Yonder Technical Test Analysis - Shahin Hussain

January 7, 2025

This notebook presents the results of the analysis conducted for the Yonder Technical Test. The tasks include:

- Exploratory analysis to answer specific questions about the dataset.
- A regression analysis to identify drivers influencing brand recommendations.

Goals:

- Provide insights into brand perceptions and recommendations.
- Identify key drivers of recommendations for Charlotte Tilbury.
- Suggest actionable recommendations based on findings.

Data Overview: - Import libraries & export relevant meta data from .sav file to .txt file.

[13]:
```python
import pandas as pd
import pyreadstat

data, meta = pyreadstat.read_sav("Test Data.sav")

# Prepare metadata for saving
metadata_content = []

# Column names and their labels
metadata_content.append("Variable Names and Labels:\n")
for var_name, var_label in zip(meta.column_names, meta.column_labels):
    metadata_content.append(f"{var_name}: {var_label}\n")

# Value labels
metadata_content.append("\nValue Labels:\n")
for label_set, label_dict in meta.value_labels.items():
    metadata_content.append(f"\n{label_set}:\n")
    for value, label in label_dict.items():
        metadata_content.append(f"  {value}: {label}\n")

# Save the metadata to a text file
metadata_file_path = "metadata_output.txt"
with open(metadata_file_path, "w") as file:
    file.writelines(metadata_content)

print(f"Metadata successfully exported to {metadata_file_path}")
```

```
data.iloc[:, :8].head()
```

```
Metadata successfully exported to metadata_output.txt
```

[13]:

	respid	q01	q02	q03_uk	q03_us	country	q04_uk	q04_us
0	4400136.0	6.0	2.0	11.0	NaN	1.0	6.0	NaN
1	4400138.0	4.0	2.0	9.0	NaN	1.0	6.0	NaN
2	4400140.0	5.0	2.0	9.0	NaN	1.0	6.0	NaN
3	4400142.0	4.0	2.0	9.0	NaN	1.0	6.0	NaN
4	4400144.0	2.0	2.0	8.0	NaN	1.0	2.0	NaN

[2]:

```
# Dataset Overview
print(f"Dataset Shape: {data.shape}")
print("\nNull values:")
print(data.isnull().sum())
```

```
Dataset Shape: (1233, 1212)

Null values:
respid         0
q01            0
q02            0
q03_uk       733
q03_us       500
            …
ukq26_12    1166
ukq26_13     964
ukq26_14     957
ukq26_15    1121
weight         0
Length: 1212, dtype: int64
```

Note on null values: We will focus more on the null values on a task specific basis. You'll see why.

1 Task 1: Exploratory Analysis

1.1 Subtask 1a: Bobbi Brown Value for Money

Question: How many respondents in the UK think Bobbi Brown offers good value for money?

Approach:

1. Filter for UK respondents using the `country` column.
2. Focus on the `q16o_03` column, which captures opinions about Bobbi Brown's value for money.
3. Count responses for "Strongly agree" and "Slightly agree".

Note:

1. For the columns using the scale: "Strongly agree" to "Strongly disagree" or "Don't know", the following mapping has been used:
   - "Strongly agree" = 1.0
   - "Slightly agree" = 2.0
2. And, 'UK' refers to a value of '1.0' in the 'country' column.

[4]:
```
data['q16o_03'].isnull().sum()
```

[4]:
```
1038
```

The filter which we'll apply will take care of these null values.

[5]:
```
uk_data = data.loc[data['country'] == 1.0]
filtered_data = uk_data[uk_data['q16o_03'].isin([1.0, 2.0])]
print(f"{len(filtered_data)} respondents in the UK think Bobbi Brown offers␣
 ↪good value for money.")
```

```
24 respondents in the UK think Bobbi Brown offers good value for money.
```

1.2 Subtask 1b: Diversity Across Brands

Question: Out of all respondents, how many agree that at least 3 brands embrace diversity?

Approach:

1. Identify all columns starting with q16r_, which capture diversity perceptions for various brands.
2. Count the number of "Strongly agree" and "Slightly agree" responses per respondent.
3. Filter respondents who agree with at least 3 brands.

Code and Results:

[6]:
```
# Filter relevant columns
q16r_columns = [col for col in data.columns if col.startswith("q16r")]

# Define agreement values
agreement_values = [1.0, 2.0]

# Count agreements per respondent
data['agreement_count'] = data[q16r_columns].isin(agreement_values).sum(axis=1)

# Filter respondents with at least 3 agreements
respondents_with_3_agreements = data[data['agreement_count'] >= 3]

# Output the count
print(f"{len(respondents_with_3_agreements)} respondents agree that at least 3␣
 ↪brands embrace diversity.")
```

```
334 respondents agree that at least 3 brands embrace diversity.
```

2 Task 2: Regression Analysis

Objective: Identify drivers influencing recommendations for Charlotte Tilbury.

Steps:

1. Filter independent variables: All columns starting with `q16` and ending with `01`.
2. Define dependent variable: `q15_01` (recommendation score for Charlotte Tilbury).
3. Handle missing values by dropping rows where `q15_01` is null.
4. Train a linear regression model and evaluate results.

