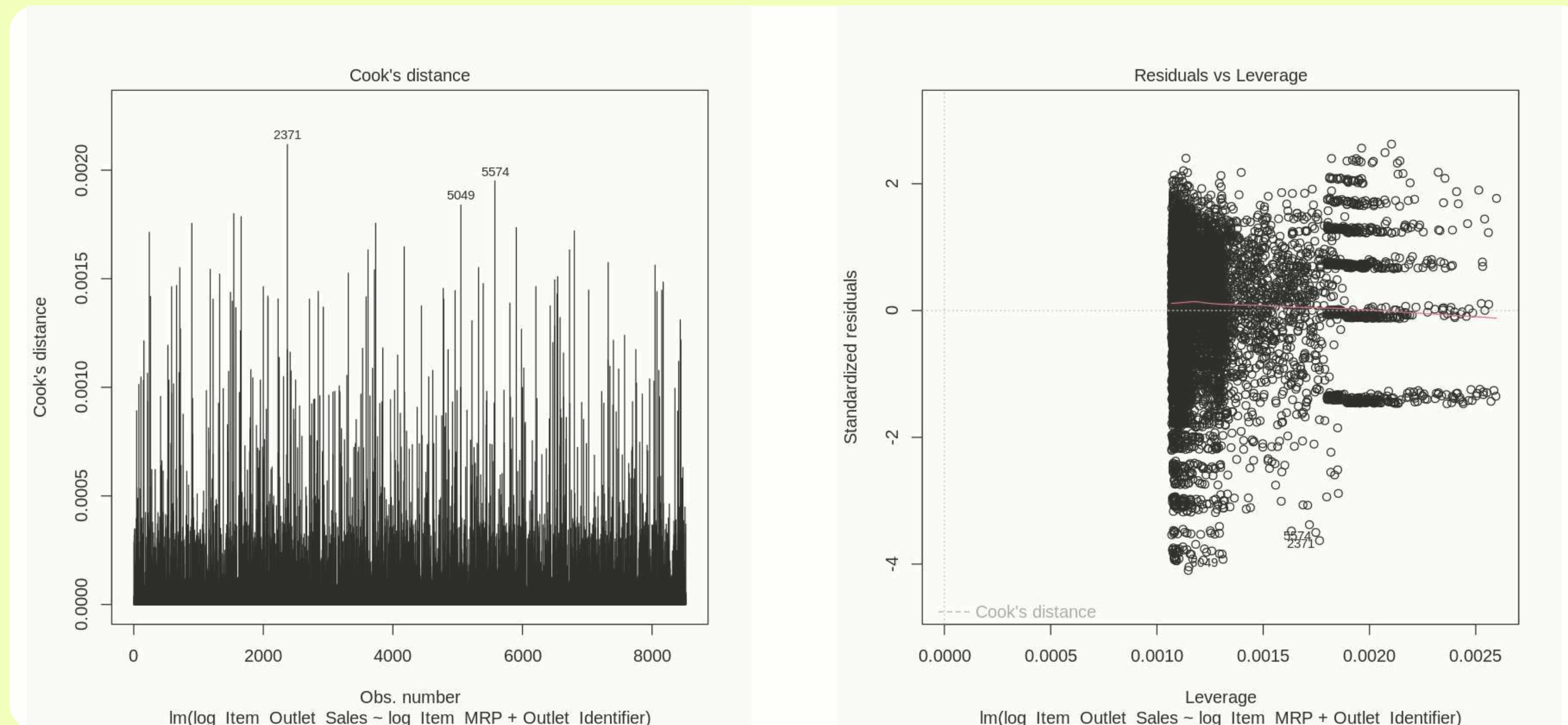
Variance formula: ~ fitted.values

Chisquare = 29.7256, Df = 1, p = 4.9773e-08



# ĐÁNH GIÁ MÔ HÌNH LOG:

- Đồ thị thứ tư chỉ ra có các quan trắc thứ 2371, 5049 và 5574 có thể là các điểm có ảnh hưởng cao trong bộ dữ liệu.



# ĐÁNH GIÁ MÔ HÌNH LOG:

Kiểm tra giả thuyết Giữa các biến độc lập không có mối quan hệ đa cộng tuyến

- vif < 10 cho thấy các biến độc lập không có đa cộng tuyến

```
vif(model_step_log)
```

A matrix: 2 × 3 of type dbl

	GVIF	Df	GVIF <sup>(1/(2*Df))</sup>
--	------	----	----------------------------

<b>log_Item_MRP</b>	1.000318	1	1.000159
---------------------	----------	---	----------

<b>Outlet_Identifier</b>	1.000318	9	1.000018
--------------------------	----------	---	----------

# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

```
ancova <- aov(log_Item_Outlet_Sales ~ log_Item_MRP + Outlet_Identifier, data = data_processed)
summary(ancova)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
log_Item_MRP	1	2478	2478.0	9264	<2e-16 ***
Outlet_Identifier	9	4061	451.2	1687	<2e-16 ***
Residuals	8512	2277	0.3		
---					
Signif. codes:	0	***	0.001 **	0.01 *	0.05 .
					0.1 ‘ ’ 1

- Cả log\_Item\_MRP và Outlet\_Identifier đều có p-value rất nhỏ (<2-e-16) ngụ ý rằng cả hai biến đều có ảnh hưởng đáng kể đến log\_Item\_Outlet\_Sales.

# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

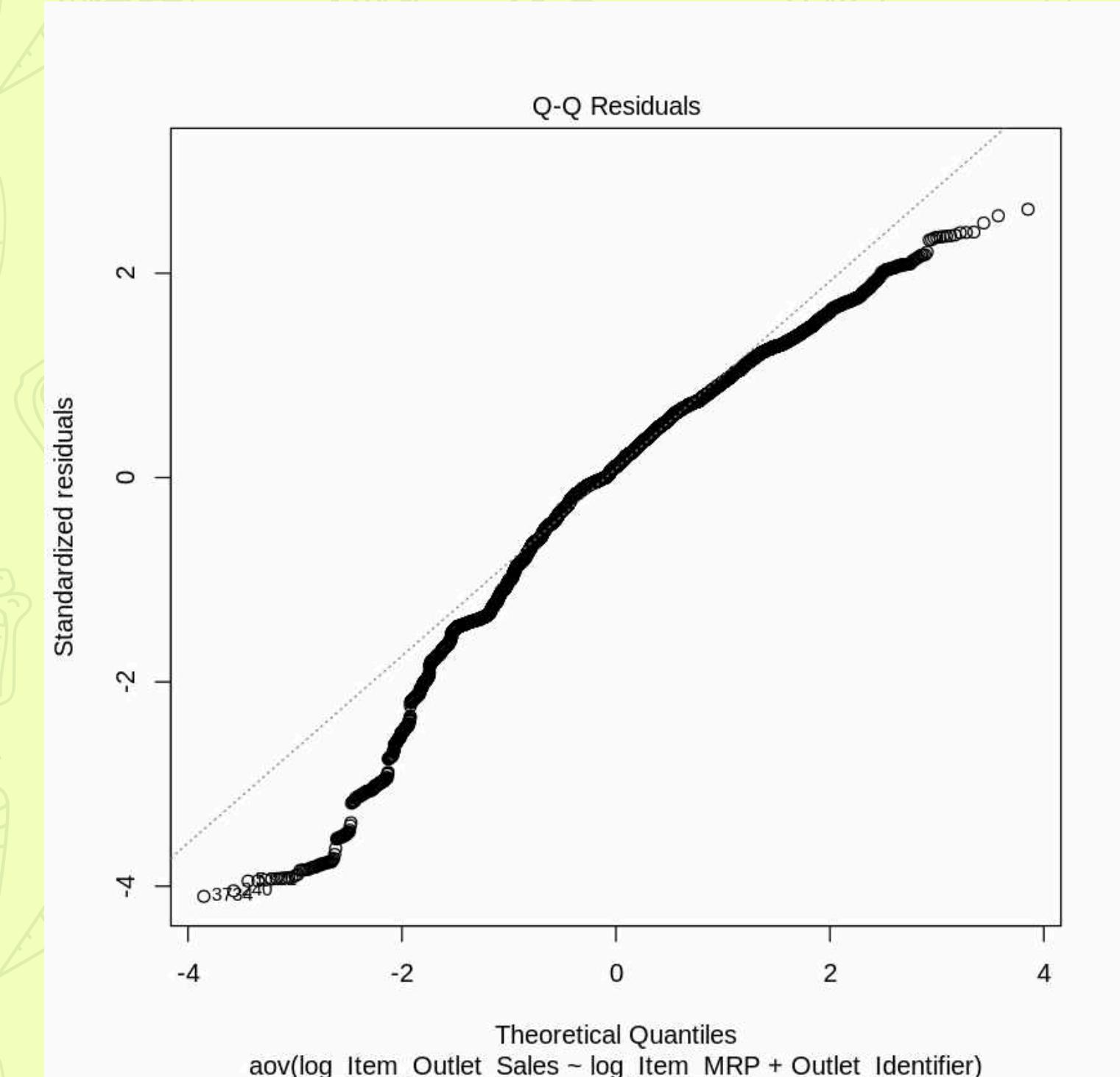
Kiểm tra giả thiết phần dư có phân phối chuẩn:

- **p-value = 2.2e-16**: phần dư từ mô hình ANCOVA **không** tuân theo phân phối chuẩn

```
plot(ancova,which=2)  
shapiro.test(sample(ancova$residuals,5000))
```

Shapiro-Wilk normality test

