Start coding or generate with AI.

Dataset uploading

```
# prompt: DATASET UPLOAD CODE

from google.colab import files
uploaded = files.upload()
# prompt: handle missing values for an csv file in googl

import pandas as pd

# Load the CSV file into a pandas DataFrame
df = pd.read_csv('customer.csv')
```

Choose Files customer.csv

- **customer.csv**(text/csv) - 3055 bytes, last modified: 5/8/2025 - 100% done

Saving customer.csv to customer.csv

Handling Missing Values

```
# prompt: check for missing values and fill the missing

# Check for missing values
print(df.isnull().sum())

# Fill missing values with mean for numerical columns
numerical_cols = df.select_dtypes(include=['number']).co
df[numerical_cols] = df[numerical_cols].fillna(df[numeri

# Fill missing values with mode for categorical columns
categorical_cols = df.select_dtypes(include=['object']).
df[categorical_cols] = df[categorical_cols].fillna(df[ca

# Verify if missing values are filled
print(df.isnull().sum())
```

```
Year                        0
Date                        0
Question_Number             0
Question                    0
Number_of_Respondents       0
Very_Satisfied              0
Satisfied                   0
Neutral                     0
Dissatisfied                0
```

ValueError NameError •••

Please explain the error:

```
ValueError: Could not interpr
```

Hypothesis

The error message "ValueError: Could not interpret value `Geography` for `x`. Value is a string, but `data` was not passed." arises because the `sns.boxplot` function is called without specifying the `data` parameter, and you're trying to use column names ('Geography', 'CreditScore') directly. When you don't provide the `data` argument, seaborn expects the `x` and `y` arguments to be arrays or lists of data, not column names.

In this case, you're likely intending to use the `original_df` DataFrame (as mentioned in your code comments) for the visualization, but you haven't passed it to the `sns.boxplot` function. As a result, seaborn doesn't know where to find the 'Geography' and 'CreditScore' columns.

Suggested Changes

```
# Example 3: Boxplot for a sp
# (using the original_df to a
plt.figure(figsize=(10, 6))
# Pass the original_df DataFr
sns.boxplot(x='Geography', y=
```

```
Very_Dissatisfied              0
Very_Satisfied_or_Satisfied    0
ObjectId                       0
dtype: int64
Year                           0
Date                           0
Question_Number                0
Question                       0
Number_of_Respondents          0
Very_Satisfied                 0
Satisfied                      0
Neutral                        0
Dissatisfied                   0
Very_Dissatisfied              0
Very_Satisfied_or_Satisfied    0
ObjectId                       0
dtype: int64
```

Duplicate records

```
# prompt: check for the duplicate records and remove the

# Check for duplicate rows
duplicate_rows = df[df.duplicated()]

# Print the duplicate rows
print("Duplicate Rows:")
print(duplicate_rows)

# Remove duplicate rows
df = df.drop_duplicates()

# Print the DataFrame after removing duplicates
print("\nDataFrame after removing duplicates:")
df
```

```
plt.title("CreditScore Distri
plt.xticks(rotation=45, ha='r
plt.show()

# Example 4: Countplot for a
plt.figure(figsize=(8, 6))
sns.countplot(x='Gender', dat
plt.title("Count of Gender")
plt.show()

# Example 5: Distribution of
plt.figure(figsize=(10, 6))
sns.countplot(x='Geography',
plt.title('Distribution of Ex
plt.xticks(rotation=45, ha='r
plt.show()

# Example 6: Histogram of a N
plt.figure(figsize=(10, 6))
sns.histplot(data=original_df
plt.title('Distribution of Ag
plt.show()
```

Use code with caution

Rate this answer

```
Duplicate Rows:
Empty DataFrame
Columns: [Year, Date, Question_Number, Question, Num
Index: []

DataFrame after removing duplicates:
```

	Year	Date	Question_Number	Question	Nu
0	2017	2017/10/31 07:00:00+00	7-13	Overall quality of customer service	
1	2016	2016/10/31 07:00:00+00	26	Overall quality of customer service	
2	2015	2015/10/31 07:00:00+00	Survey Not Conducted	Survey Not Conducted	
3	2014	2014/10/31 07:00:00+00	10b	How easy was the City to contact	
4	2014	2014/10/31 07:00:00+00	10c	The way you were treated	
5	2014	2014/10/31 07:00:00+00	10d	The accuracy of the information you were given	
6	2014	2014/10/31 07:00:00+00	10e	How quickly staff responded to your request	
7	2014	2014/10/31 07:00:00+00	10f	How well your issue was handled	
8	2013	2013/10/31 07:00:00+00	14b	How easy was the City to contact	
9	2013	2013/10/31 07:00:00+00	14c	The way you were treated	

10	2013	2013/10/31 07:00:00+00	14d	The accuracy of the information you were given
11	2013	2013/10/31 07:00:00+00	14e	How quickly staff responded to your request
12	2013	2013/10/31 07:00:00+00	14f	How well your issue was handled
13	2012	2012/10/31 07:00:00+00	14b	How easy was the city to contact
14	2012	2012/10/31 07:00:00+00	14c	The way you were treated
15	2012	2012/10/31 07:00:00+00	14d	The accuracy of the information you were given
16	2012	2012/10/31 07:00:00+00	14e	How quickly responded to request
17	2012	2012/10/31 07:00:00+00	14f	How well issue was handled
18	2011	2011/10/31 07:00:00+00	14b	How easy was the contact
19	2011	2011/10/31 07:00:00+00	14c	The way you were treated
20	2011	2011/10/31 07:00:00+00	14d	Accuracy of information you were given

21	2011	2011/10/31 07:00:00+00	14e	How quickly staff responded to your request
22	2011	2011/10/31 07:00:00+00	14f	How well your issue was handled
23	2010	2010/10/31 07:00:00+00	14B	How easy was the city to contact
24	2010	2010/10/31 07:00:00+00	14C	The way you were treated
25	2010	2010/10/31 07:00:00+00	14D	Accuracy of information you were given
26	2010	2010/10/31 07:00:00+00	14E	How quickly staff responded to your request
27	2010	2010/10/31 07:00:00+00	14F	How well your issue was handled

Next steps: Generate code with `df` View recommended plots

Outliers

```
# prompt: check for the outliers in the above dataset

import pandas as pd
import numpy as np

# Assuming 'df' is your DataFrame with numerical feature

def find_outliers_iqr(data):
    Q1 = np.percentile(data, 25)
    Q3 = np.percentile(data, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = data[(data < lower_bound) | (data > upper
    return outliers


numerical_features = df.select_dtypes(include=np.number)
for col in numerical_features:
    outliers = find_outliers_iqr(df[col])
    print(f"Outliers in {col}:")
    print(outliers)
    print("-" * 20)
```

```
Outliers in Year:
Series([], Name: Year, dtype: int64)
--------------------
Outliers in Number_of_Respondents:
0      882
1     1202
2    99999
Name: Number_of_Respondents, dtype: int64
--------------------
Outliers in Very_Satisfied:
0       22.16
1       22.39
2    99999.00
Name: Very_Satisfied, dtype: float64
--------------------
Outliers in Satisfied:
2    99999.0
Name: Satisfied, dtype: float64
--------------------
Outliers in Neutral:
0       25.03
1       23.37
2    99999.00
Name: Neutral, dtype: float64
```

```
--------------------
Outliers in Dissatisfied:
2    99999.0
Name: Dissatisfied, dtype: float64
--------------------
Outliers in Very_Dissatisfied:
2    99999.0
Name: Very_Dissatisfied, dtype: float64
--------------------
Outliers in Very_Satisfied_or_Satisfied:
2    99999.0
Name: Very_Satisfied_or_Satisfied, dtype: float64
--------------------
Outliers in ObjectId:
Series([], Name: ObjectId, dtype: int64)
--------------------
```

Standardization

```
# prompt: standardize the above dataset

from sklearn.preprocessing import StandardScaler

# Assuming 'df' is your DataFrame with numerical feature

# Create a StandardScaler object
scaler = StandardScaler()

# Select numerical columns for standardization
numerical_cols = df.select_dtypes(include=np.number).col

# Fit and transform the numerical columns
df[numerical_cols] = scaler.fit_transform(df[numerical_c

# Print the standardized DataFrame
print("\nStandardized DataFrame:")
df
```

Standardized DataFrame:

	Year	Date	Question_Number	Questio
0	2.483682	2017/10/31 07:00:00+00	7-13	Overa quality c custome servic
1	1.940376	2016/10/31 07:00:00+00	26	Overa quality c custome servic
2	1.397071	2015/10/31 07:00:00+00	Survey Not Conducted	Survey Nc Conducte
3	0.853766	2014/10/31 07:00:00+00	10b	How eas was th City t contac
4	0.853766	2014/10/31 07:00:00+00	10c	The wa you wer treate
5	0.853766	2014/10/31 07:00:00+00	10d	Th accurac of th informatio you wer give
6	0.853766	2014/10/31 07:00:00+00	10e	Hov quickl sta respond to you reques
7	0.853766	2014/10/31 07:00:00+00	10f	How we your issu wa handle
8	0.310460	2013/10/31 07:00:00+00	14b	How eas was th City t contac
9	0.310460	2013/10/31 07:00:00+00	14c	The wa you wer treate
10	0.310460	2013/10/31 07:00:00+00	14d	Th accurac of th informatio

				you were given
11	0.310460	2013/10/31 07:00:00+00	14e	How quickly staff responded to your request
12	0.310460	2013/10/31 07:00:00+00	14f	How well your issue was handled
13	-0.232845	2012/10/31 07:00:00+00	14b	How easy was the city to contact
14	-0.232845	2012/10/31 07:00:00+00	14c	The way you were treated
15	-0.232845	2012/10/31 07:00:00+00	14d	The accuracy of the information you were given
16	-0.232845	2012/10/31 07:00:00+00	14e	How quickly responded to request
17	-0.232845	2012/10/31 07:00:00+00	14f	How well issue was handled
18	-0.776151	2011/10/31 07:00:00+00	14b	How easy was the contact
19	-0.776151	2011/10/31 07:00:00+00	14c	The way you were treated
20	-0.776151	2011/10/31 07:00:00+00	14d	Accuracy information you were given
21	-0.776151	2011/10/31 07:00:00+00	14e	How quickly staff responded to your

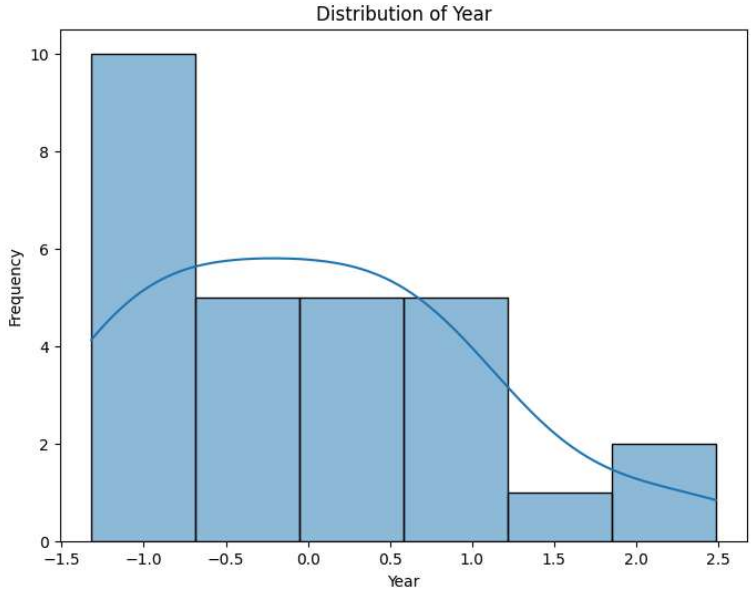
				to you reques
22	-0.776151	2011/10/31 07:00:00+00	14f	How we your issu wa handle
23	-1.319456	2010/10/31 07:00:00+00	14B	How eas was th city t contac
24	-1.319456	2010/10/31 07:00:00+00	14C	The wa you wer treate
25	-1.319456	2010/10/31 07:00:00+00	14D	Accurac c informatio you wer give
26	-1.319456	2010/10/31 07:00:00+00	14E	Hov quickl sta responde to you reques
27	-1.319456	2010/10/31 07:00:00+00	14F	How we your issu wa handle

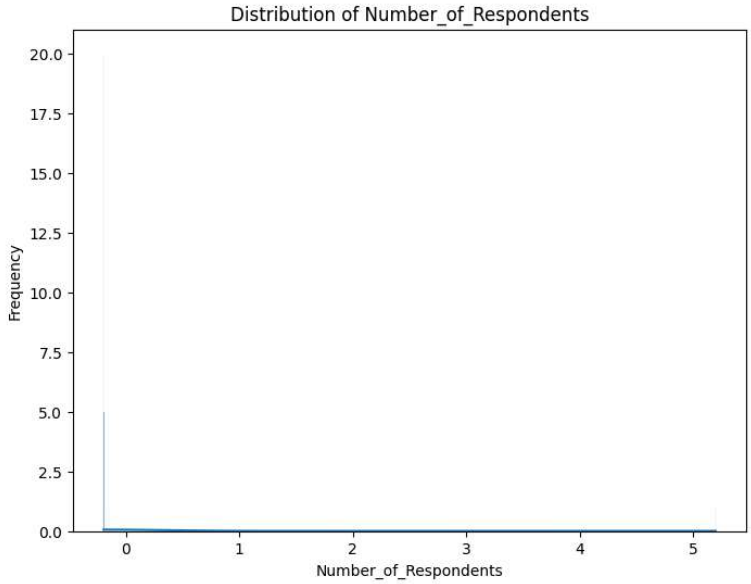
Next steps: Generate code with df View recommended plots

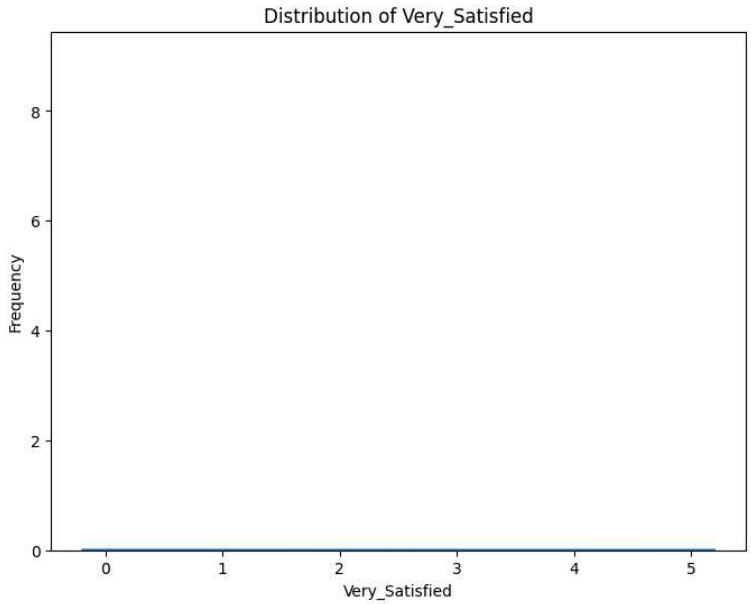
EDA

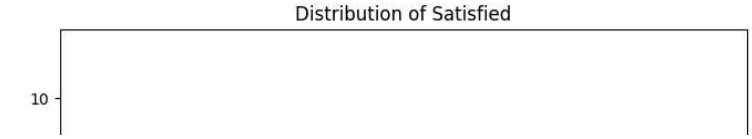
```
# prompt: visualize the dataset using eda by univariate,

import matplotlib.pyplot as plt
import seaborn as sns

# Univariate Analysis

# Histograms for numerical features
for col in numerical_features:
    plt.figure(figsize=(8, 6))
    sns.histplot(df[col], kde=True)
    plt.title(f'Distribution of {col}')
    plt.xlabel(col)
    plt.ylabel('Frequency')
    plt.show()

# Box plots for numerical features
for col in numerical_features:
    plt.figure(figsize=(8, 6))
    sns.boxplot(df[col])
    plt.title(f'Box Plot of {col}')
    plt.ylabel(col)
    plt.show()

# Count plots for categorical features
for col in categorical_cols:
    plt.figure(figsize=(8, 6))
    sns.countplot(x=col, data=df)
    plt.title(f'Count Plot of {col}')
    plt.xlabel(col)
    plt.ylabel('Count')
    plt.xticks(rotation=45, ha='right') # Rotate x-axis
    plt.show()


# Bivariate Analysis

# Scatter plots for numerical features
for col1 in numerical_features:
    for col2 in numerical_features:
        if col1 != col2:
            plt.figure(figsize=(8, 6))
            sns.scatterplot(x=col1, y=col2, data=df)
            plt.title(f'Scatter Plot of {col1} vs {col2}
            plt.xlabel(col1)
            plt.ylabel(col2)
            plt.show()
```

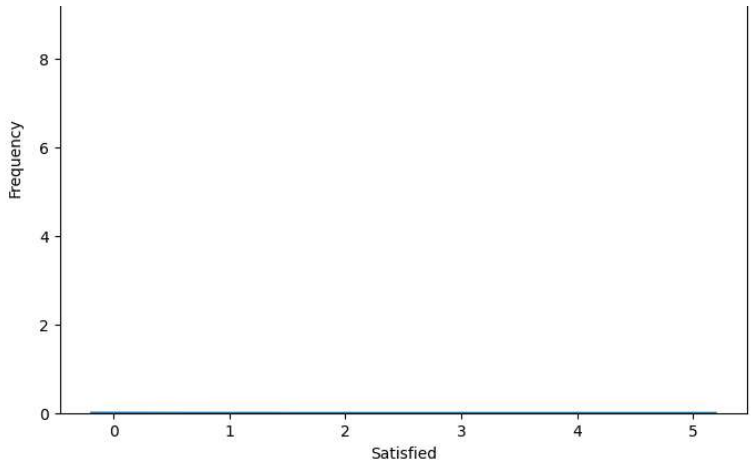
```
# Correlation Heatmap
plt.figure(figsize=(12, 10))
sns.heatmap(df[numerical_cols].corr(), annot=True, cmap=
plt.title('Correlation Heatmap of Numerical Features')
plt.show()

# Box plots for numerical features grouped by a categori
for col in numerical_features:
  for cat_col in categorical_cols:
    plt.figure(figsize=(10,6))
    sns.boxplot(x = cat_col, y = col, data=df)
    plt.title(f"Box plot of {col} grouped by {cat_col}")
    plt.show()
```

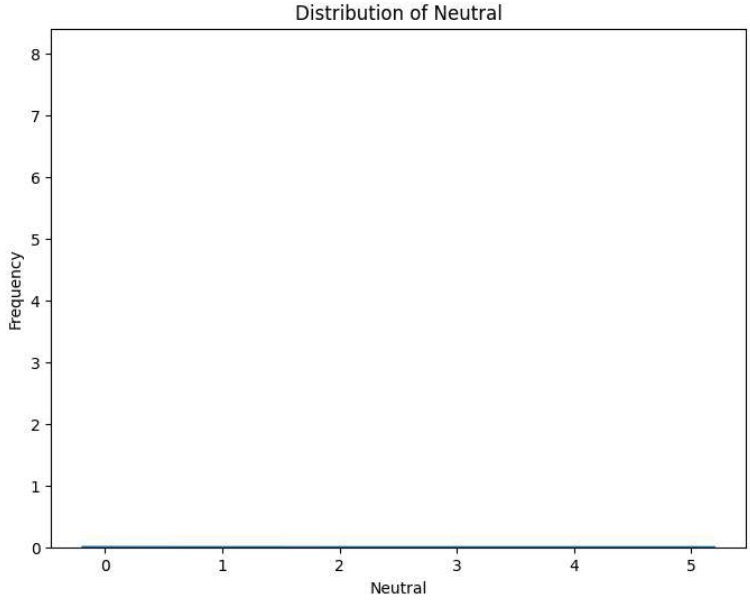

Distribution of Year
Frequency
Year
0
2
4
6
8
10
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5