[7]:
```
# Filter independent variables
independent_vars = [col for col in data.columns if col.startswith("q16") and␣
↪col.endswith("01")]
print("Independent Variables:", independent_vars)
```

```
Independent Variables: ['q16a_01', 'q16b_01', 'q16c_01', 'q16d_01', 'q16e_01',
'q16f_01', 'q16g_01', 'q16h_01', 'q16i_01', 'q16j_01', 'q16k_01', 'q16l_01',
'q16m_01', 'q16n_01', 'q16o_01', 'q16p_01', 'q16q_01', 'q16r_01', 'q16s_01',
'q16t_01']
```

[8]:
```
# Null value count for q15_01
print(f"Null values in q15_01: {data['q15_01'].isnull().sum()}")
print(f"% of column null: {data['q15_01'].isnull().sum()/len(data) * 100:.2f}%␣
↪")
```

```
Null values in q15_01: 543
% of column null: 44.04%
```

[9]:
```
# Filter rows where q15_01 is non-zero
data_cleaned = data.dropna(subset=['q15_01'])
print(f"Number of rows after dropping nulls in q15_01: {len(data_cleaned)}")
```

```
Number of rows after dropping nulls in q15_01: 690
```

[10]:
```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score

# Define X (independent) and y (dependent)
X = data_cleaned[independent_vars]
y = data_cleaned['q15_01']

## Split into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
↪random_state=42)

# Train the regression model
model = LinearRegression()
model.fit(X_train, y_train)
```

```
# Evaluate the model
y_pred = model.predict(X_test)
r2 = r2_score(y_test, y_pred)

print(f"R-squared: {r2}")
```

```
R-squared: 0.47302466511376406
```

2.1 Results:

- **R-squared**: Indicates that the model explains 47% of the variance in recommendations.
- **Coefficients**: Highlight the impact of each independent variable.

Code to Display Coefficients:

[11]:
```
coefficients = pd.DataFrame({
    'Variable': independent_vars,
    'Coefficient': model.coef_
})
print(coefficients.sort_values(by='Coefficient', ascending=False))
```

```
   Variable  Coefficient
16  q16q_01     0.192975
17  q16r_01     0.187795
11  q16l_01     0.130552
13  q16n_01     0.072768
3   q16d_01     0.056635
12  q16m_01    -0.015917
5   q16f_01    -0.016197
18  q16s_01    -0.059518
2   q16c_01    -0.102029
19  q16t_01    -0.162375
7   q16h_01    -0.172013
14  q16o_01    -0.186139
10  q16k_01    -0.200283
8   q16i_01    -0.215181
4   q16e_01    -0.234090
9   q16j_01    -0.258700
6   q16g_01    -0.282280
0   q16a_01    -0.388976
1   q16b_01    -0.576377
15  q16p_01    -0.605331
```

3 Conclusion

3.1 Key Insights:

1. Bobbi Brown: 24 UK respondents think it offers good value for money.

2. Diversity: 334 respondents agree that at least 3 brands embrace diversity.
3. Regression: Diversity perceptions (e.g., `q16r_01`) are key positive drivers of recommendations, while variables like `q16p_01` negatively impact recommendations.

3.2 Recommendations:

- Focus marketing efforts on diversity-related messaging to reinforce brand strengths.
- Investigate negative drivers to identify improvement opportunities.

3.3 Limitations of the Regression Analysis

While the regression analysis provides valuable insights, several limitations should be noted:

1. **Missing Data**:
   - 44% of rows were dropped due to missing values in `q15_01` (recommendation scores). This may introduce bias if the missingness is not random, potentially affecting the generalizability of the results.
2. **Model Assumptions**:
   - The regression model assumes linearity, normality of residuals, and independence of observations. These assumptions may not fully hold, which could impact the validity of the results.
3. **Multicollinearity**:
   - Several independent variables (`q16_*`) may be highly correlated, leading to multicollinearity. This can distort the reliability of individual coefficient estimates, making it harder to identify the true drivers of recommendations.
4. **Model Fit**:
   - The R-squared value of 0.47 indicates that the model explains only 47% of the variance in recommendations. This suggests other unmeasured factors, such as demographic information or external influences, may significantly impact recommendations.
5. **Omitted Variable Bias**:
   - Potentially important predictors, such as demographic features or other brand-related perceptions, were not included in the analysis. Their absence could lead to biased estimates and incomplete insights.

3.4 Implications for Decision-Making

These limitations highlight the importance of interpreting the results with caution and combining them with domain expertise. Future work could address these issues by:

- Exploring the patterns and reasons for missing data.
- Investigating additional predictors to improve model fit.
- Validating the findings with more representative datasets or alternative models.

3.5 Additional Analysis Suggestions

1. **Segmentation Analysis**:

    - Segment respondents by demographics (e.g., age, gender, region) to uncover group-specific insights about brand perceptions.
    - **Benefit**: Helps the client tailor marketing strategies to specific audience groups.
2. **Cross-Brand Comparison**:
    - Compare perceptions and recommendations across brands.
    - **Benefit**: Helps identify competitive advantages and areas for improvement relative to competitors.