```
data: sample(ancova$residuals, 5000)  
W = 0.96131, p-value < 2.2e-16
```



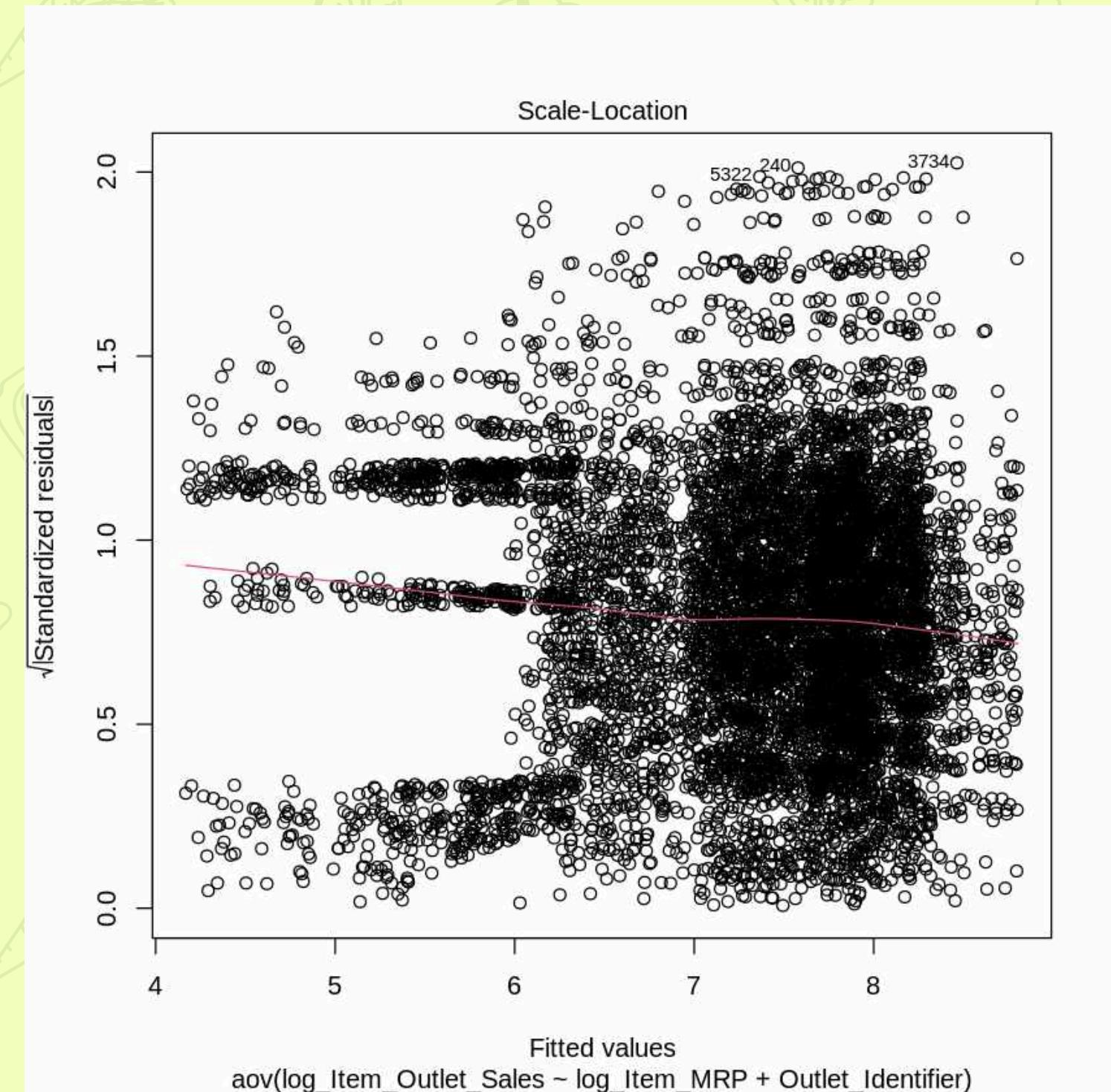
# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

Kiểm tra giả thiết phuơng sai không đồng nhất:

- **p-value = 6.620172e-16**: có sự khác biệt đáng kể về phuơng sai của phần dư giữa các nhóm cửa hàng bán lẻ.

```
plot(ancova,which=3)
levene_test(data = data.frame(ancova$residuals,
                             Outlet_Identifier = data_processed$Outlet_Identifier),
            ancova$residuals ~ Outlet_Identifier)