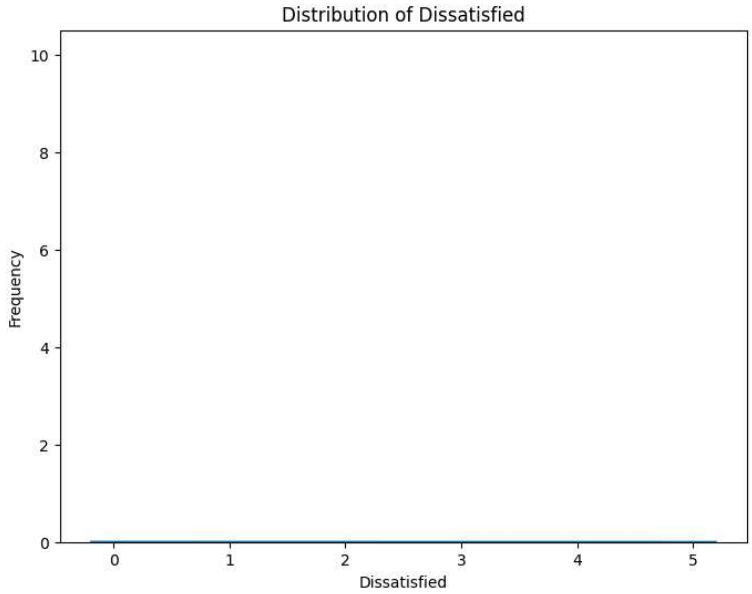

Distribution of Number_of_Respondents
Frequency
Number_of_Respondents
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0
1
2
3
4
5

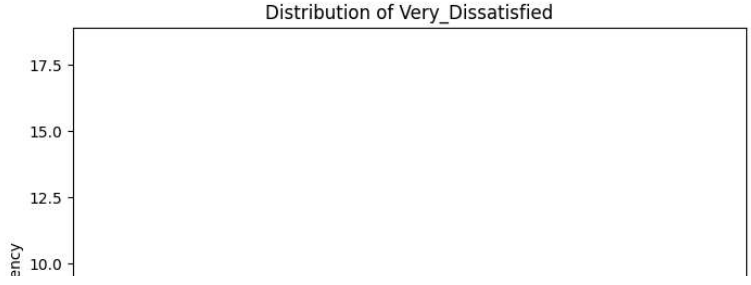

Distribution of Very_Satisfied
Frequency
Very_Satisfied
0
2
4
6
8
0
1
2
3
4
5

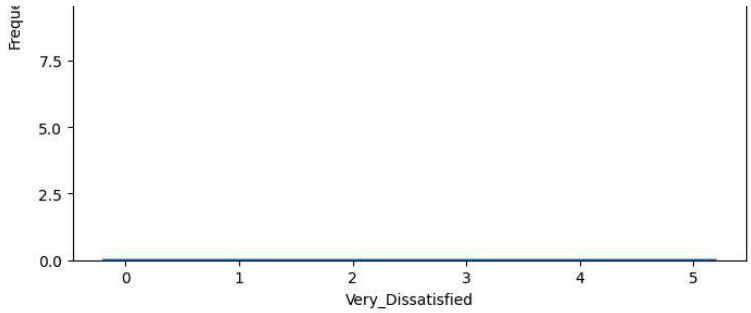

Distribution of Satisfied
10

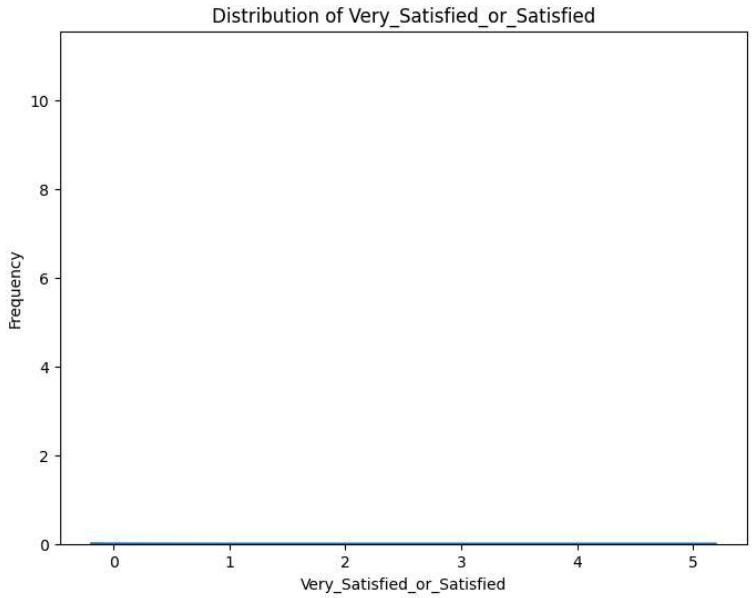

Frequency
Satisfied

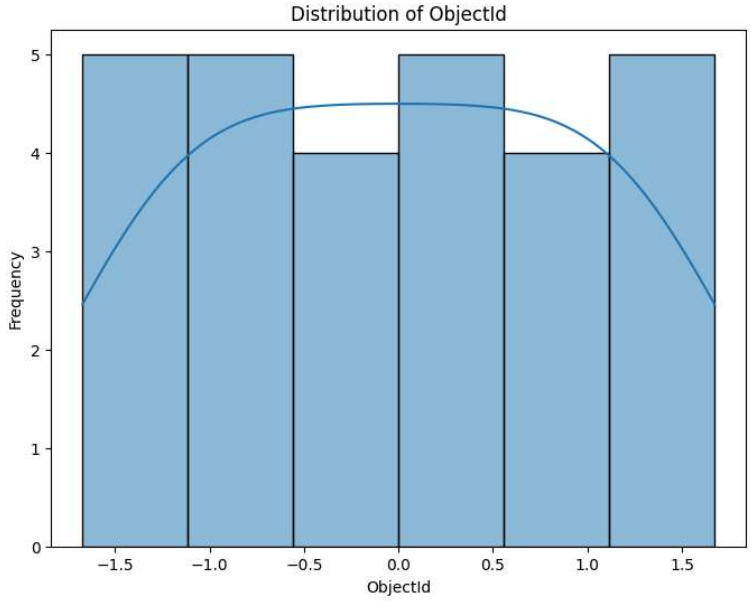

Distribution of Neutral
Frequency
Neutral

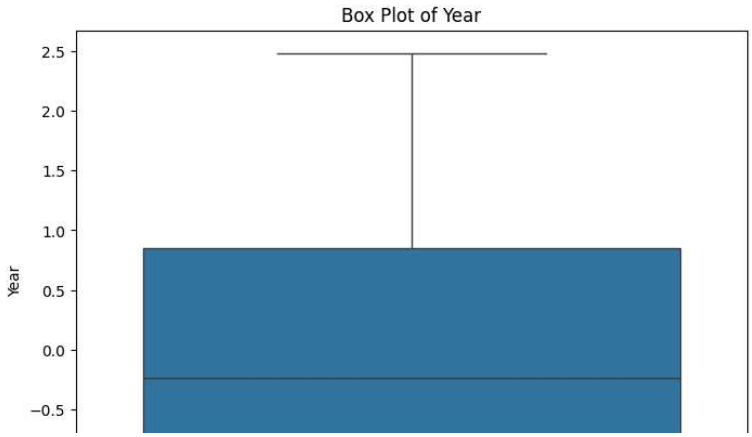

Distribution of Dissatisfied
Frequency
Dissatisfied

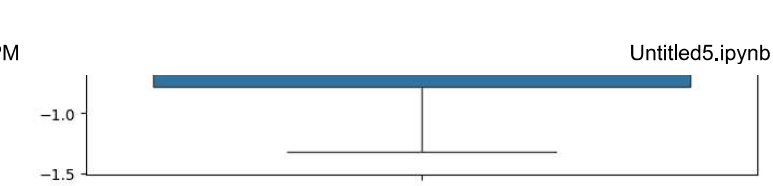

Distribution of Very_Dissatisfied

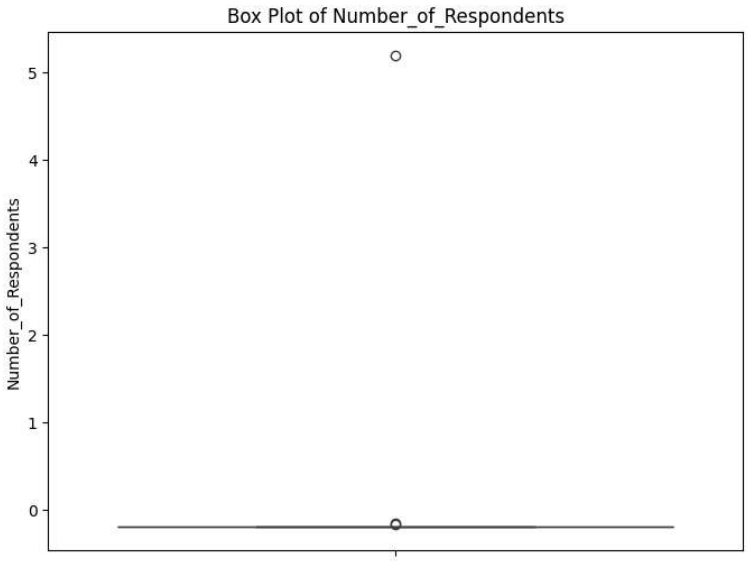

Frequ
7.5
5.0
2.5
0.0
0
1
2
3
4
5
Very_Dissatisfied

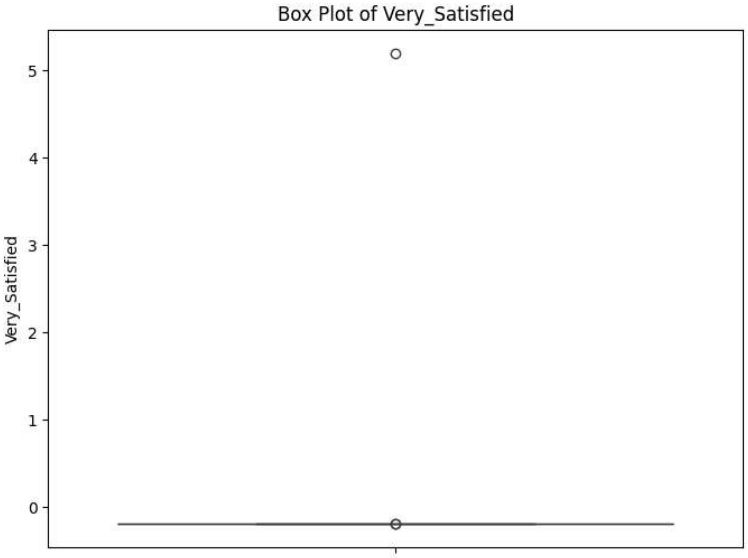

Distribution of Very_Satisfied_or_Satisfied
Frequency
10
8
6
4
2
0
0
1
2
3
4
5
Very_Satisfied_or_Satisfied

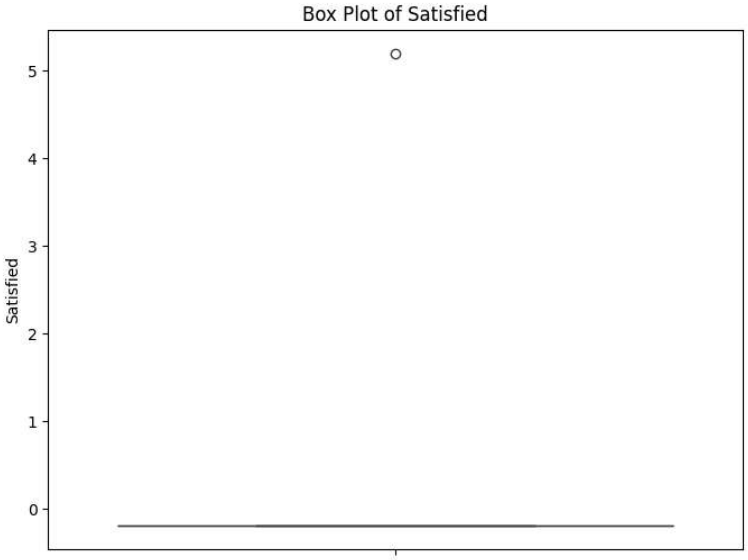

Distribution of ObjectId
Frequency
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