Warning message in leveneTest.default(y = y, group = group, ...):
“group coerced to factor.”
# A tibble: 1 × 4
  df1   df2 statistic      p
  <int> <int>    <dbl>    <dbl>
1     9   8513  10.26736 6.620172e-16
```



# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

Kiểm tra giả thiết sự độc lập của các phần dư:

- D-W Statistic = 2.020271, gần với 2.
- p-value = 0.348 ( $>0.05$ )

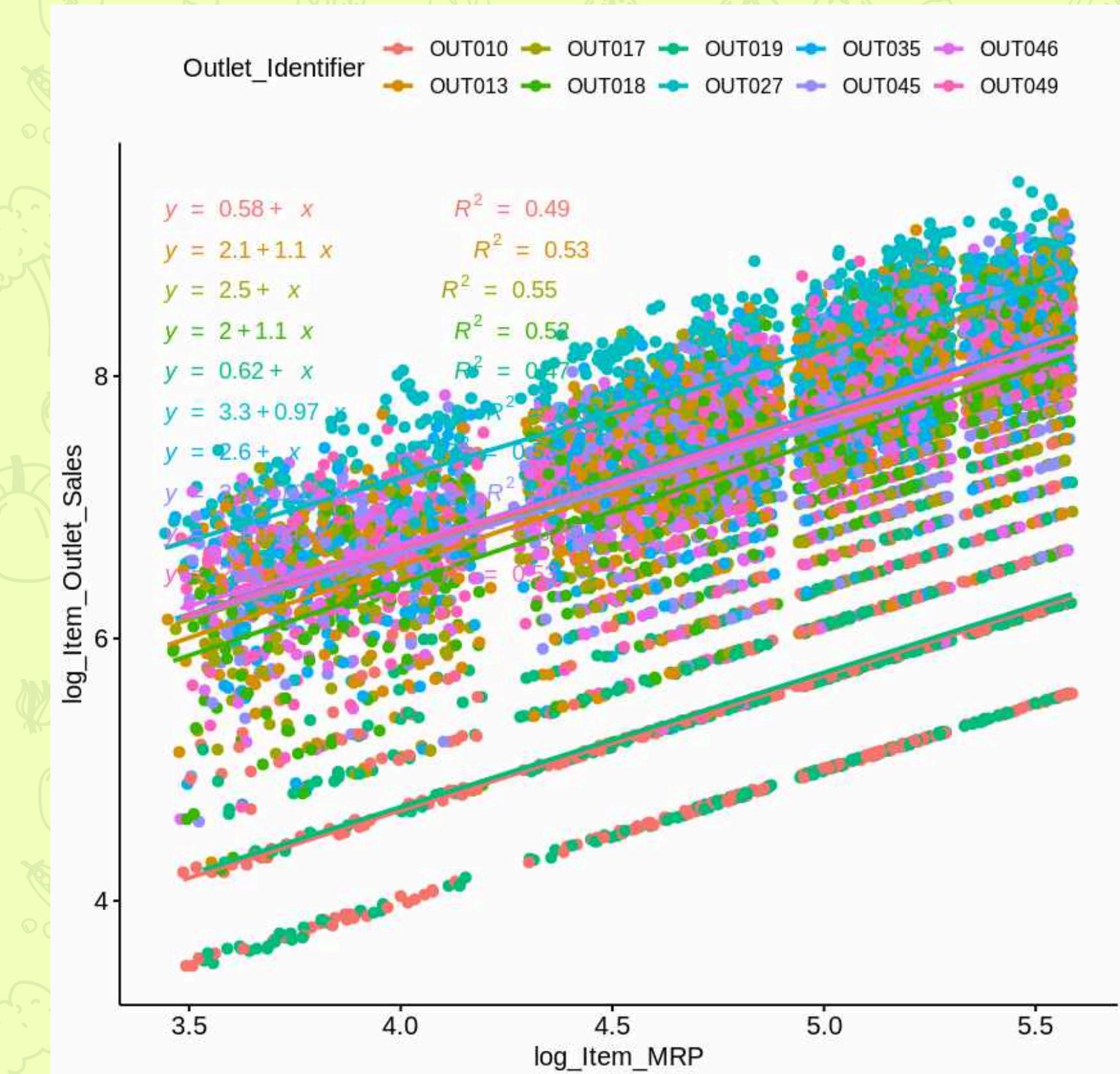
=> Không có sự tự tương quan đáng kể giữa các phần dư trong mô hình

```
#Kiểm tra sự độc lập của phần dư  
durbinWatsonTest(ancova)
```

```
lag Autocorrelation D-W Statistic p-value  
1      -0.01016045      2.020271    0.348  
Alternative hypothesis: rho != 0
```

# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

- Biểu đồ scatter plot với đường hồi quy tuyến tính cho từng nhóm Outlet\_Identifier giúp kiểm tra mối quan hệ tuyến tính giữa hai biến log\_Item\_MRP và log\_Item\_Outlet\_Sales cho từng loại cửa hàng khác nhau.



# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

```
mod.full <- aov(log_Item_Outlet_Sales ~ log_Item_MRP * Outlet_Identifier,  
                  data = data_processed)  
summary(mod.full)
```

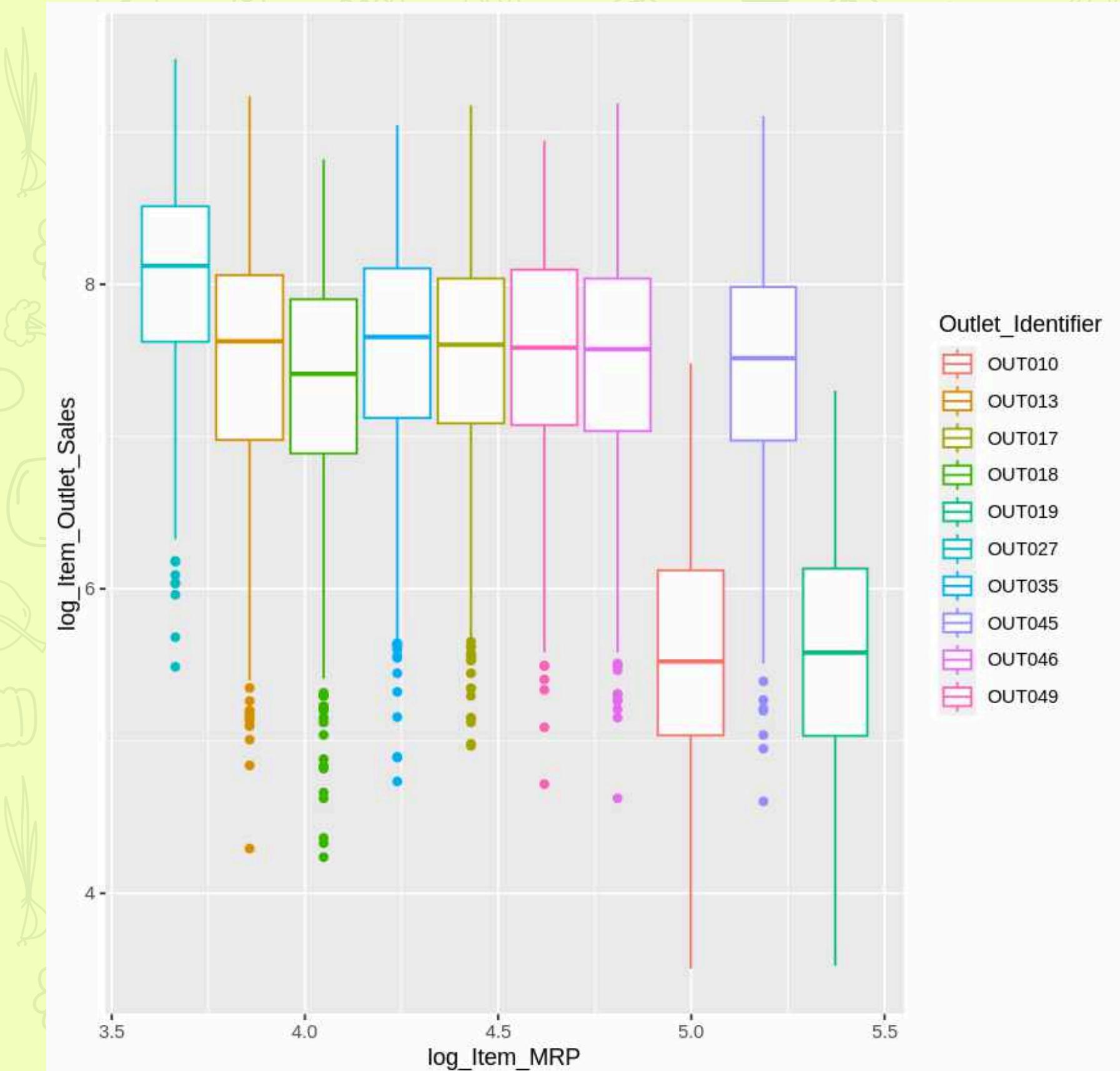
	Df	Sum Sq	Mean Sq	F value	Pr(>F)		
log_Item_MRP	1	2478	2478.0	9280.711	< 2e-16 ***		
Outlet_Identifier	9	4061	451.2	1689.968	< 2e-16 ***		
log_Item_MRP:Outlet_Identifier	9	6	0.7	2.702	0.00386 **		
Residuals	8503	2270	0.3				
---							
Signif. codes:	0	***	0.001	**	0.01 *	0.05 .	0.1 ‘ ’ 1

Kiểm tra **tính đồng nhất của hệ số góc** trong mô hình tương tác:

- Hệ số tương tác giữa **log\_Item\_MRP** và **Outlet\_Identifier** có giá trị **p-value = 0.00386**, nhỏ hơn mức ý nghĩa thông thường (0.05).  
=> Tính đồng nhất của hệ số góc không được chấp nhận