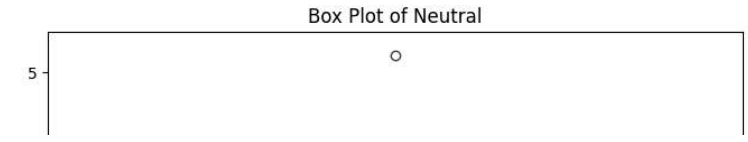

Box Plot of Year
Year
2.5
2.0
1.5
1.0
0.5
0.0
−0.5

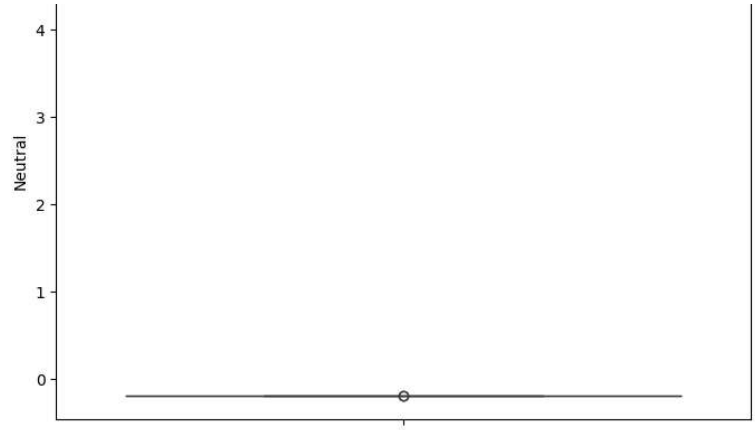

−1.0
−1.5

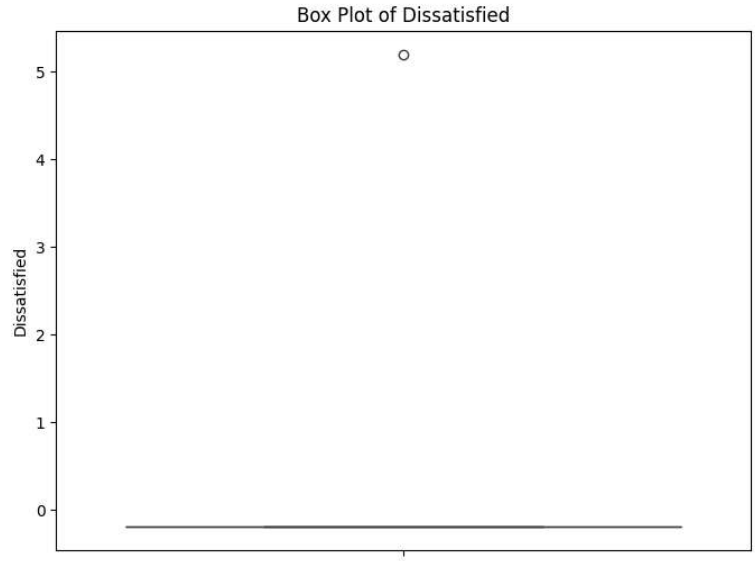

Box Plot of Number_of_Respondents
Number_of_Respondents
5
4
3
2
1
0

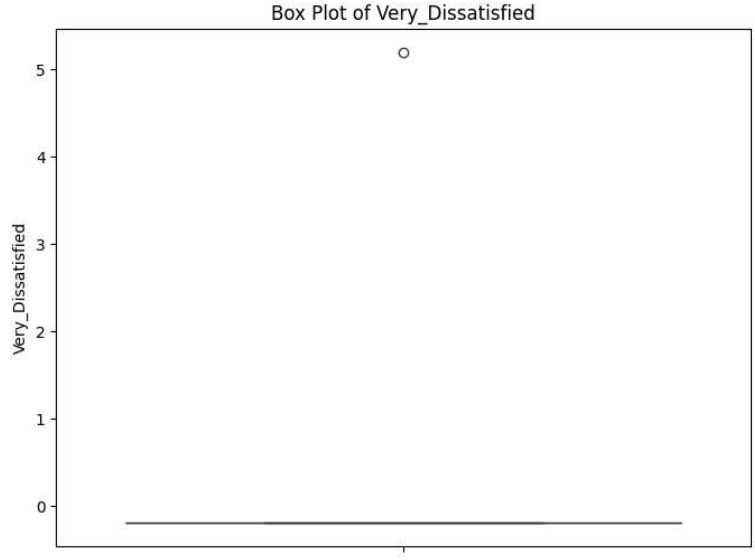

Box Plot of Very_Satisfied
Very_Satisfied
5
4
3
2
1
0

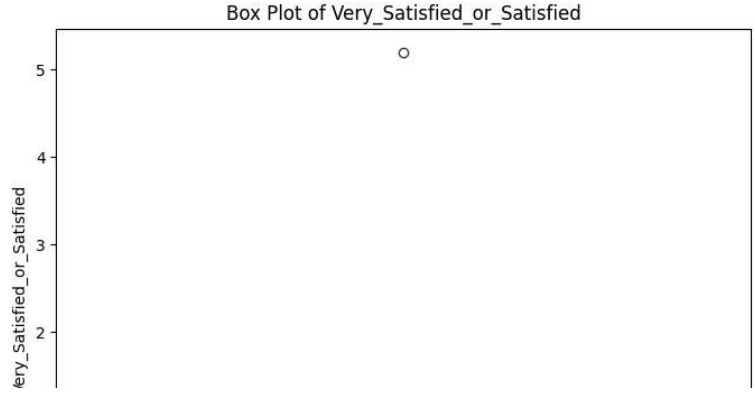

Box Plot of Satisfied
Satisfied
5
4
3
2
1
0


Box Plot of Neutral
5

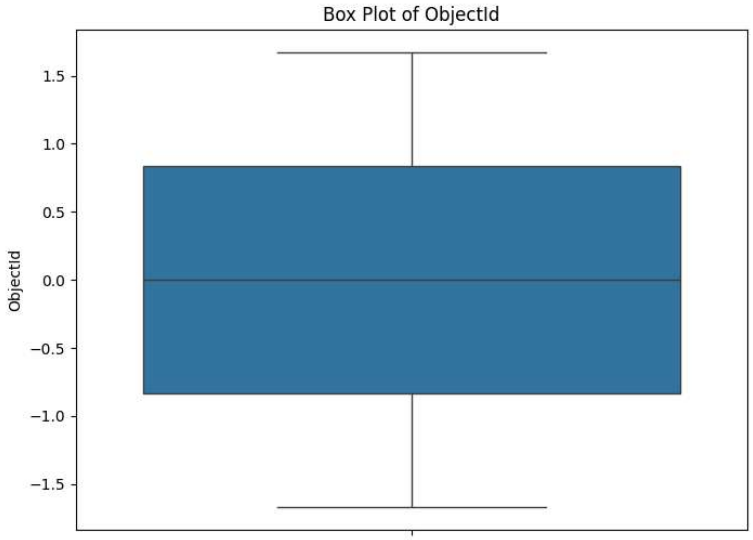

Neutral
4
3
2
1
0

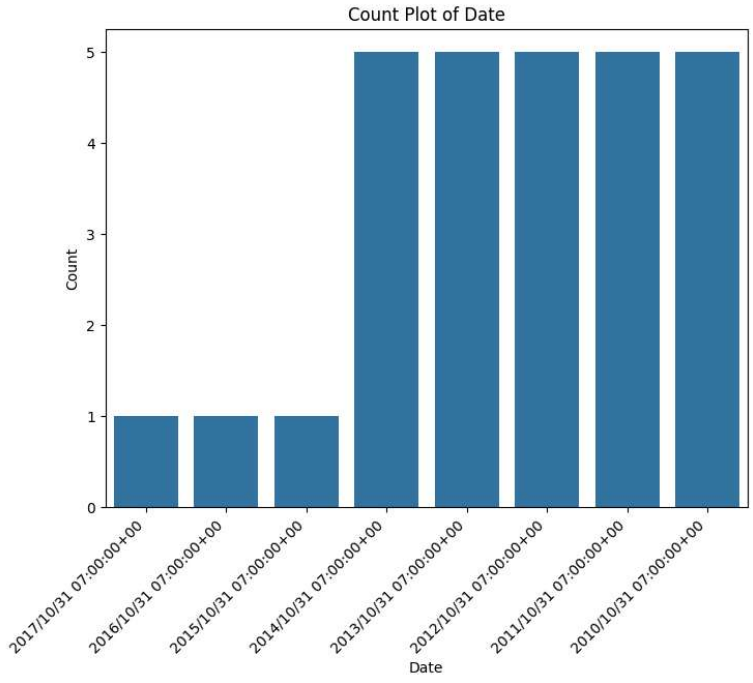

Box Plot of Dissatisfied
Dissatisfied
5
4
3
2
1
0

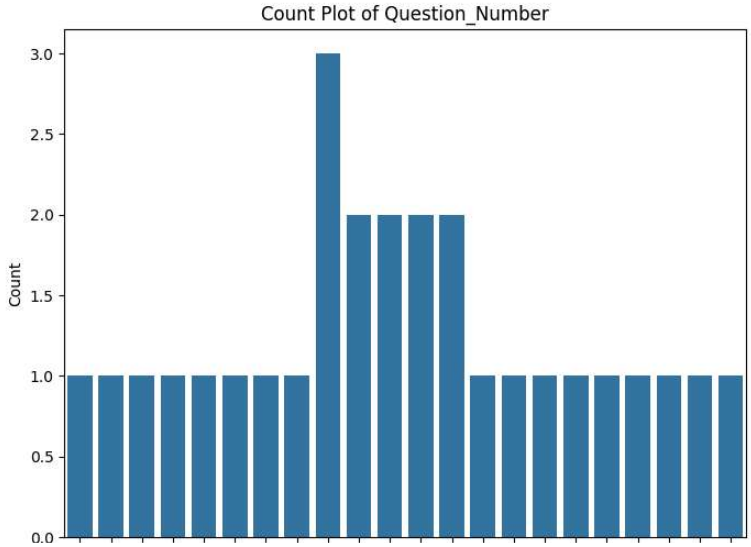

Box Plot of Very_Dissatisfied
Very_Dissatisfied
5
4
3
2
1
0

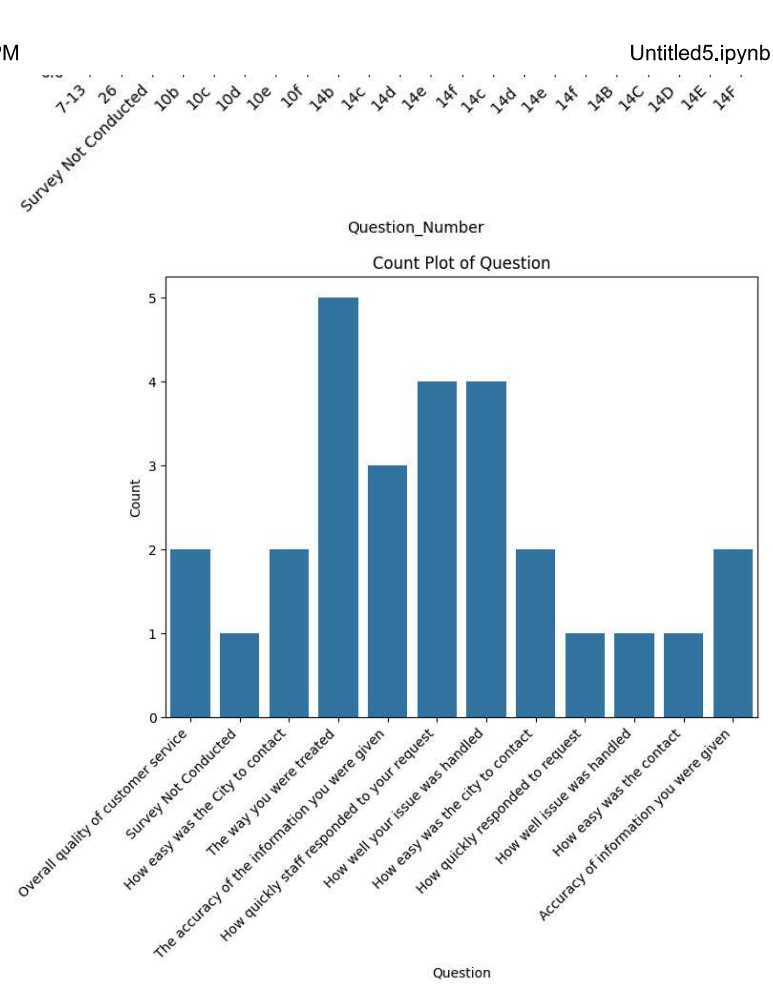

Box Plot of Very_Satisfied_or_Satisfied
ery_Satisfied_or_Satisfied
5
4
3
2

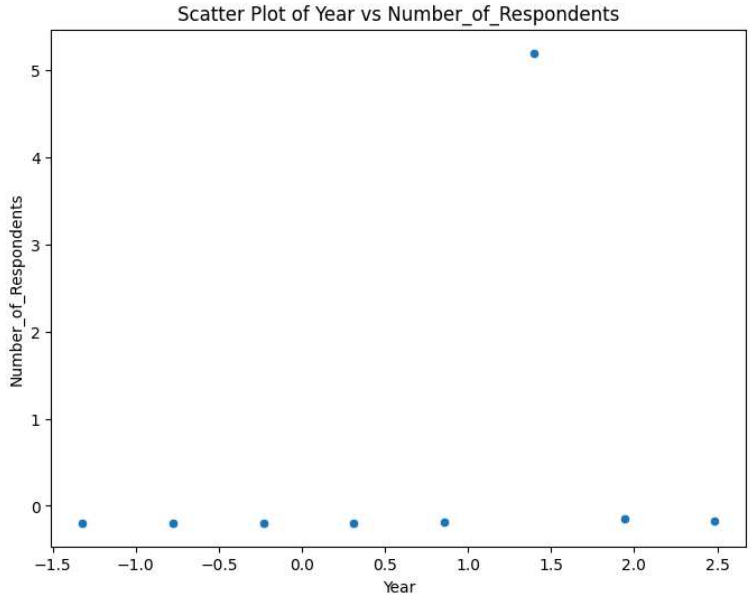

1
0

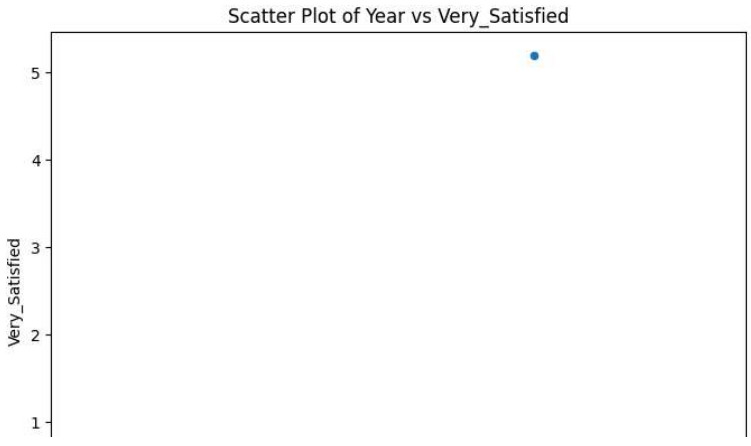

Box Plot of ObjectId
ObjectId
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5

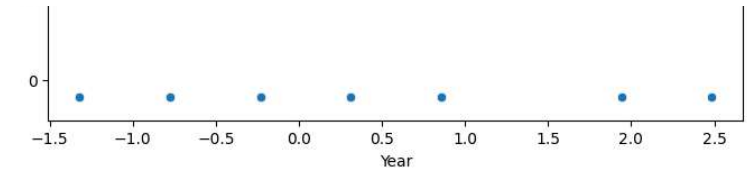

Count Plot of Date
Count
Date
5
4
3
2
1
0
2017/10/31 07:00:00+00
2016/10/31 07:00:00+00
2015/10/31 07:00:00+00
2014/10/31 07:00:00+00
2013/10/31 07:00:00+00
2012/10/31 07:00:00+00
2011/10/31 07:00:00+00
2010/10/31 07:00:00+00

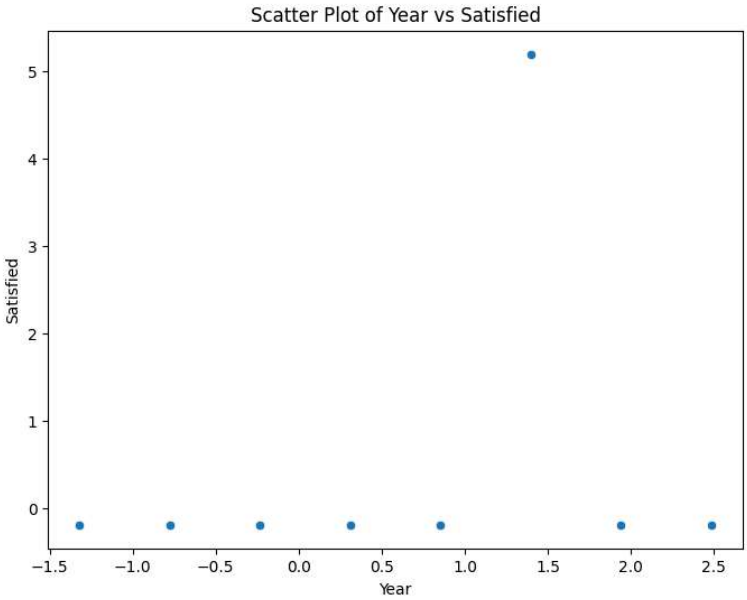

Count Plot of Question_Number
Count
3.0
2.5
2.0
1.5
1.0
0.5
0.0

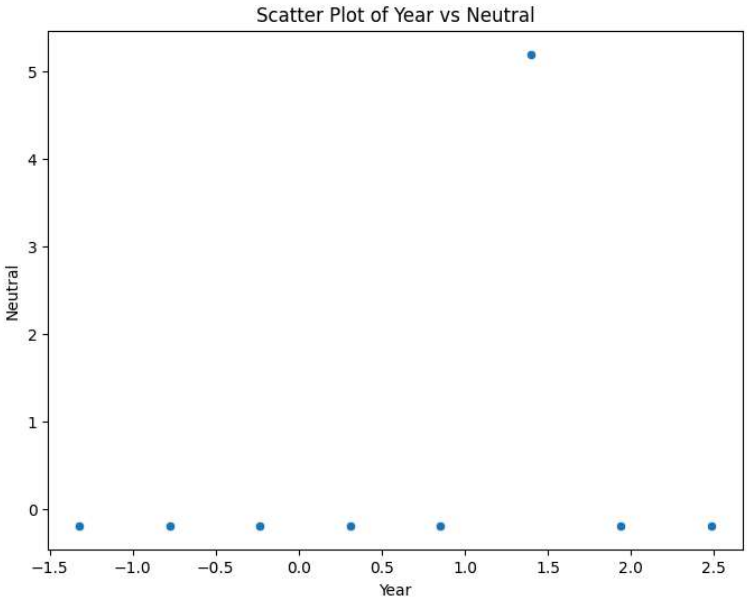

7-13
26
10b
10c
10d
10e
10f
14b
14c
14d
14e
14f
14c
14d
14e
14f
14B
14C
14D
14E
14F
Survey Not Conducted
Question_Number
Count Plot of Question
Count
Overall quality of customer service
Survey Not Conducted
How easy was the City to contact
The way you were treated
The accuracy of the information you were given
How quickly staff responded to your request
How well your issue was handled
How easy was the city to contact
How quickly responded to request
How well issue was handled
How easy was the contact
Accuracy of information you were given
Question

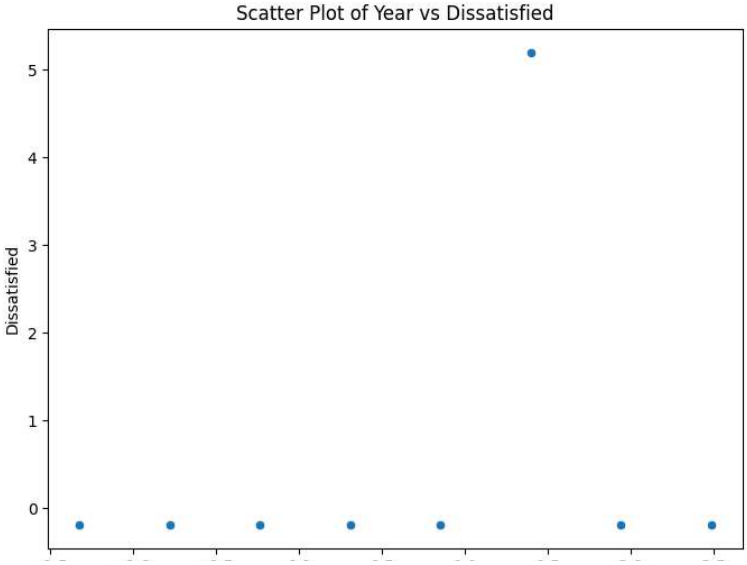

Scatter Plot of Year vs Number_of_Respondents
Number_of_Respondents
Year

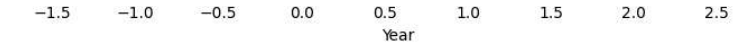

Scatter Plot of Year vs Very_Satisfied
Very_Satisfied

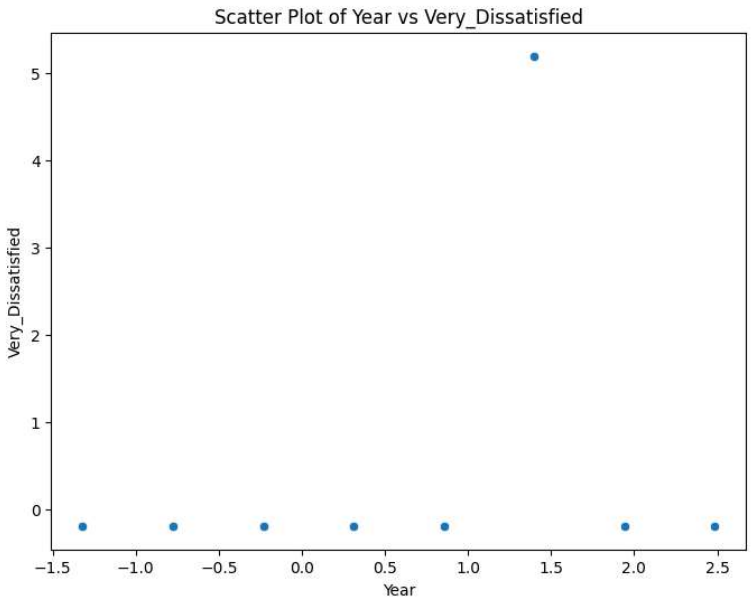

0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
Year

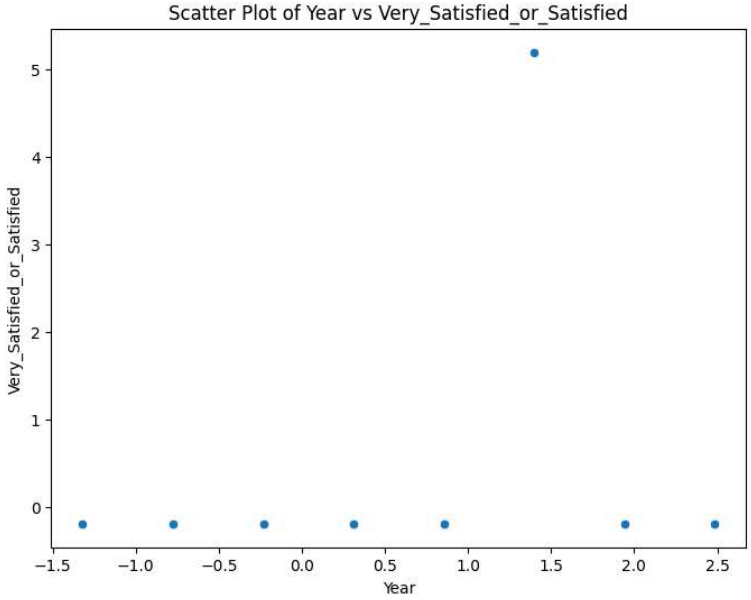

Scatter Plot of Year vs Satisfied
Satisfied
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
Year

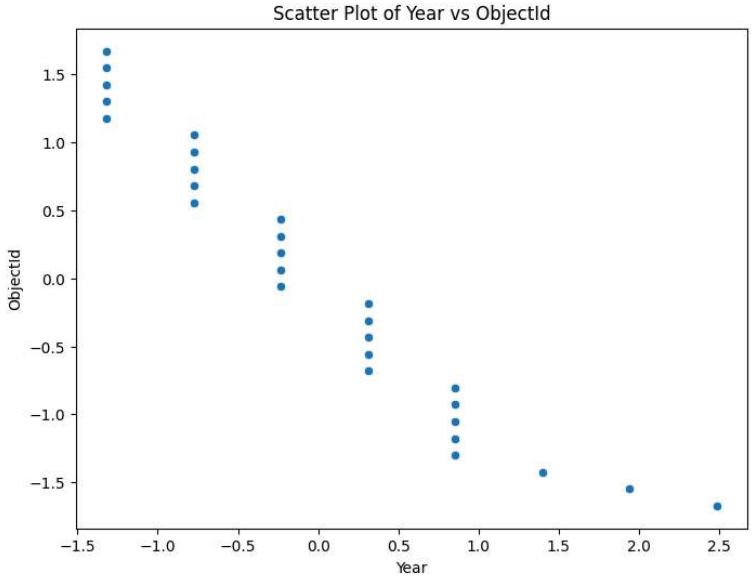

Scatter Plot of Year vs Neutral
Neutral
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
Year

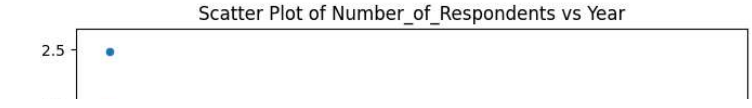

Scatter Plot of Year vs Dissatisfied
Dissatisfied
5
4
3
2
1
0

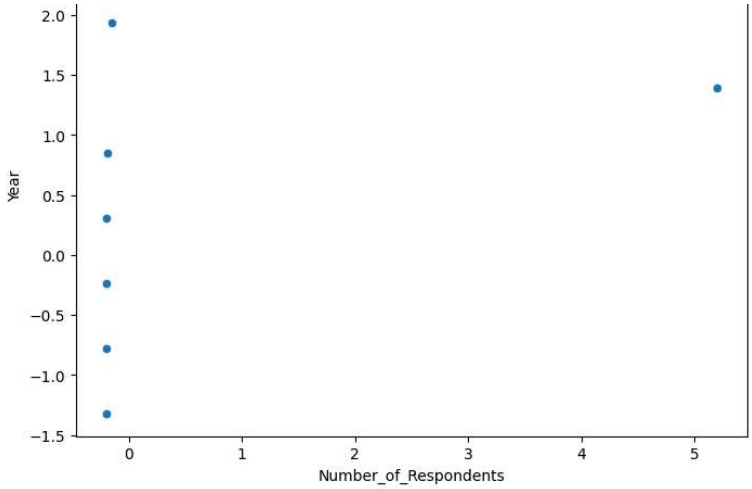

−1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0 2.5
Year

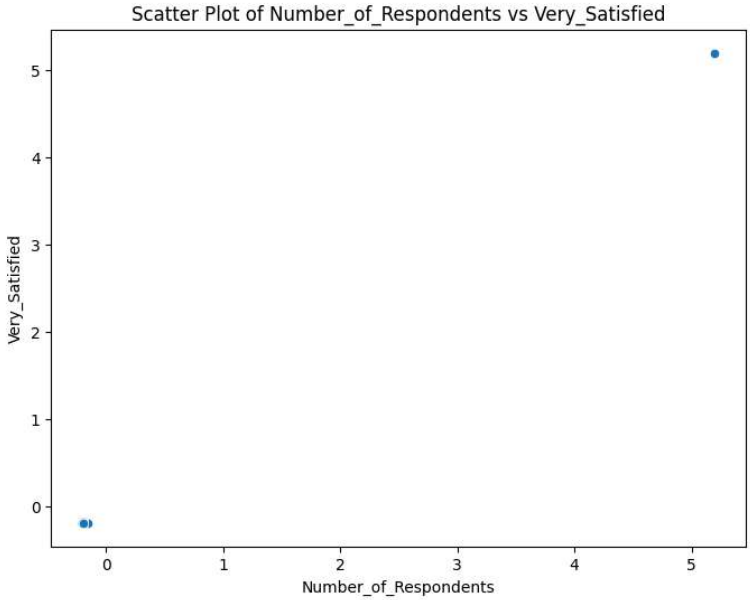

Scatter Plot of Year vs Very_Dissatisfied
Very_Dissatisfied
Year
5
4
3
2
1
0
−1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0 2.5

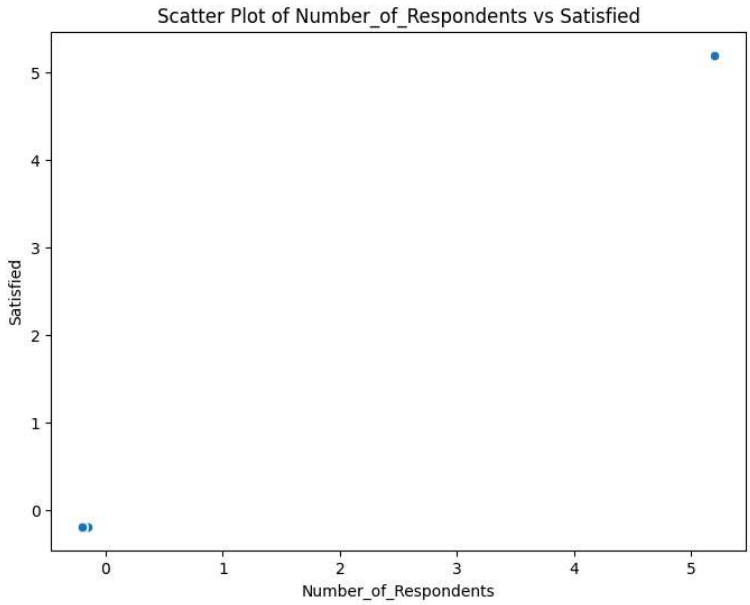

Scatter Plot of Year vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
Year
5
4
3
2
1
0
−1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0 2.5

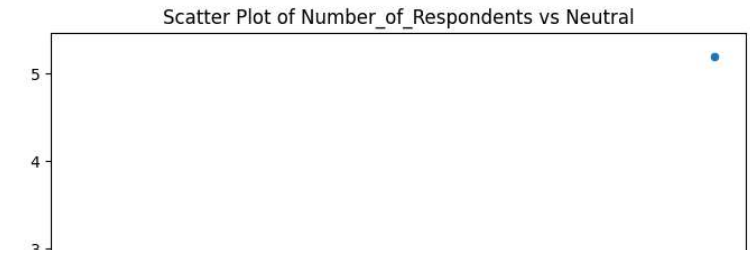

Scatter Plot of Year vs ObjectId
ObjectId
Year
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0 2.5

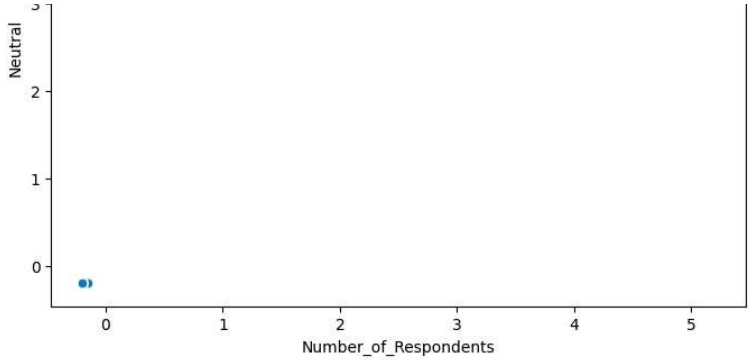

Scatter Plot of Number_of_Respondents vs Year
2.5

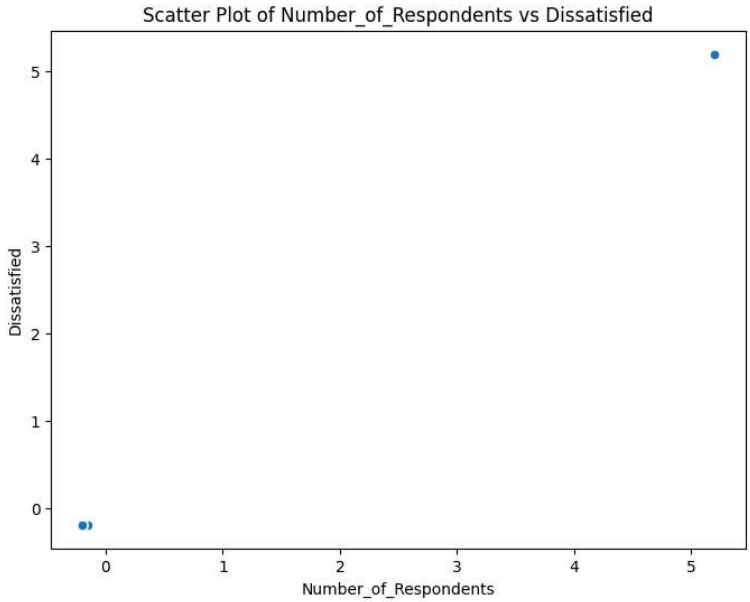

Year
Number_of_Respondents

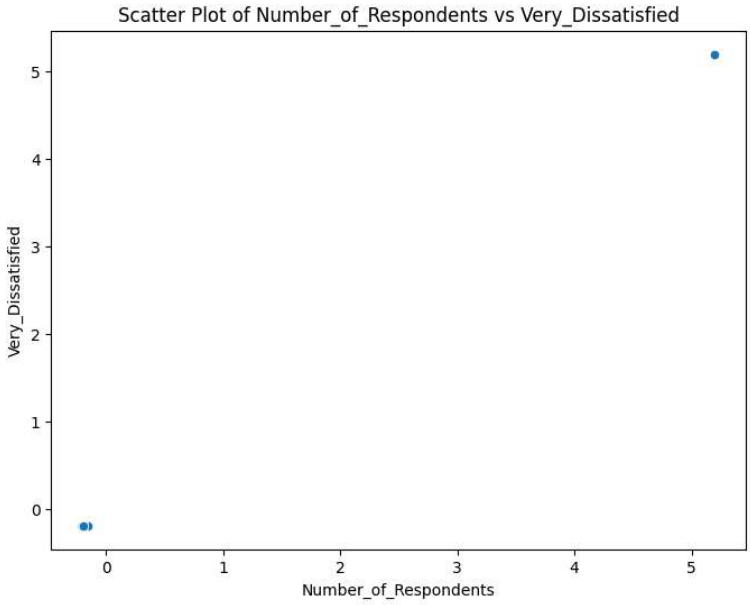

Scatter Plot of Number_of_Respondents vs Very_Satisfied
Very_Satisfied
Number_of_Respondents

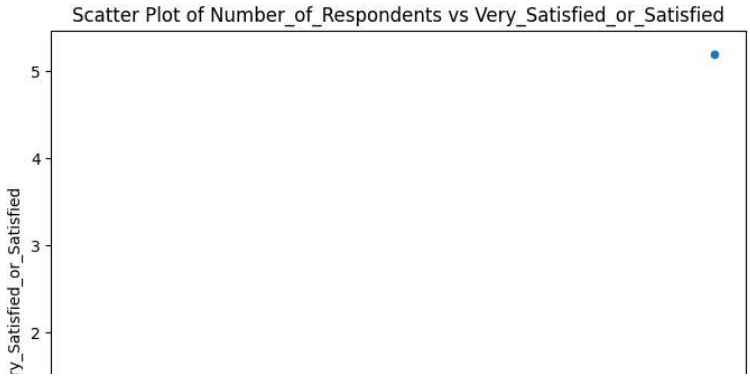

Scatter Plot of Number_of_Respondents vs Satisfied
Satisfied
Number_of_Respondents

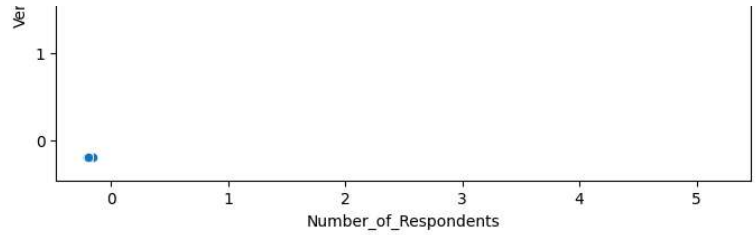

Scatter Plot of Number_of_Respondents vs Neutral

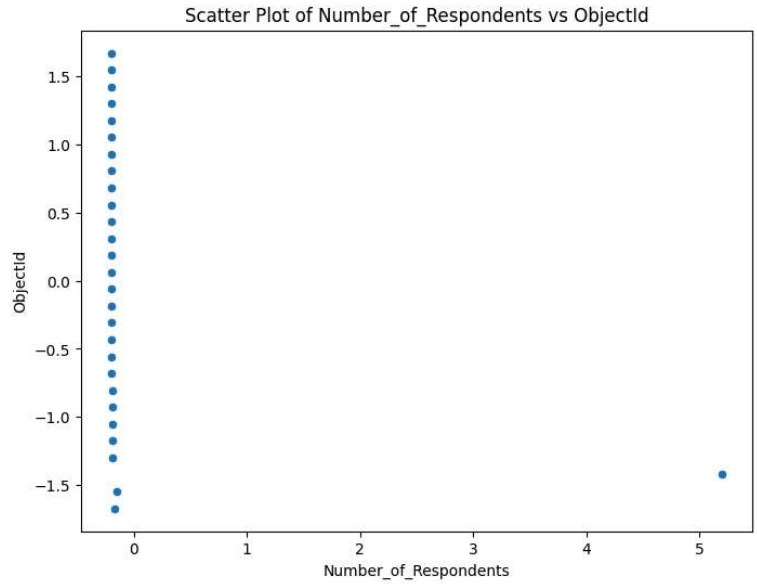

Neutral
Number_of_Respondents

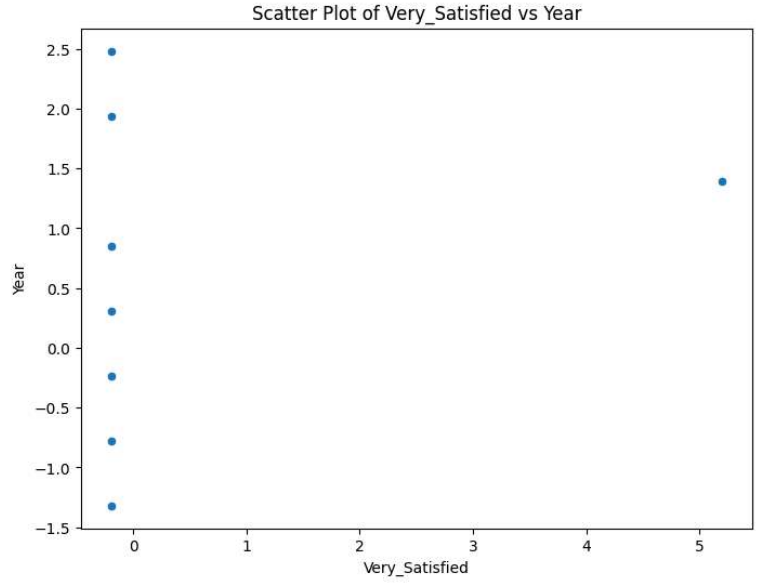

Scatter Plot of Number_of_Respondents vs Dissatisfied
Dissatisfied
Number_of_Respondents

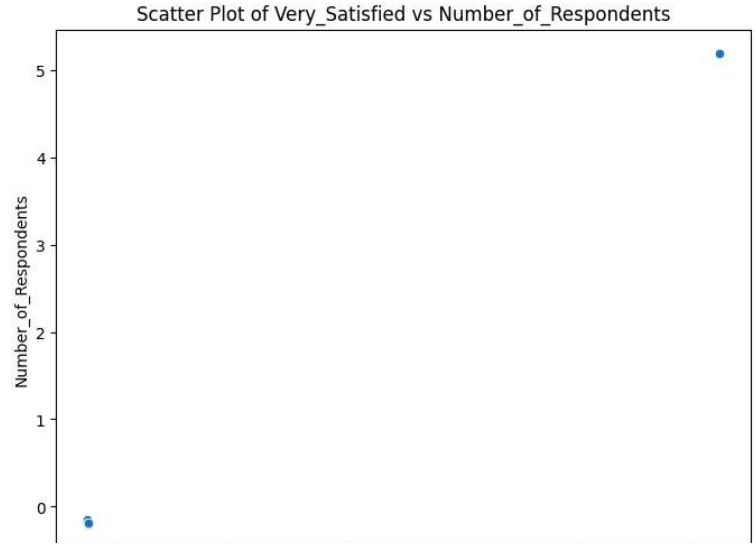

Scatter Plot of Number_of_Respondents vs Very_Dissatisfied
Very_Dissatisfied
Number_of_Respondents


Scatter Plot of Number_of_Respondents vs Very_Satisfied_or_Satisfied

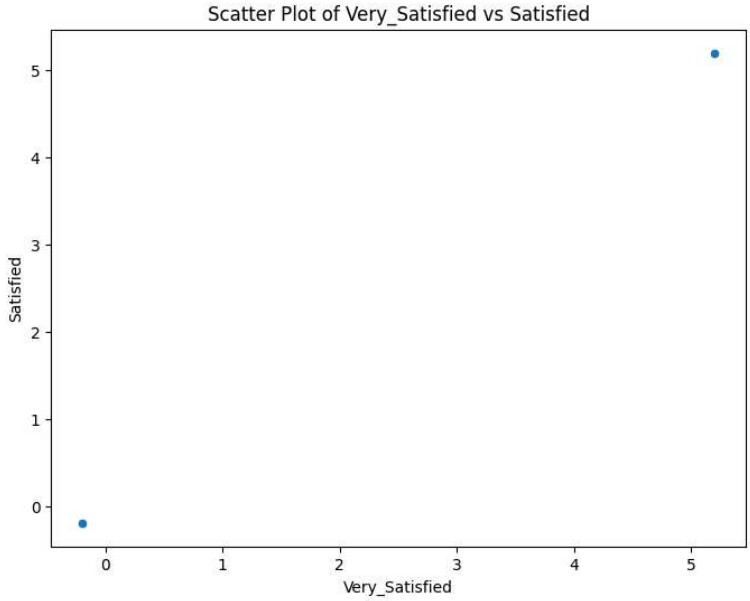

Ver
1
0
0
1
2
3
4
5
Number_of_Respondents

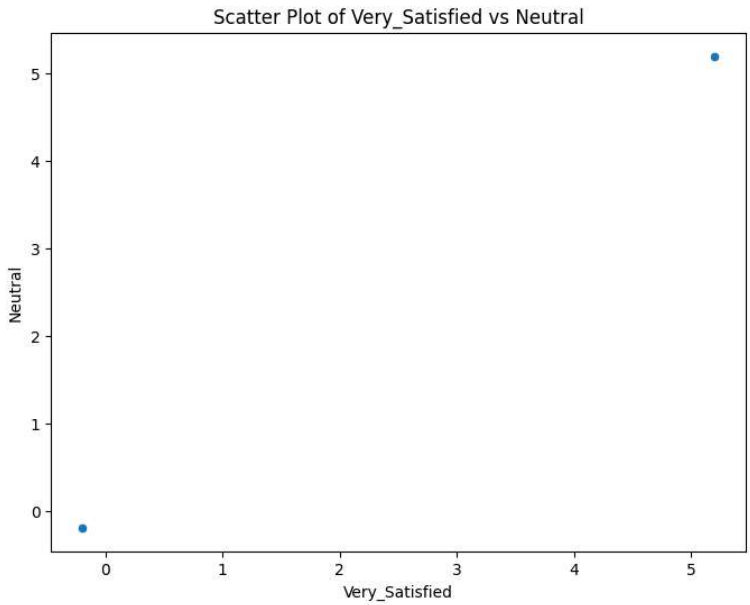

Scatter Plot of Number_of_Respondents vs ObjectId
ObjectId
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
Number_of_Respondents

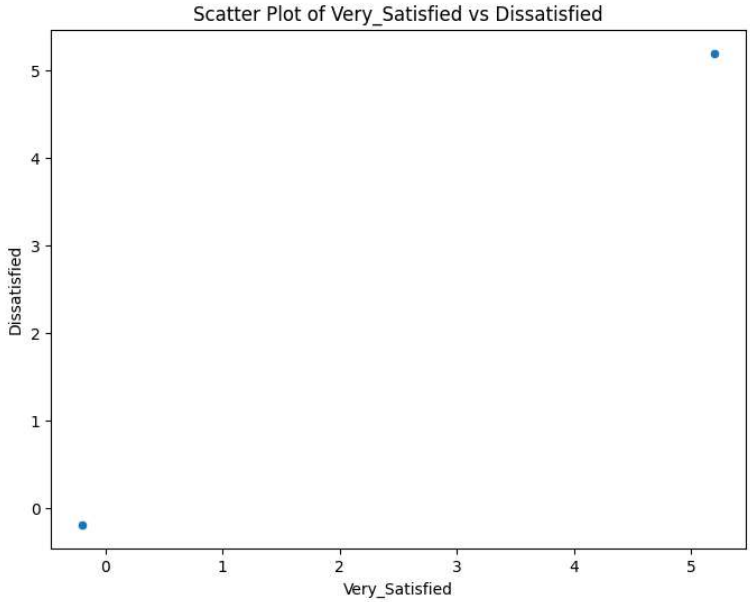

Scatter Plot of Very_Satisfied vs Year
Year
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
Very_Satisfied

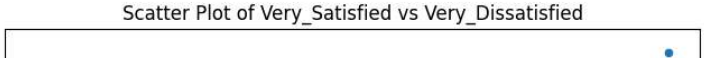

Scatter Plot of Very_Satisfied vs Number_of_Respondents
Number_of_Respondents
5
4
3
2
1
0