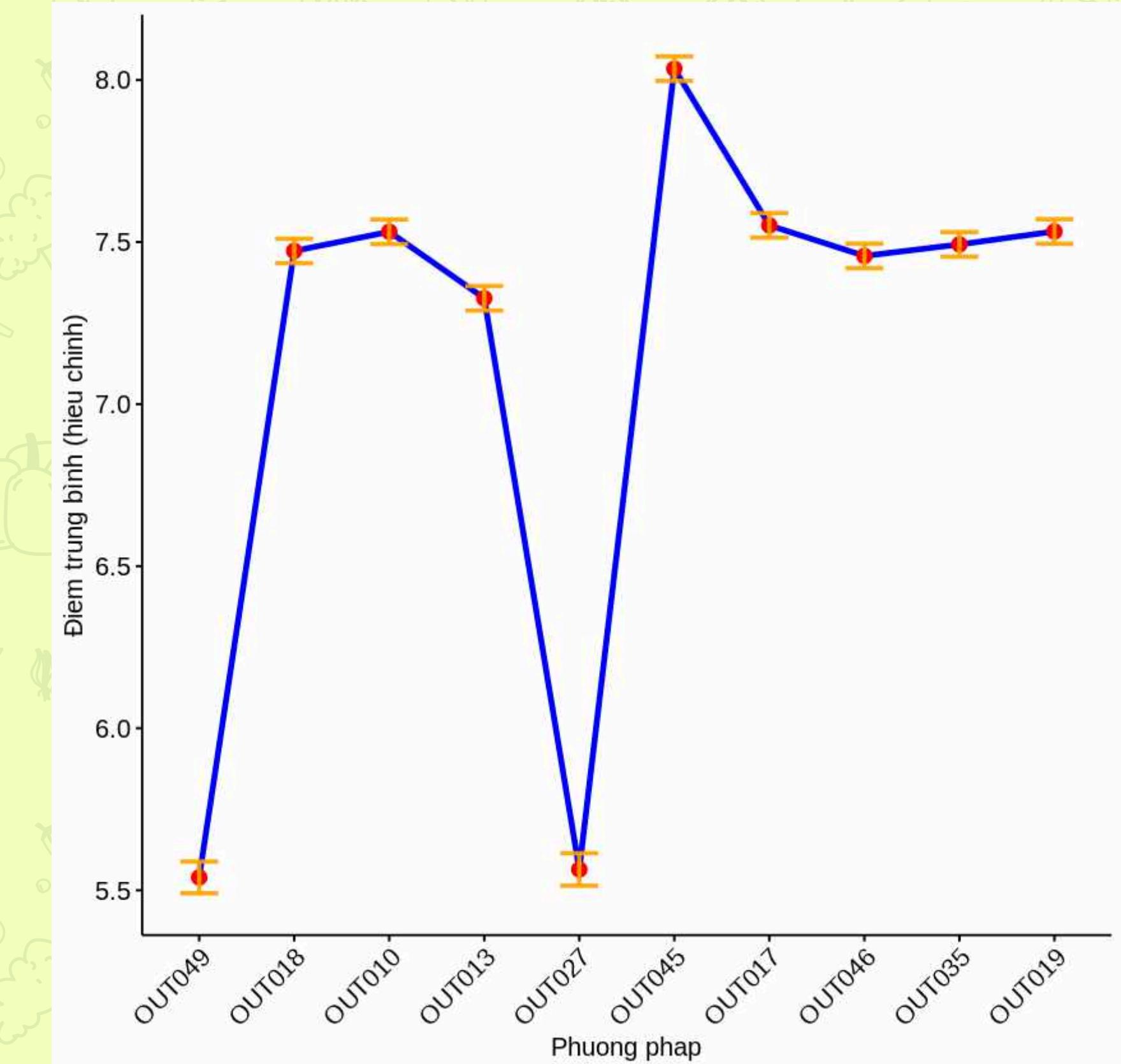
# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

- Biểu đồ hộp sẽ hiển thị phân phối của log\_Item\_Outlet\_Sales dưới dạng các hộp, các đường median, và các outliers cho mỗi nhóm Outlet\_Identifier.
- OUT019 và OUT010 có sự khác biệt lớn giữa các nhóm, điều này có thể ảnh hưởng đến mô hình.
- Nhưng có thể thấy các nhóm còn lại có nhiều outliers

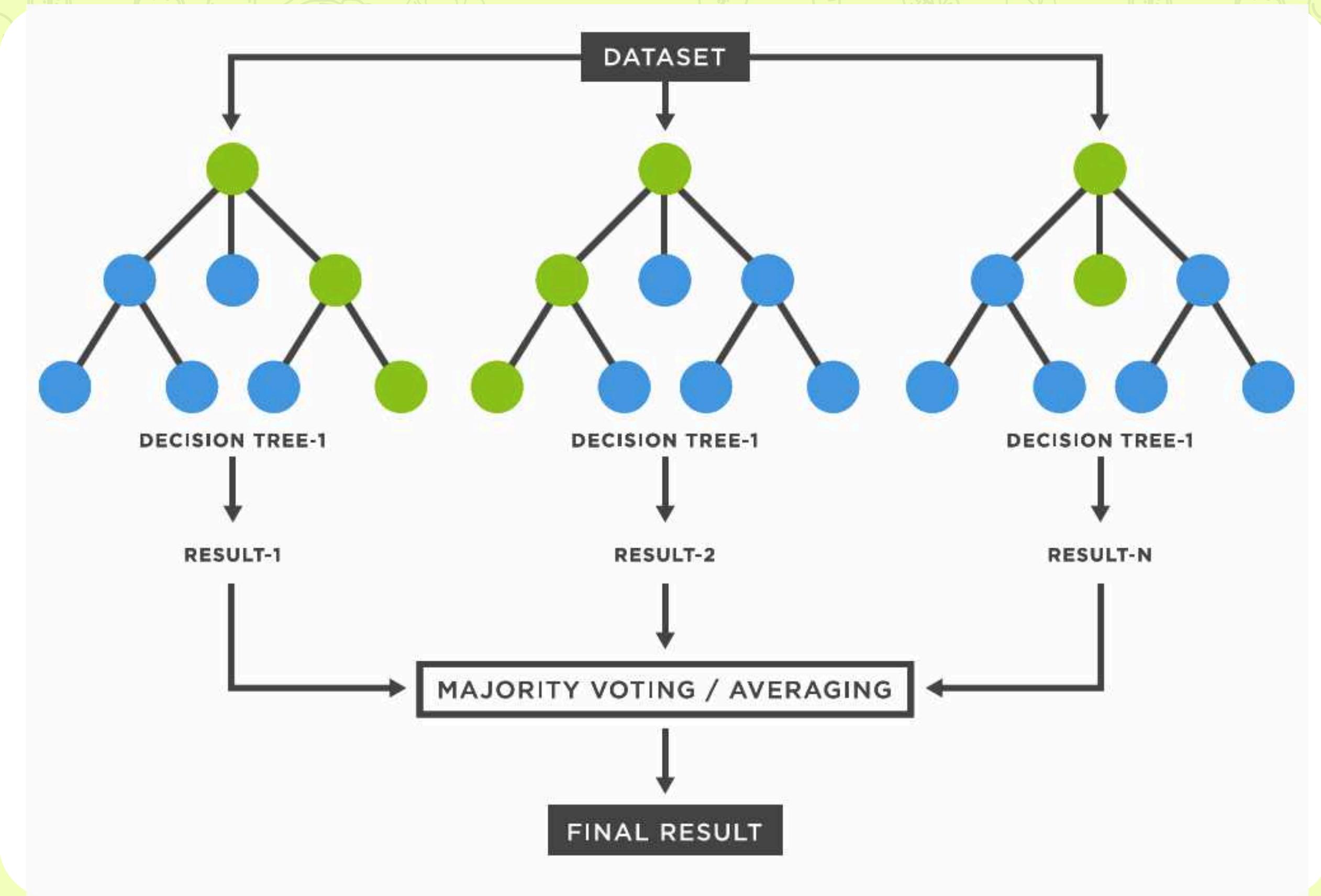


# PHÂN TÍCH ANCOVA CHO MÔ HÌNH LOG

- Biểu đồ hiển thị giá trị trung bình (đã được điều chỉnh) của biến phụ thuộc (**log\_Item\_Outlet\_Sales**) theo từng nhóm của biến độc lập (**Outlet\_Identifier**)
- Tại các cửa hàng OUT049 và OUT027 có ảnh hưởng kém doanh số bán hàng hơn so với 8 cửa hàng khác.
- Trong đó OUT045 có ảnh hưởng đến doanh số bán hàng cao nhất



# RANDOM FOREST:



# FEATURE SELECTION:

- Áp dụng mô hình XGBoost để tìm ra các đặc trưng quan trọng ảnh hưởng đến doanh số bán hàng
- Chọn ra top 3 đặc trưng quan trọng nhất để tiến hành xây dựng mô hình Random Forest

	feature	XGBRF_importance
8	Outlet_Type	0.867226
4	Outlet_Identifier	0.057199
9	log_Item_MRP	0.047209
5	Outlet_Establishment_Year	0.020499
7	Outlet_Location_Type	0.002377
6	Outlet_Size	0.002179
0	Item_Weight	0.000895
2	Item_Visibility	0.000895
3	Item_Type	0.000884
1	Item_Fat_Content	0.000637

# CẢI TIẾN RANDOM FOREST:

- Sau khi dùng grid\_search để tìm tham số tốt nhất cho mô hình với fold cross-validation =10