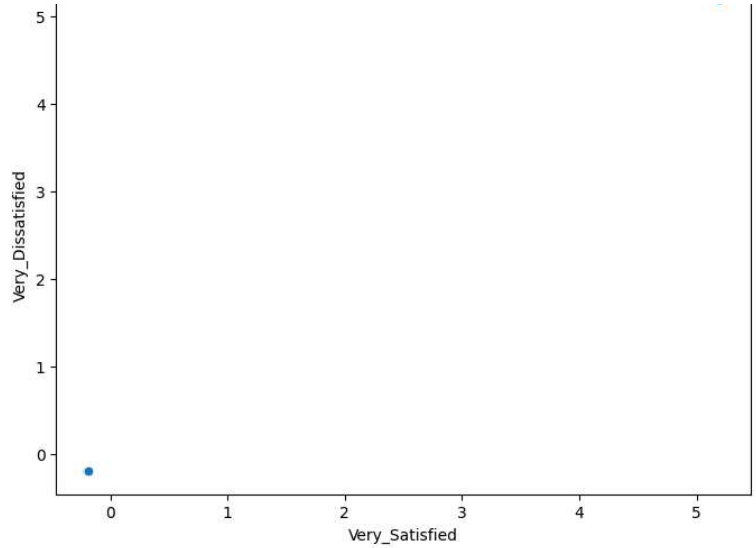

0
1
2
3
4
5
Very_Satisfied

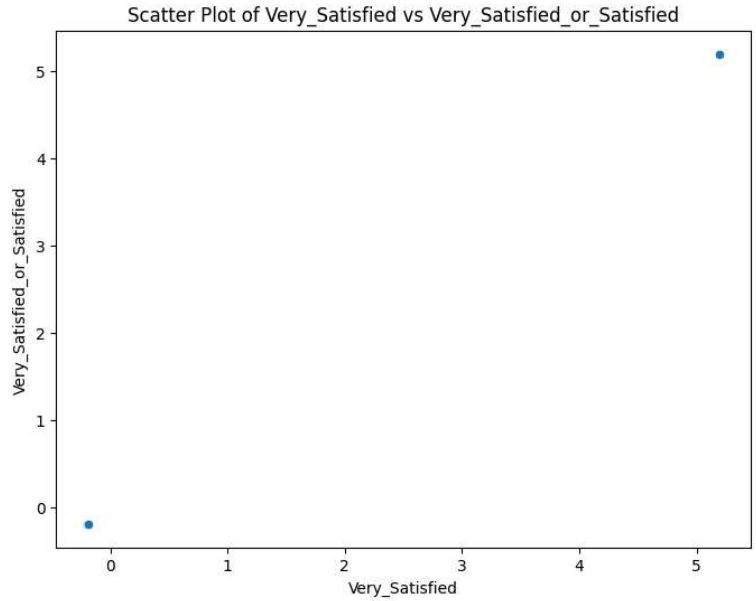

Scatter Plot of Very_Satisfied vs Satisfied
Satisfied
Very_Satisfied
0
1
2
3
4
5

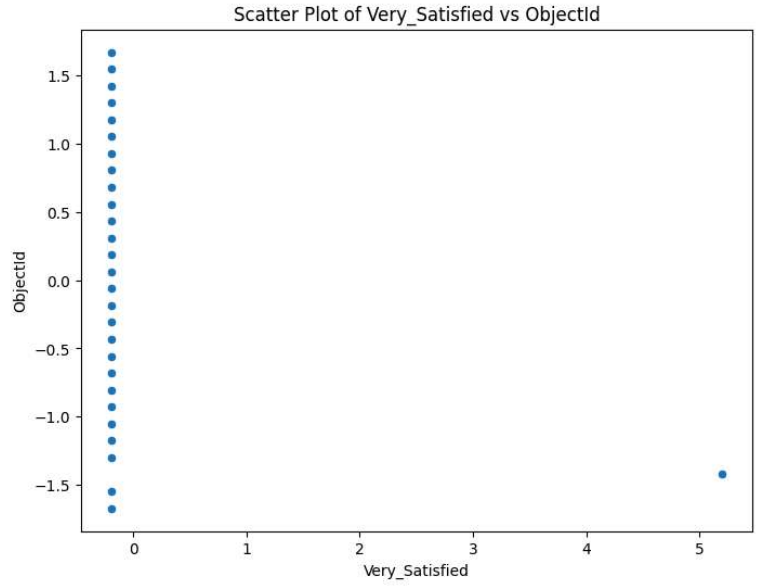

Scatter Plot of Very_Satisfied vs Neutral
Neutral
Very_Satisfied
0
1
2
3
4
5

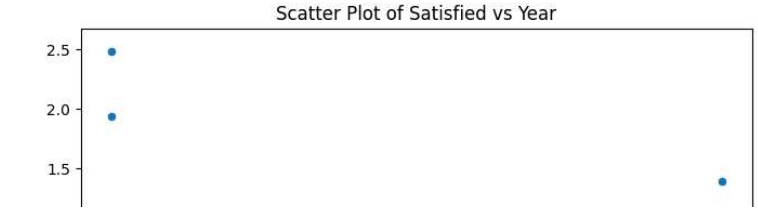

Scatter Plot of Very_Satisfied vs Dissatisfied
Dissatisfied
Very_Satisfied
0
1
2
3
4
5

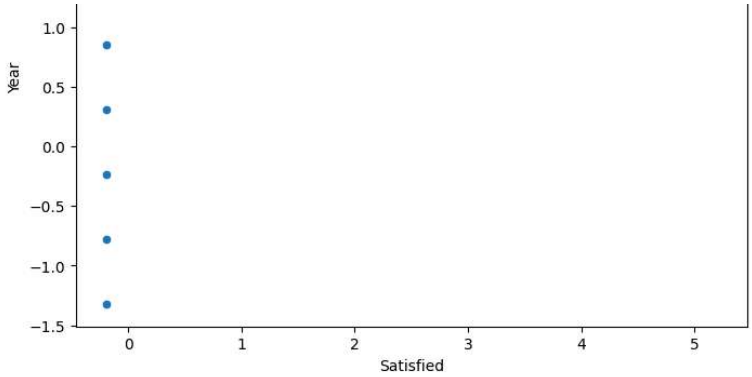

Scatter Plot of Very_Satisfied vs Very_Dissatisfied

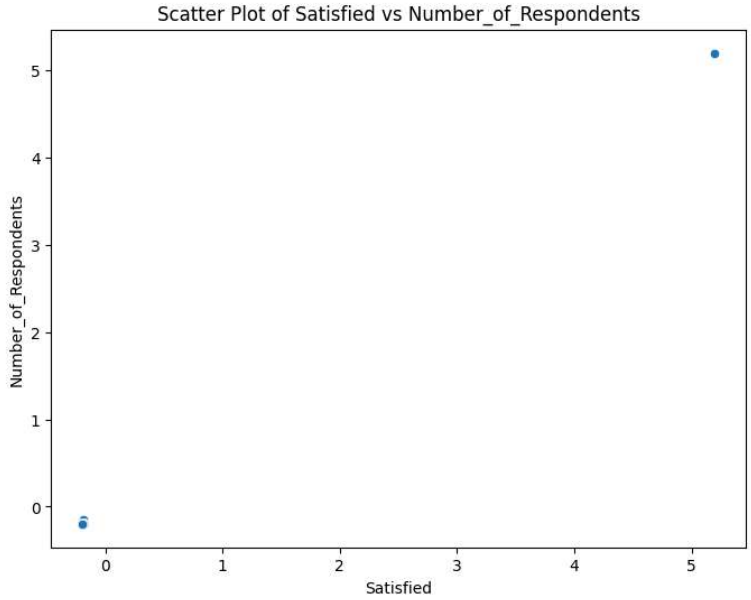

Very_Dissatisfied
Very_Satisfied

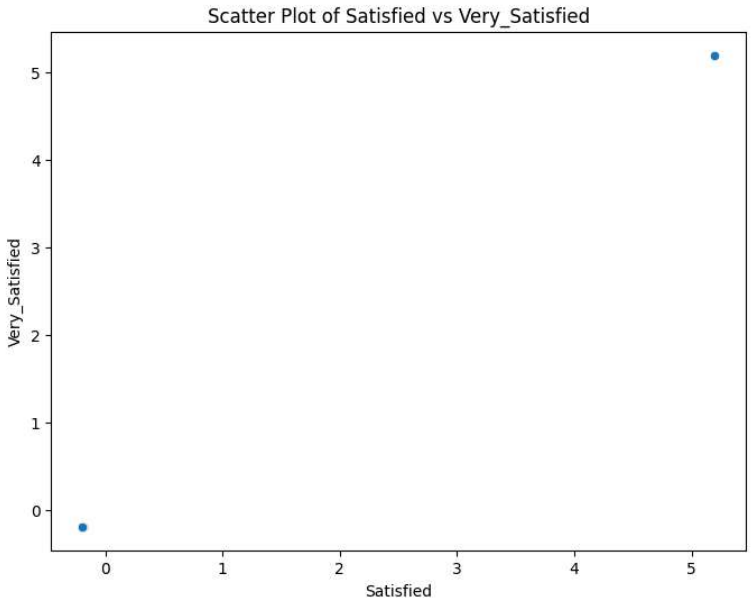

Scatter Plot of Very_Satisfied vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
Very_Satisfied

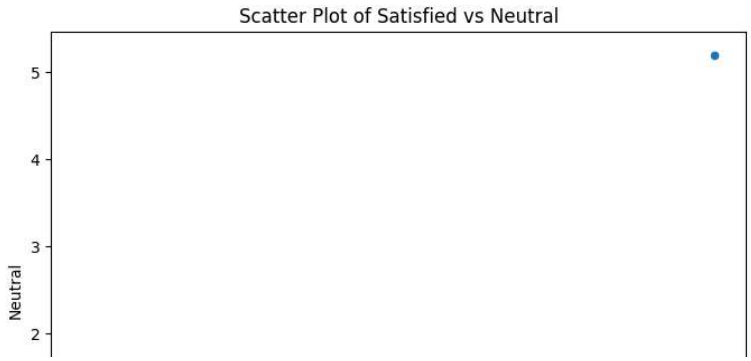

Scatter Plot of Very_Satisfied vs ObjectId
ObjectId
Very_Satisfied

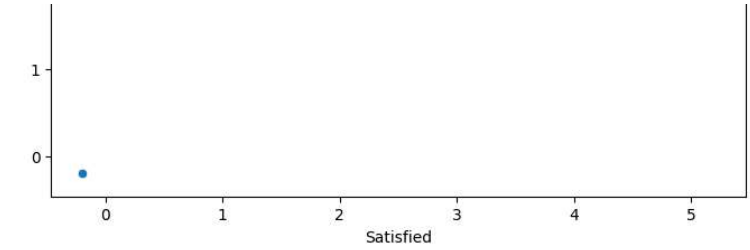

Scatter Plot of Satisfied vs Year

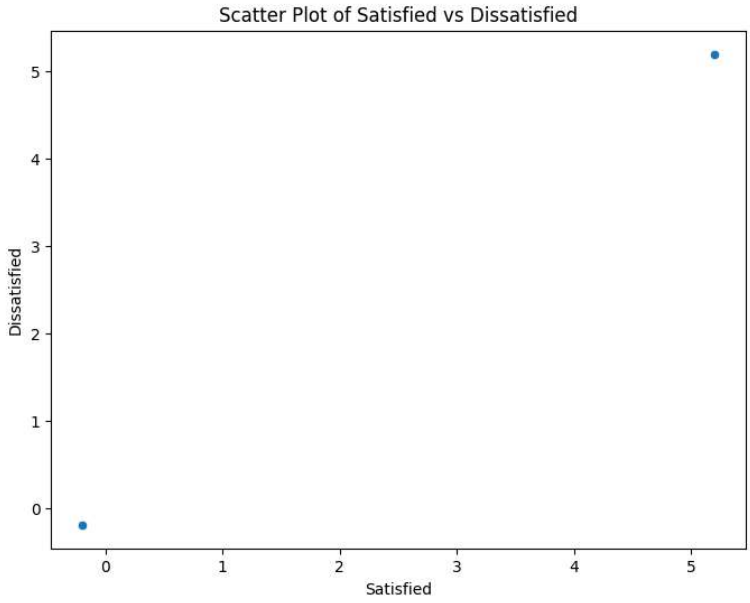

Year
Satisfied
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5

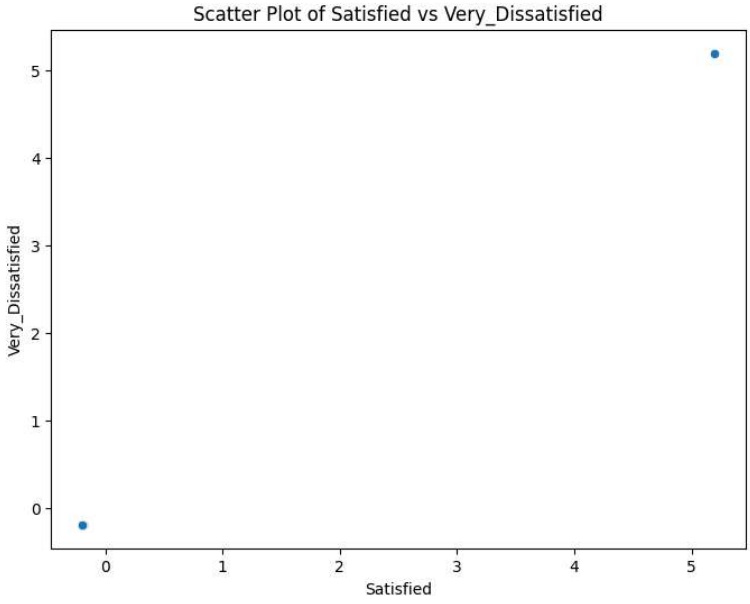

Scatter Plot of Satisfied vs Number_of_Respondents
Number_of_Respondents
Satisfied
5
4
3
2
1
0
0
1
2
3
4
5

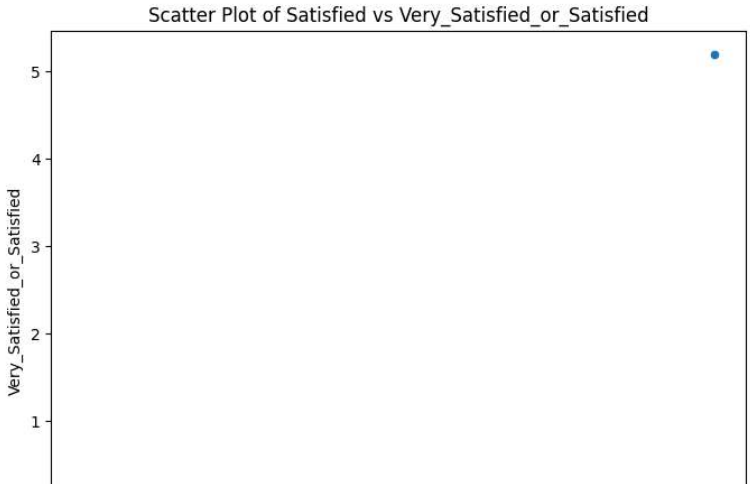

Scatter Plot of Satisfied vs Very_Satisfied
Very_Satisfied
Satisfied
5
4
3
2
1
0
0
1
2
3
4
5


Scatter Plot of Satisfied vs Neutral
Neutral
5
4
3
2

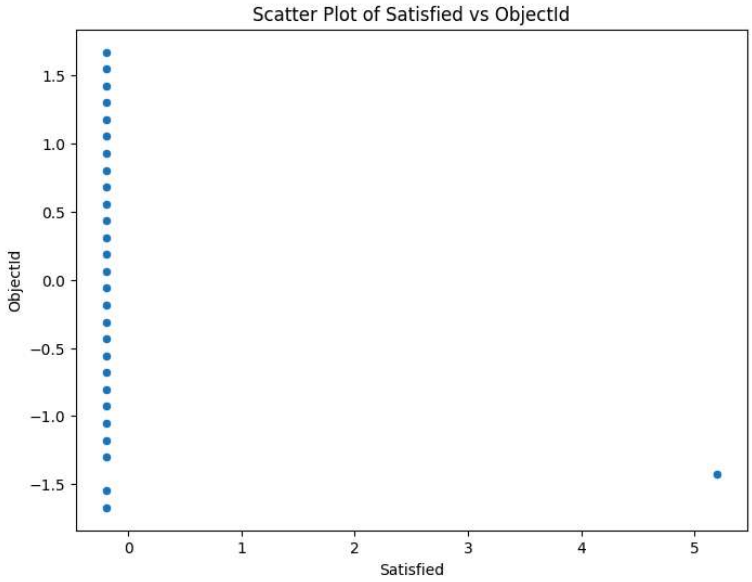

1
0
0
1
2
3
4
5
Satisfied

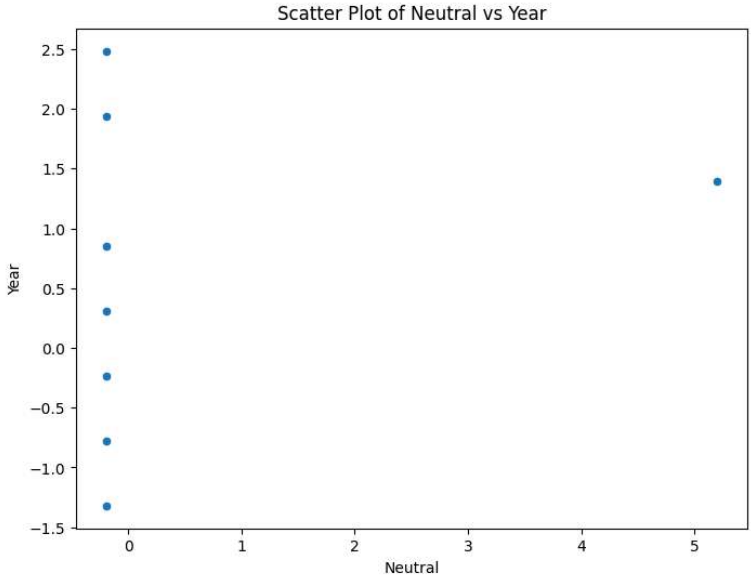

Scatter Plot of Satisfied vs Dissatisfied
Dissatisfied
5
4
3
2
1
0
0
1
2
3
4
5
Satisfied

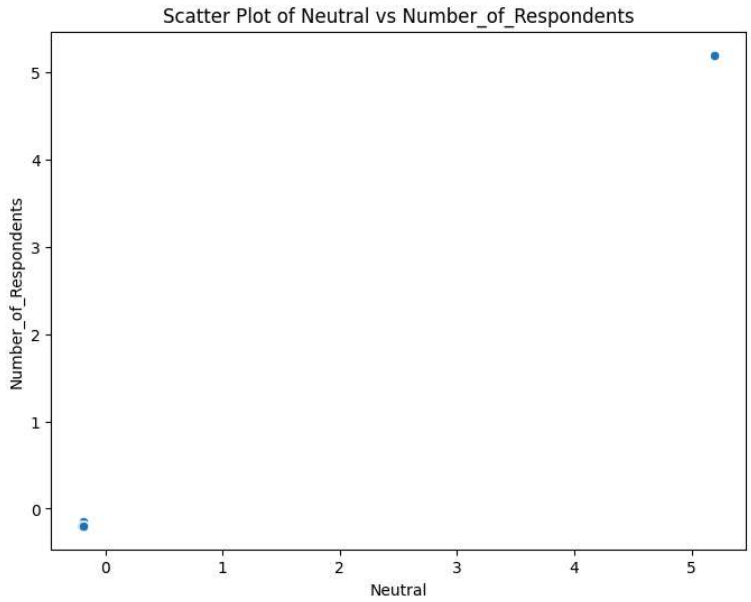

Scatter Plot of Satisfied vs Very_Dissatisfied
Very_Dissatisfied
5
4
3
2
1
0
0
1
2
3
4
5
Satisfied

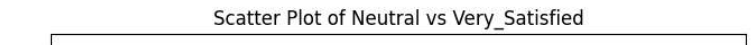

Scatter Plot of Satisfied vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
5
4
3
2
1

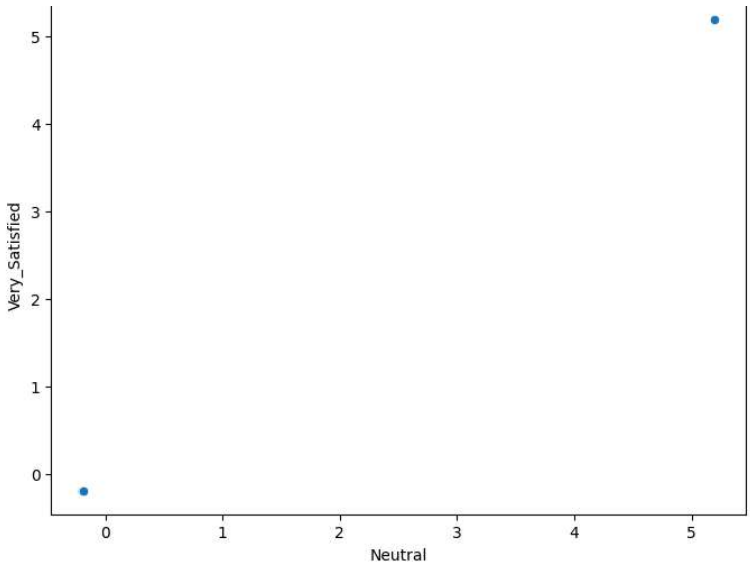

0
0
1
2
3
4
5
Satisfied


Scatter Plot of Satisfied vs ObjectId
ObjectId
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
Satisfied

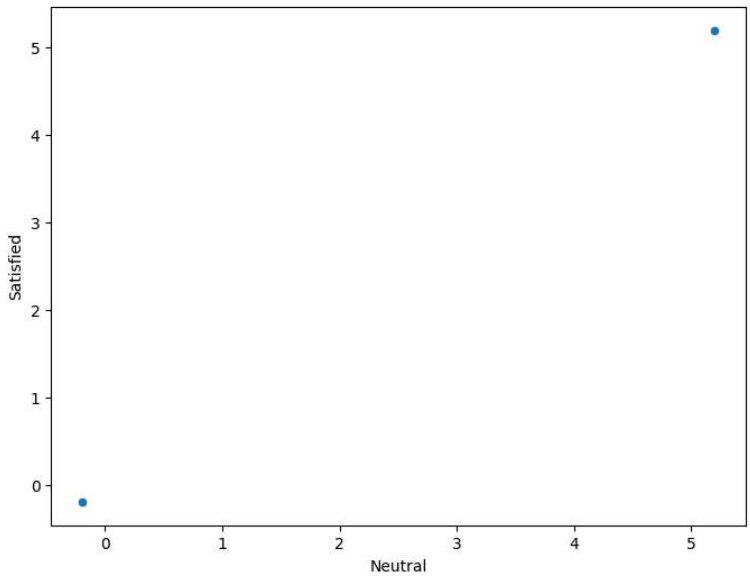

Scatter Plot of Neutral vs Year
Year
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
Neutral


Scatter Plot of Neutral vs Number_of_Respondents
Number_of_Respondents
5
4
3
2
1
0
0
1
2
3
4
5
Neutral

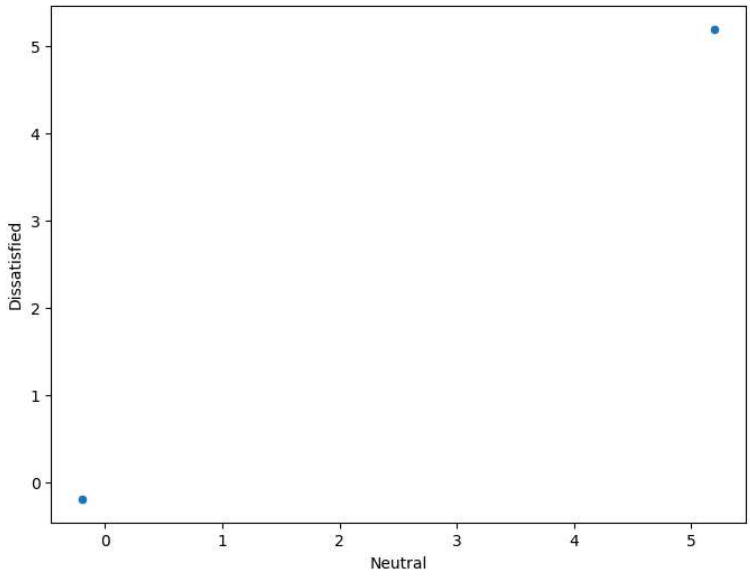

Scatter Plot of Neutral vs Very_Satisfied



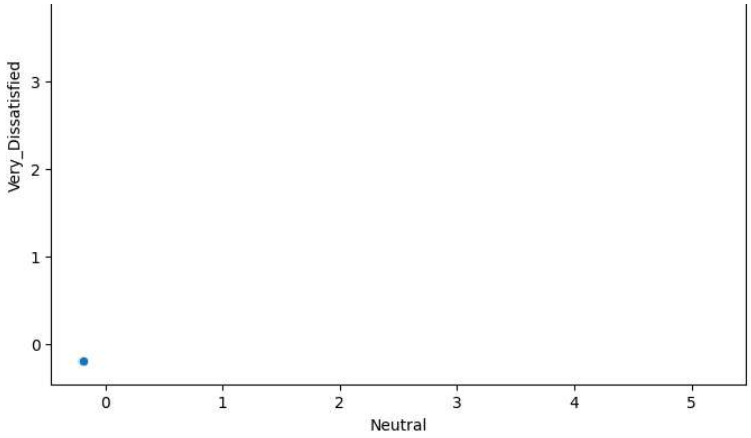
Scatter Plot of Neutral vs Satisfied

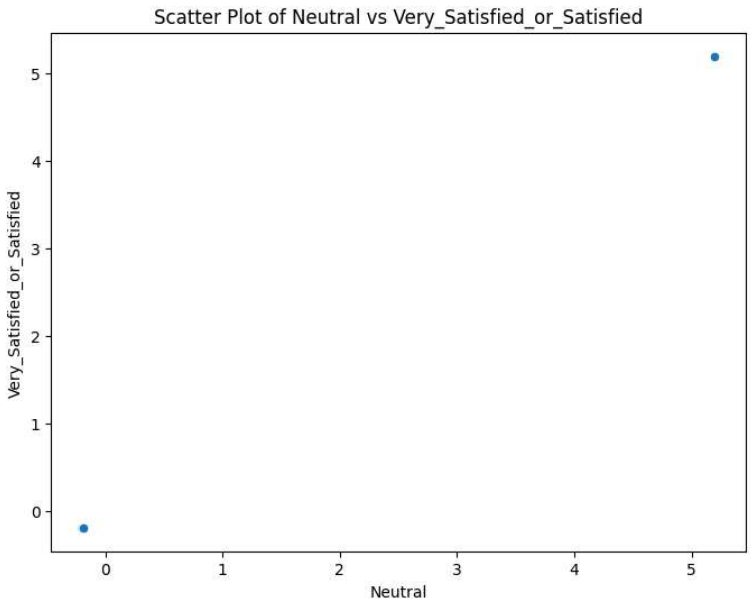


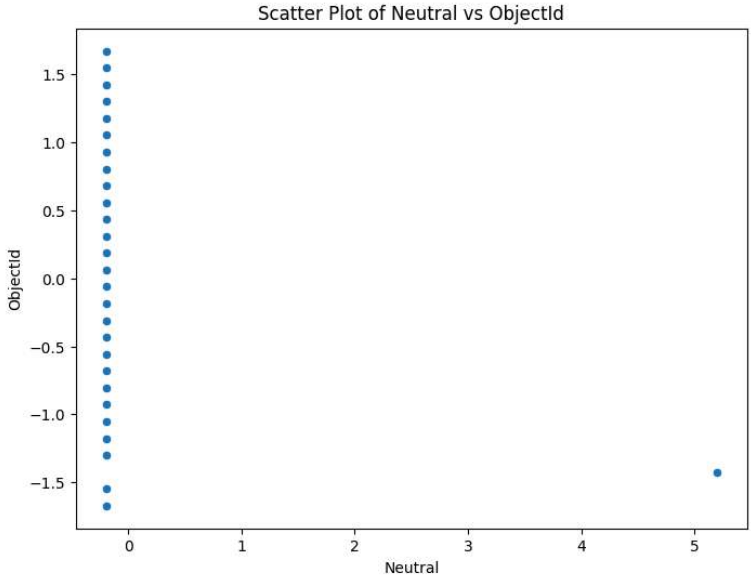
Scatter Plot of Neutral vs Dissatisfied

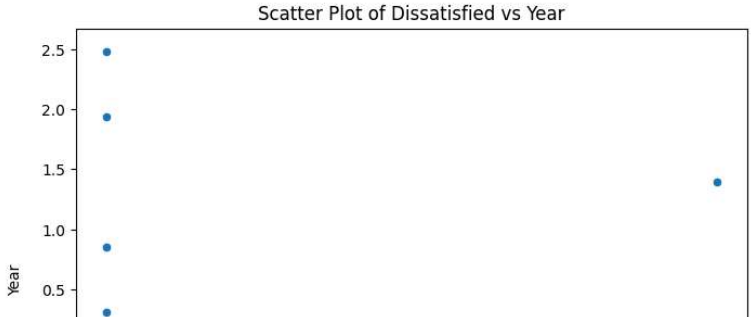


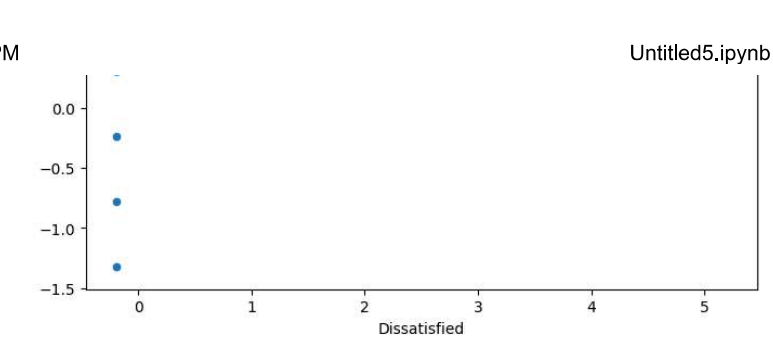
Scatter Plot of Neutral vs Very_Dissatisfied

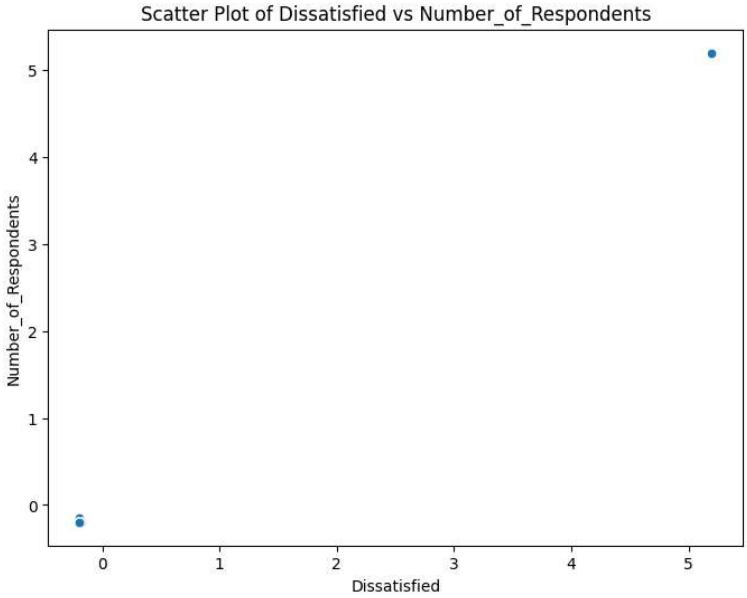

Very_Dissatisfied
Neutral

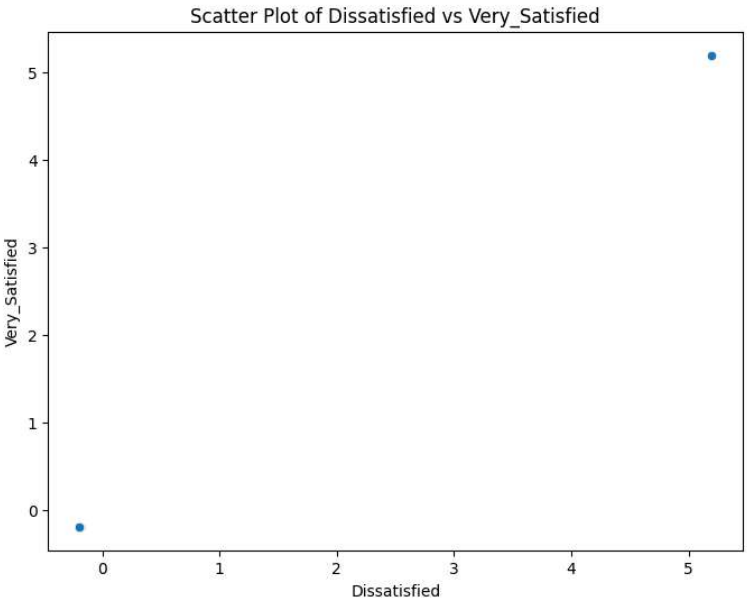

Scatter Plot of Neutral vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
Neutral


Scatter Plot of Neutral vs ObjectId
ObjectId
Neutral

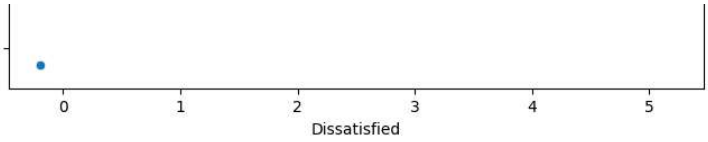

Scatter Plot of Dissatisfied vs Year
Year

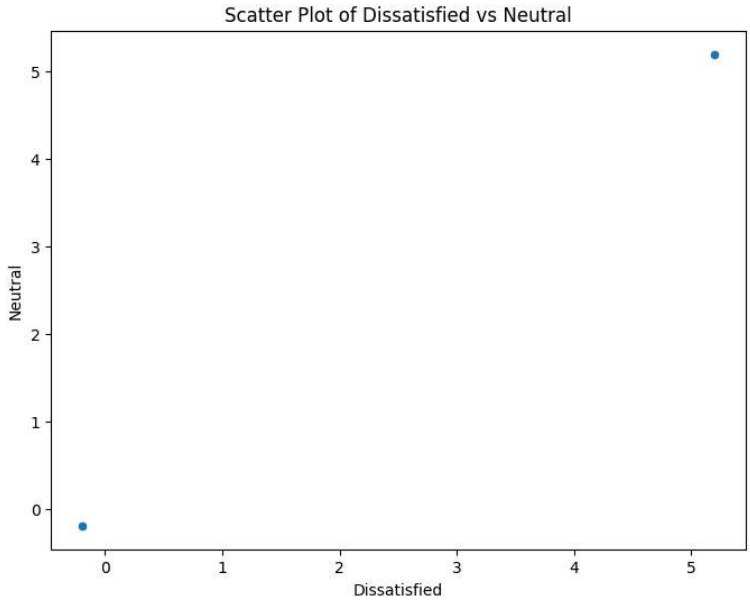

0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
Dissatisfied

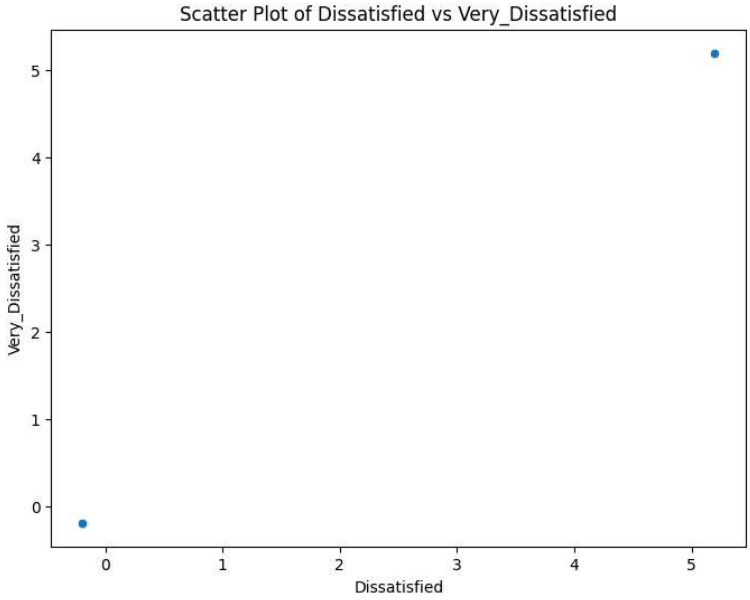

Scatter Plot of Dissatisfied vs Number_of_Respondents
Number_of_Respondents
Dissatisfied
0
1
2
3
4
5

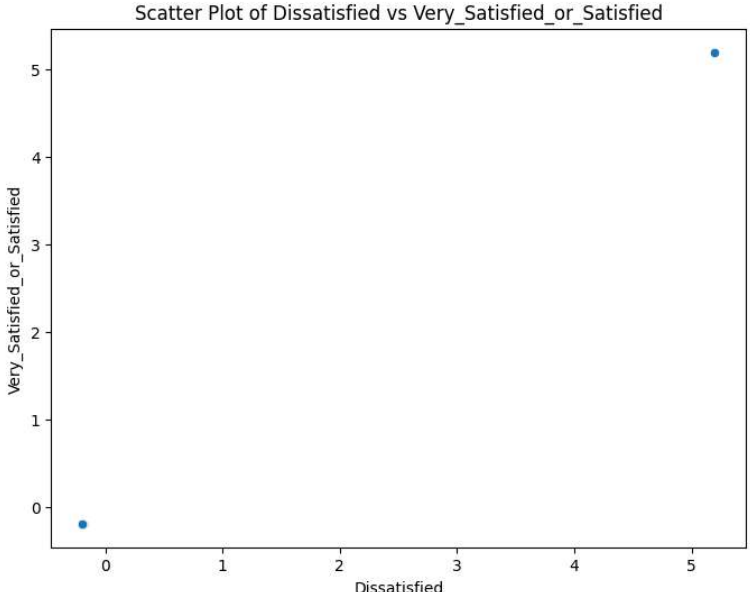

Scatter Plot of Dissatisfied vs Very_Satisfied
Very_Satisfied
Dissatisfied
0
1
2
3
4
5

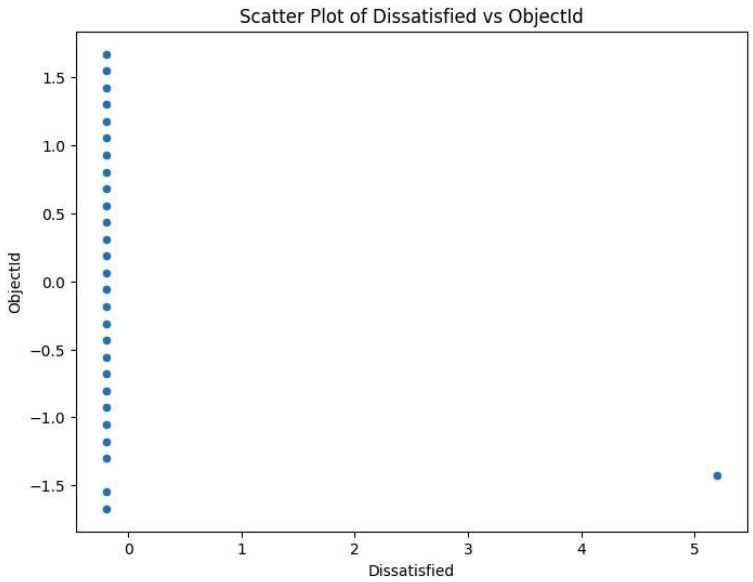

Scatter Plot of Dissatisfied vs Satisfied
Satisfied
1
2
3
4
5

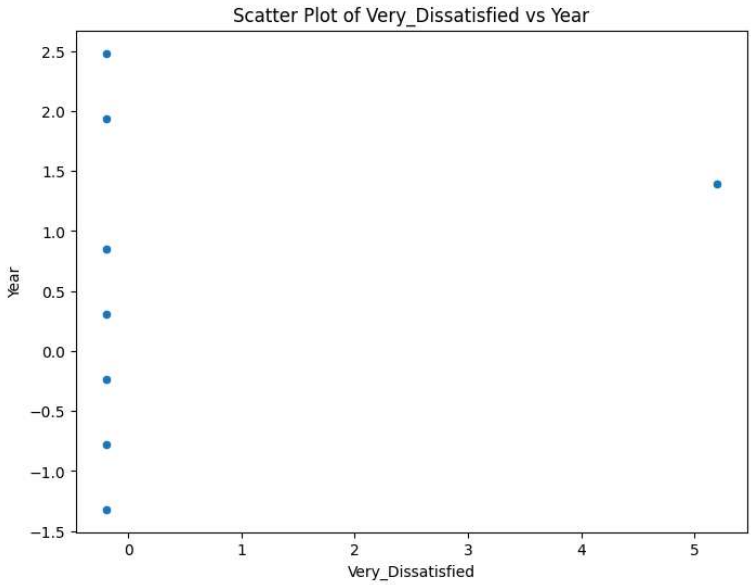

0
0
1
2
3
4
5
Dissatisfied

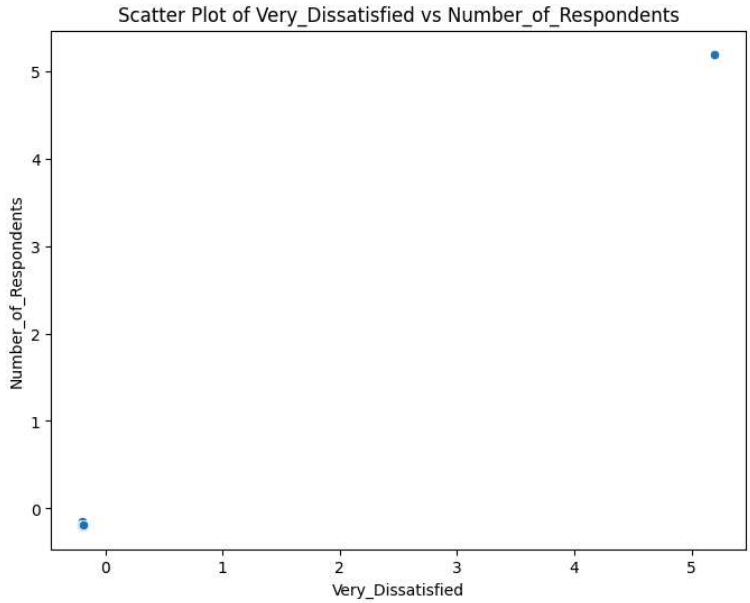

Scatter Plot of Dissatisfied vs Neutral
Neutral
Dissatisfied
0
1
2
3
4
5

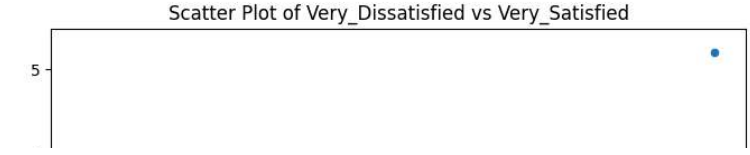

Scatter Plot of Dissatisfied vs Very_Dissatisfied
Very_Dissatisfied
Dissatisfied
0
1
2
3
4
5