```
rf = RandomForestRegressor(n_estimators = 1500, max_depth = None,  
                          min_samples_leaf = 40, criterion = 'squared_error',  
                          min_samples_split = 5, max_features = 'sqrt',  
                          max_leaf_nodes = None,  
                          random_state = 42)
```

# CẢI TIẾN RANDOM FOREST:

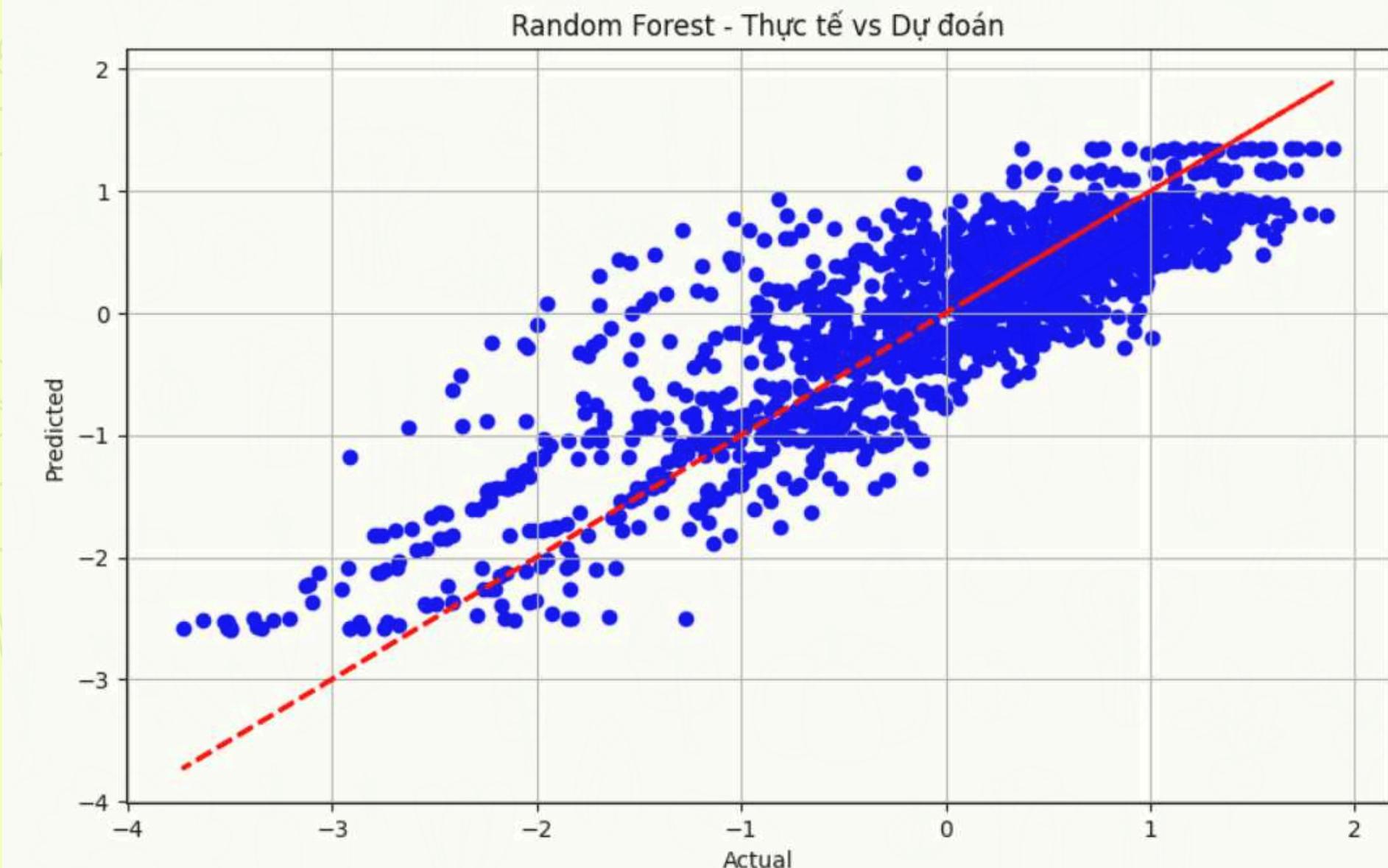
Kết quả thu được mô hình Random Forest khá tốt:

- **Mean Squared Error (MSE)**: 0.268
- **Mean Absolute Error (MAE)**: 0.398
- **R-squared (R<sup>2</sup>)**: 0.744

Qua biểu đồ quan sát được:

- Sự phân tán của các điểm xung quanh đường đỏ, tuy nhiên chúng vẫn tương đối gần với đường này,

Test MSE: 0.2678575800235008  
Test MAE: 0.3978375319583195  
Test R<sup>2</sup>: 0.7436579524409194





## 4. KẾT LUẬN & ĐỊNH HƯỚNG



# ĐÁNH GIÁ TỔNG QUÁT CHO MÔ HÌNH LOG

Tính phù hợp của mô hình:

- $R\text{-squared} = 0.7417$ : Chỉ số R-squared đã được cải thiện rõ rệt

Ý nghĩa thống kê của các biến độc lập:

- log\_Item\_MRP: Có ý nghĩa thống kê cao với **p-value < 2.2e-16**
- Outlet\_Identifier: Các hạng mục trong biến này đều có ý nghĩa thống kê cao (**p-value < 2.2e-16**), ngoại trừ Outlet\_Identifier4

Vi phạm giả định:

- Non-constant Variance Score Test: Cho thấy phương sai không đồng nhất với **p-value = 4.9773e-08**.
- Shapiro-Wilk Normality Test: Phần dư không tuân theo phân phối chuẩn với **p-value < 2.2e-16**.

Đa cộng tuyến:

- VIF: Giá trị **VIF rất thấp (gần 1)**, chỉ ra rằng không có vấn đề đa cộng tuyến giữa các biến độc lập.

# ĐÁNH GIÁ TỔNG QUÁT CHO MÔ HÌNH LOG

	Linear Regression	Random Forest
R^2	74,18%	74,36%
MSE	0.269	0.268
MAE	0.399	0.398

- Mô hình Random Forest có một chút cải thiện nhỏ về MSE và R<sup>2</sup> so với mô hình tuyến tính, nhưng sự khác biệt này không đáng kể.
- Random Forest có độ phức tạp cao hơn Linear Regression.

# ĐỊNH HƯỚNG ĐỂ CẢI THIỆN MÔ HÌNH

- Xem xét **loại bỏ outliers** hoặc xử lý các điểm dữ liệu bất thường để cải thiện sự phù hợp của mô hình.
- Xây dựng thêm các **mô hình khác**: mô hình phi tuyến tính, mô hình mạng nơ-ron (Deep Learning)