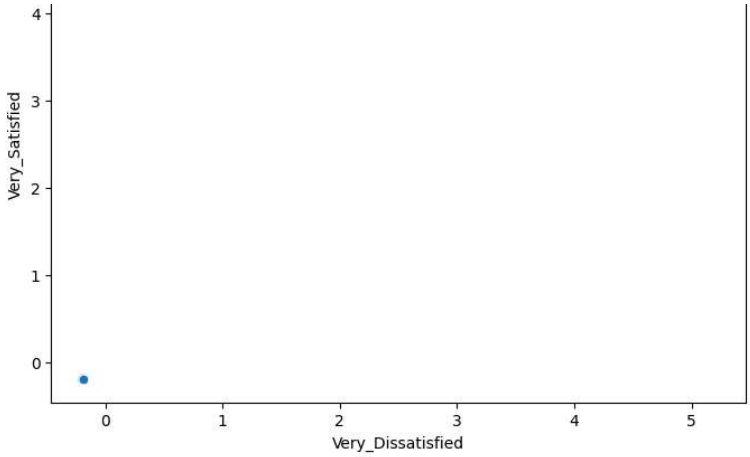

Scatter Plot of Dissatisfied vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
Dissatisfied
0
1
2
3
4
5


Scatter Plot of Dissatisfied vs ObjectId
ObjectId
Dissatisfied

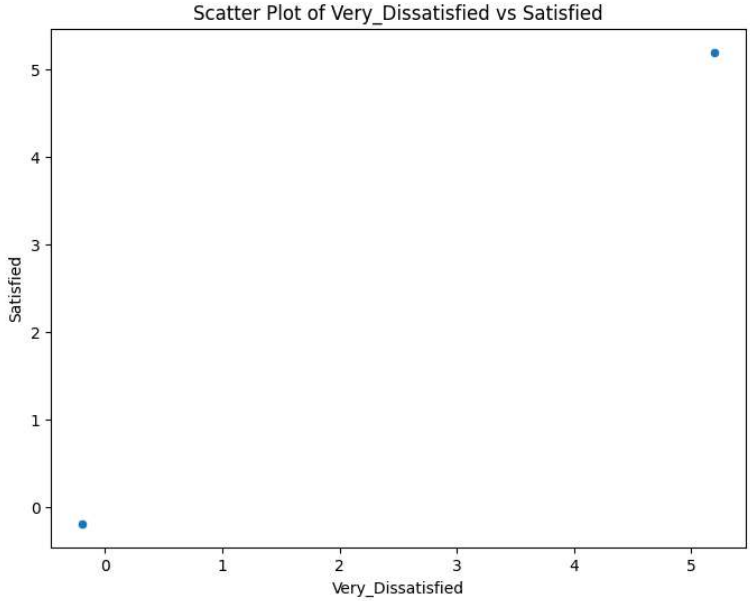

Scatter Plot of Very_Dissatisfied vs Year
Year
Very_Dissatisfied


Scatter Plot of Very_Dissatisfied vs Number_of_Respondents
Number_of_Respondents
Very_Dissatisfied

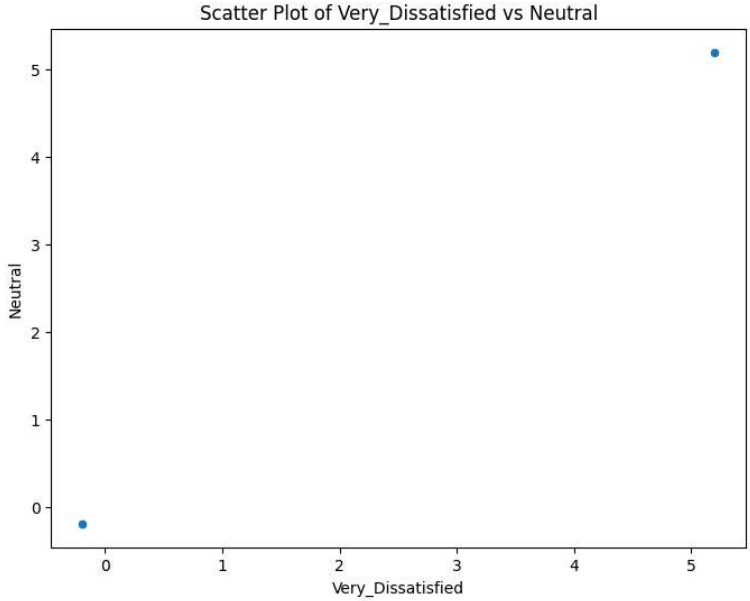

Scatter Plot of Very_Dissatisfied vs Very_Satisfied



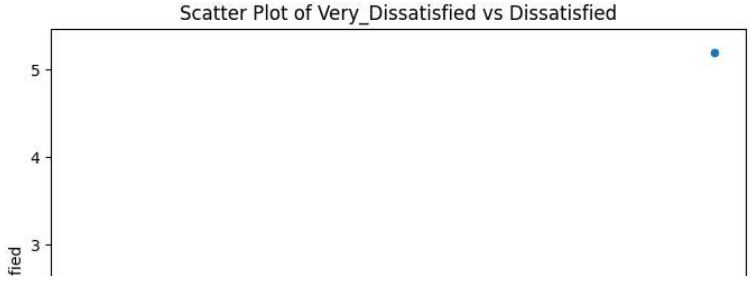
Scatter Plot of Very_Dissatisfied vs Satisfied

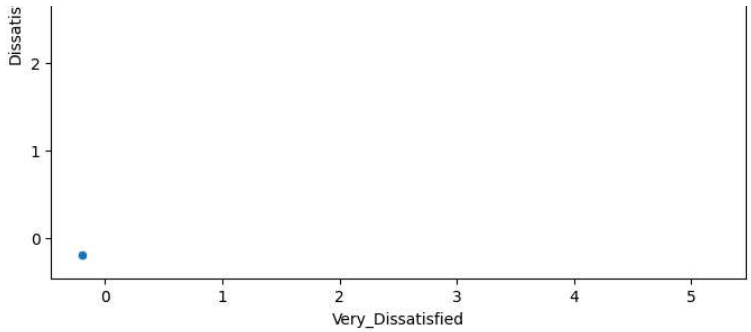


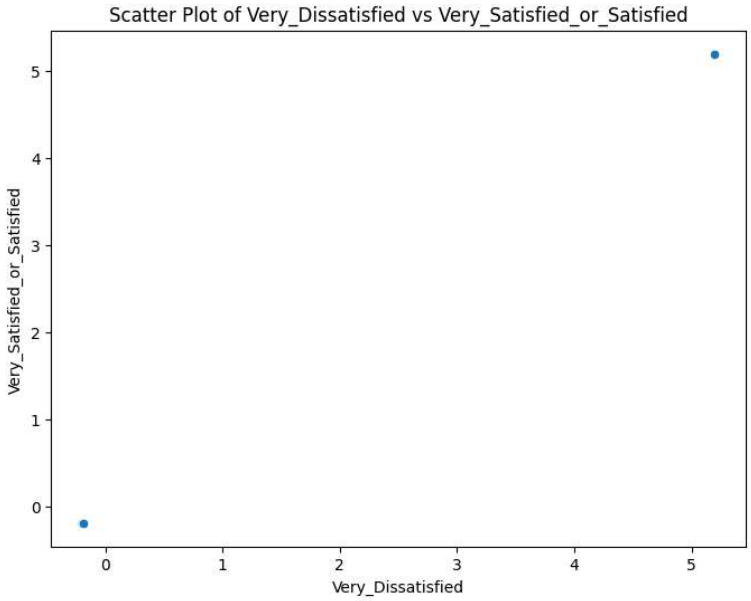
Scatter Plot of Very_Dissatisfied vs Neutral

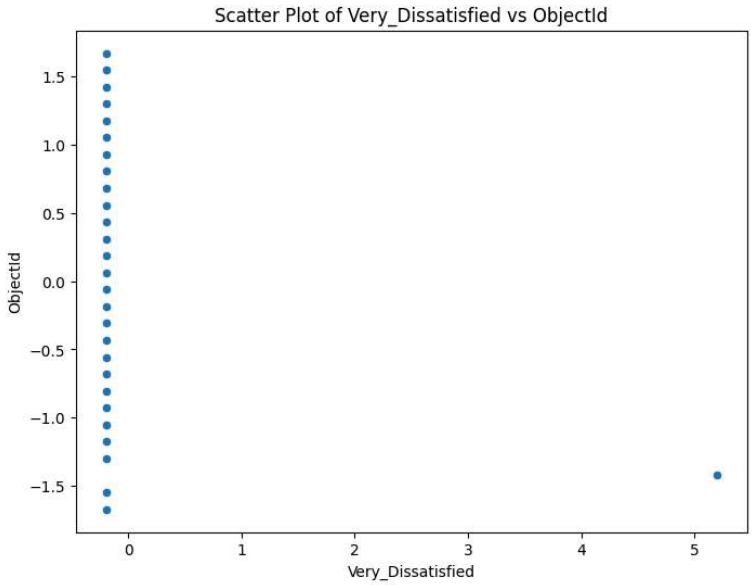


Scatter Plot of Very_Dissatisfied vs Dissatisfied



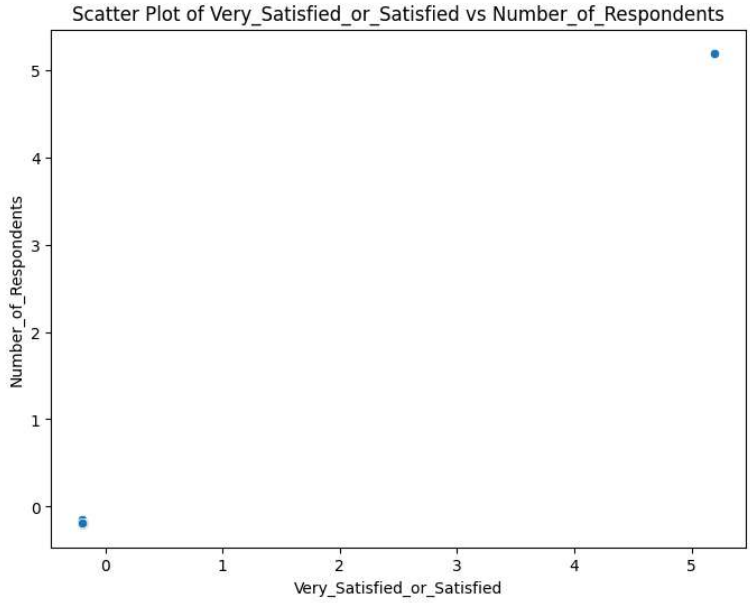

Dissatis
Very_Dissatisfied

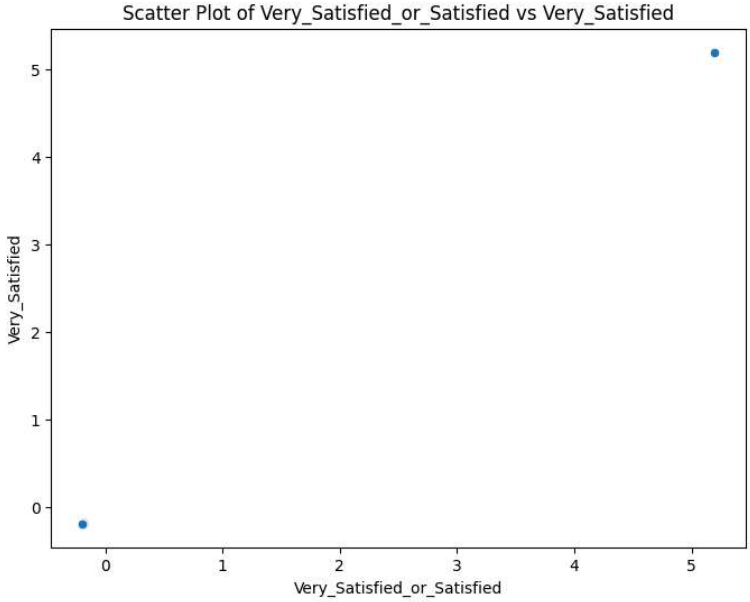

Scatter Plot of Very_Dissatisfied vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
Very_Dissatisfied

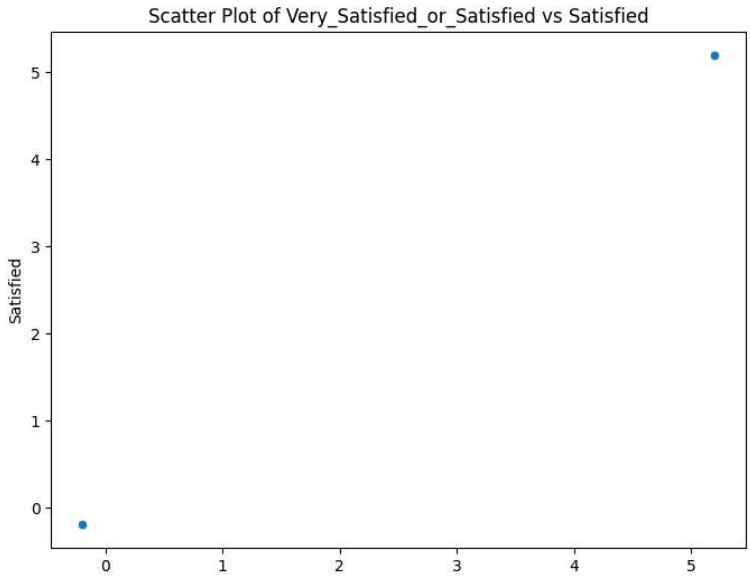

Scatter Plot of Very_Dissatisfied vs ObjectId
ObjectId
Very_Dissatisfied

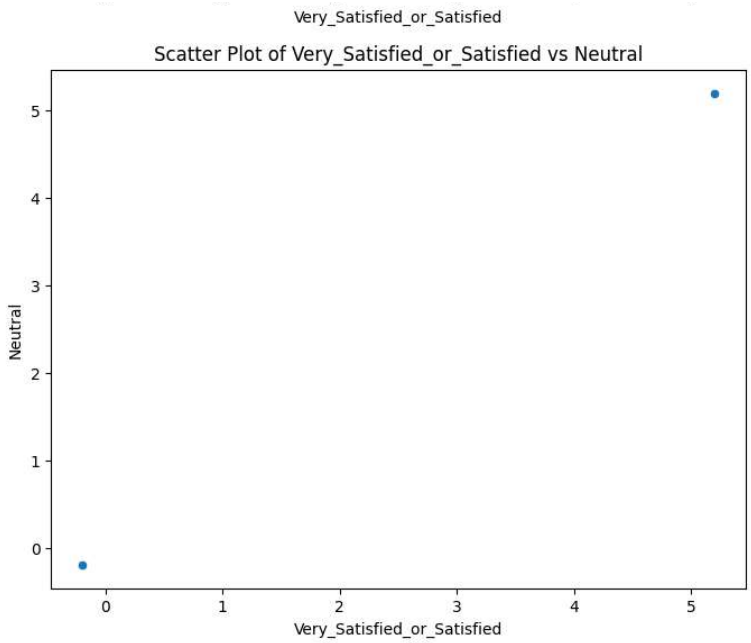

Scatter Plot of Very_Satisfied_or_Satisfied vs Year
Year

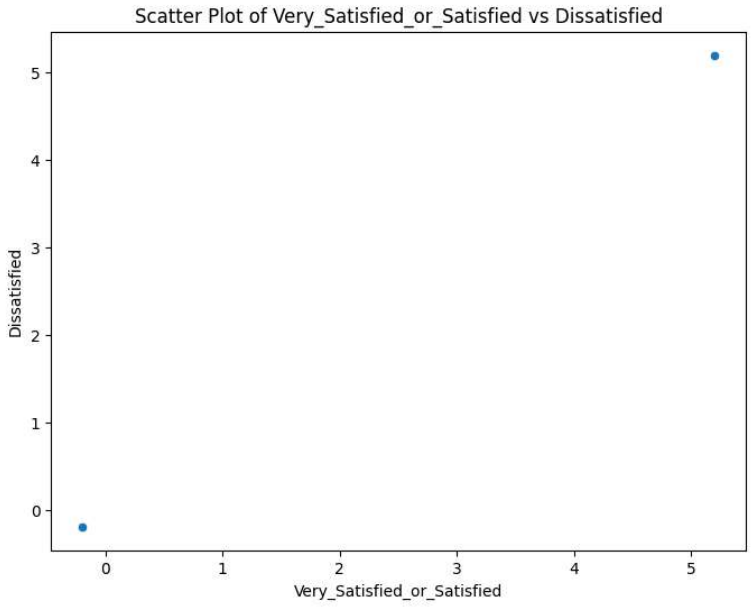

−1.0
−1.5
0
1
2
3
4
5
Very_Satisfied_or_Satisfied

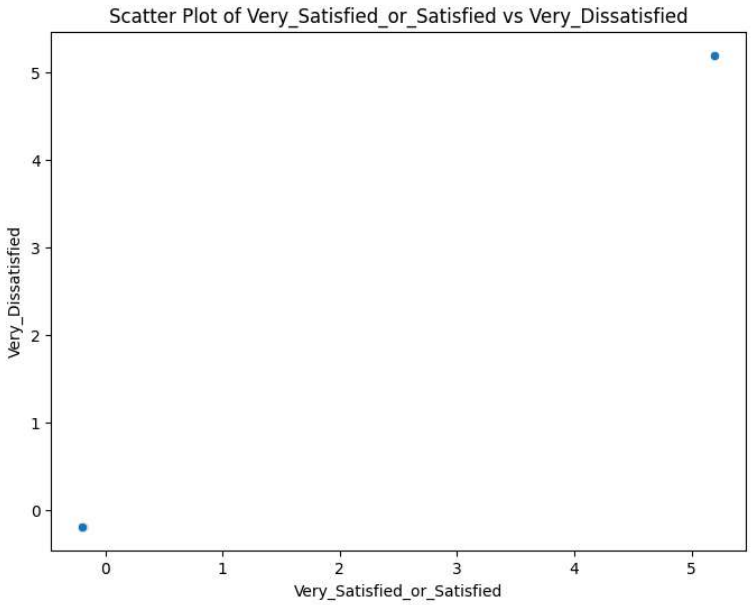

Scatter Plot of Very_Satisfied_or_Satisfied vs Number_of_Respondents
Number_of_Respondents
Very_Satisfied_or_Satisfied
0
1
2
3
4
5

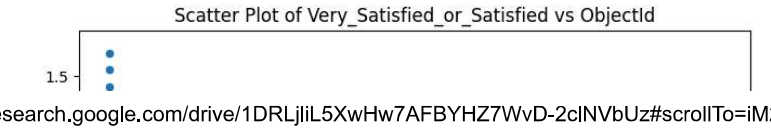

Scatter Plot of Very_Satisfied_or_Satisfied vs Very_Satisfied
Very_Satisfied
Very_Satisfied_or_Satisfied
0
1
2
3
4
5

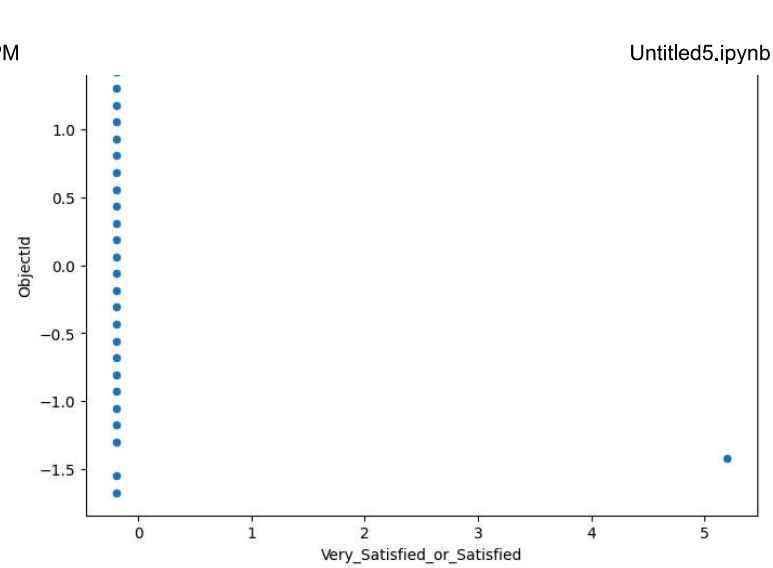

Scatter Plot of Very_Satisfied_or_Satisfied vs Satisfied
Satisfied
0
1
2
3
4
5

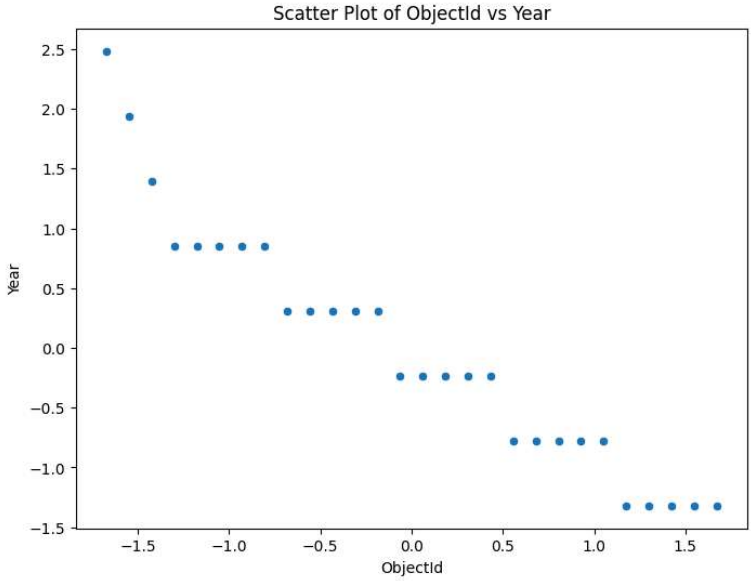

Very_Satisfied_or_Satisfied
Scatter Plot of Very_Satisfied_or_Satisfied vs Neutral
Neutral
Very_Satisfied_or_Satisfied

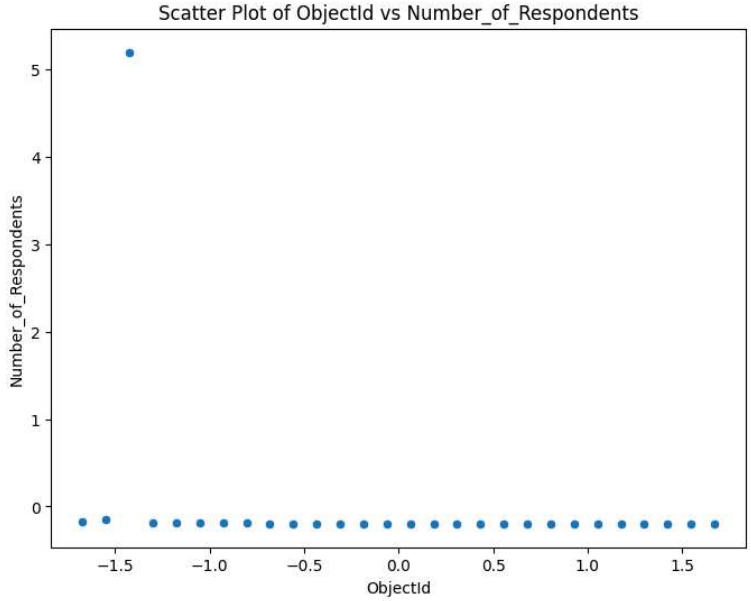

Scatter Plot of Very_Satisfied_or_Satisfied vs Dissatisfied
Dissatisfied
Very_Satisfied_or_Satisfied

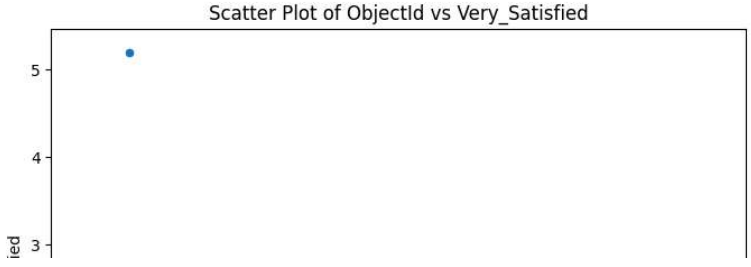

Scatter Plot of Very_Satisfied_or_Satisfied vs Very_Dissatisfied
Very_Dissatisfied
Very_Satisfied_or_Satisfied

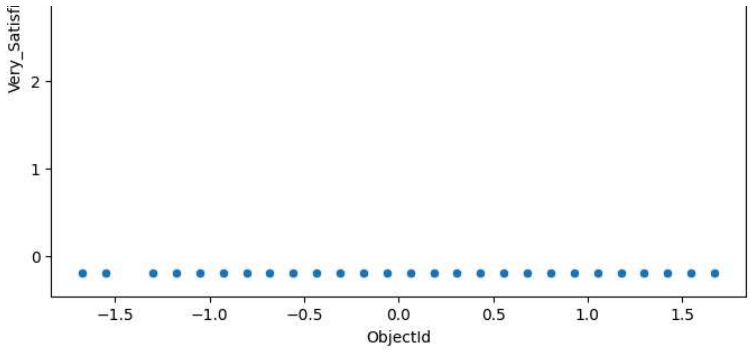

Scatter Plot of Very_Satisfied_or_Satisfied vs ObjectId
1.5

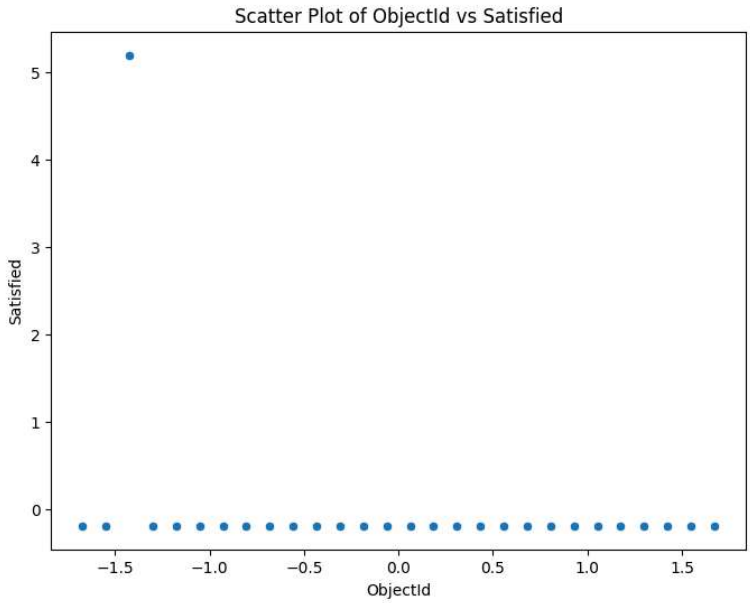

ObjectId
Very_Satisfied_or_Satisfied

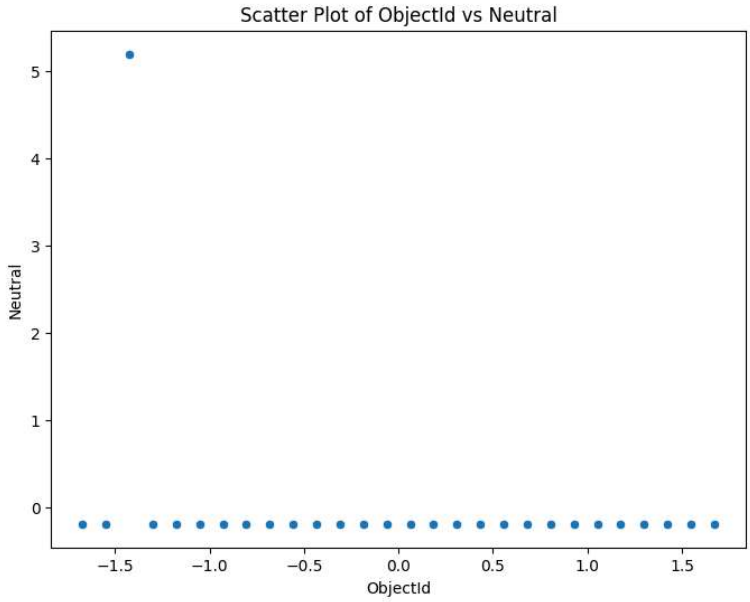

Scatter Plot of ObjectId vs Year
Year
ObjectId

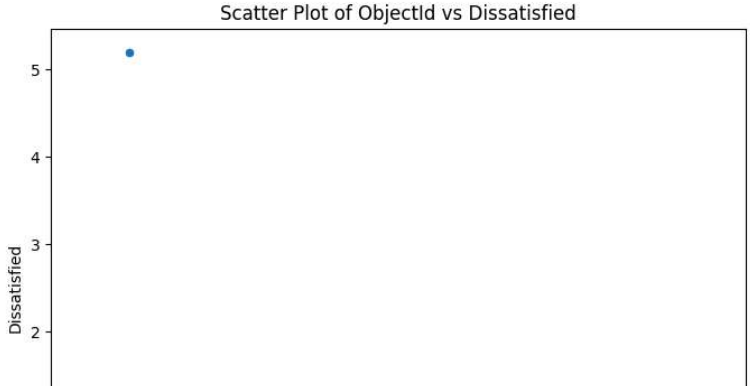

Scatter Plot of ObjectId vs Number_of_Respondents
Number_of_Respondents
ObjectId

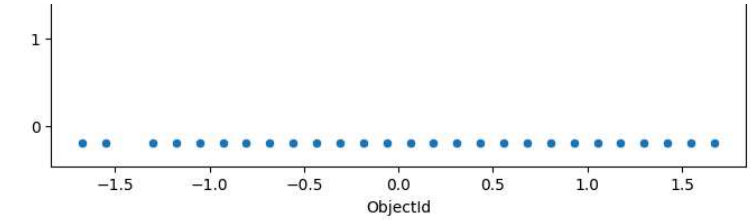

Scatter Plot of ObjectId vs Very_Satisfied

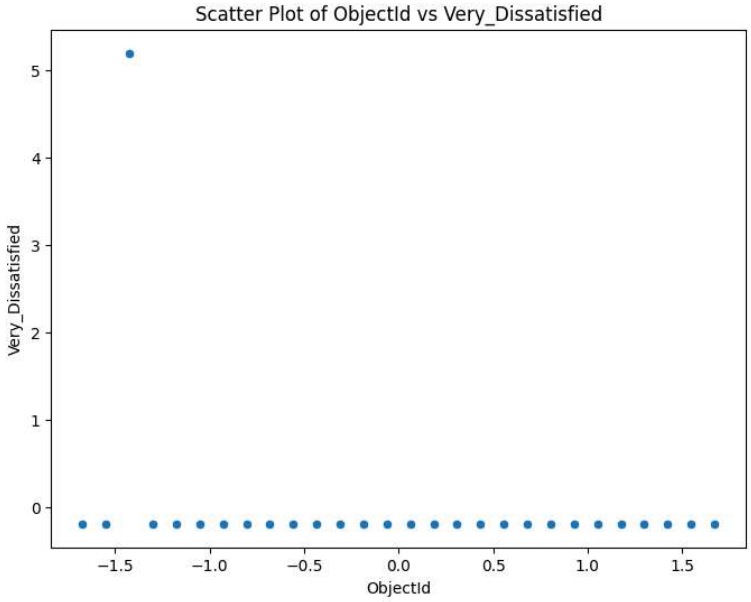

Very_Satisfi
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

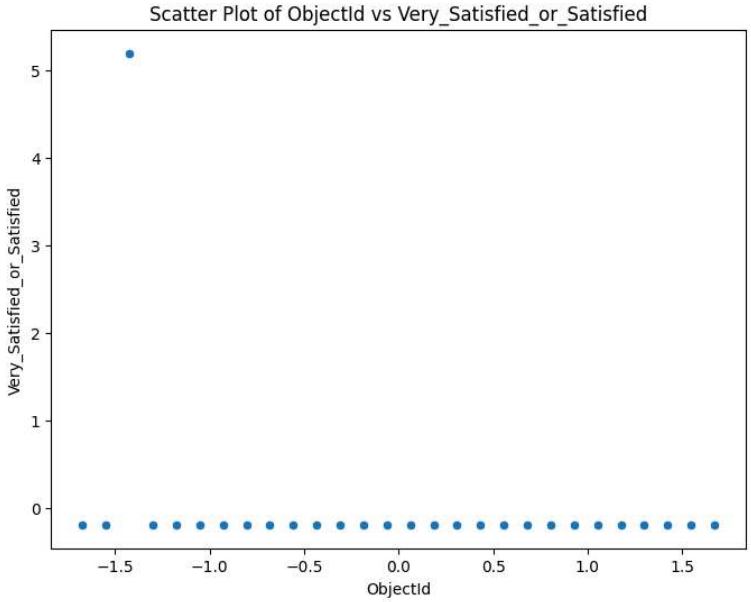

Scatter Plot of ObjectId vs Satisfied
Satisfied
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

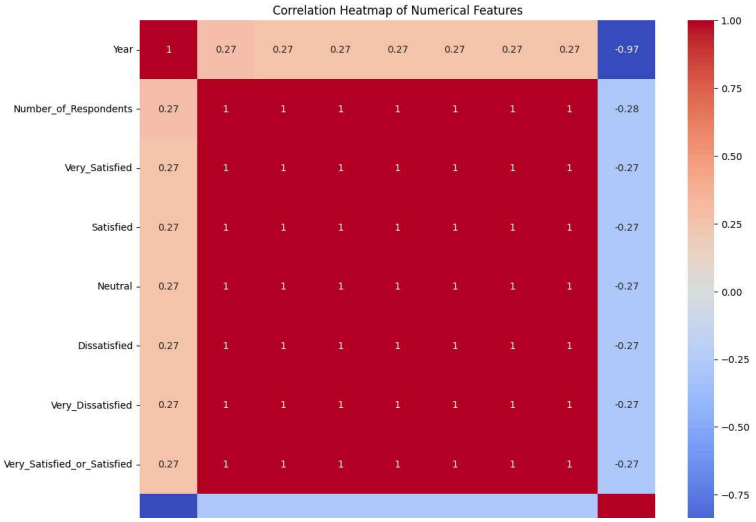

Scatter Plot of ObjectId vs Neutral
Neutral
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

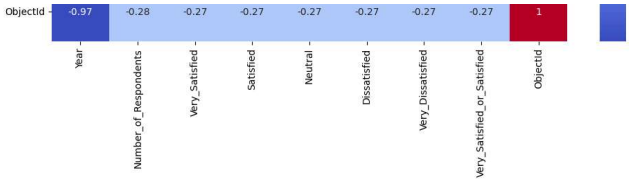

Scatter Plot of ObjectId vs Dissatisfied
Dissatisfied
5
4
3
2

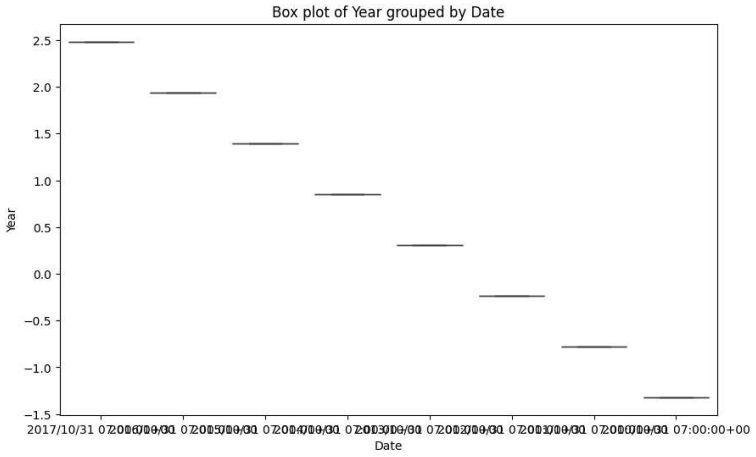

1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

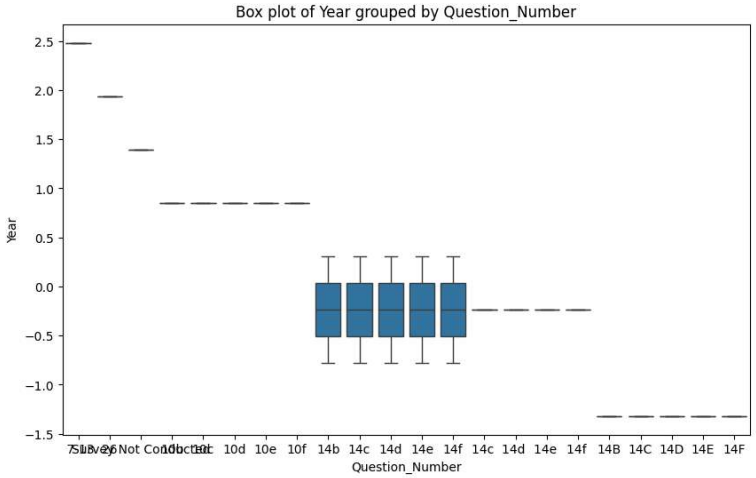

Scatter Plot of ObjectId vs Very_Dissatisfied
Very_Dissatisfied
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

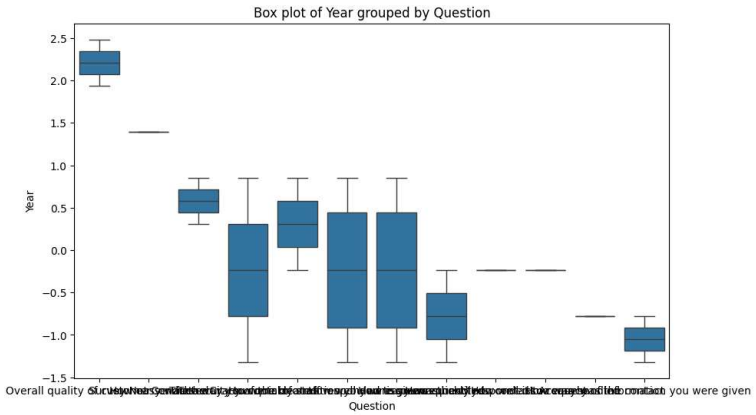

Scatter Plot of ObjectId vs Very_Satisfied_or_Satisfied
Very_Satisfied_or_Satisfied
5
4
3
2
1
0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
ObjectId

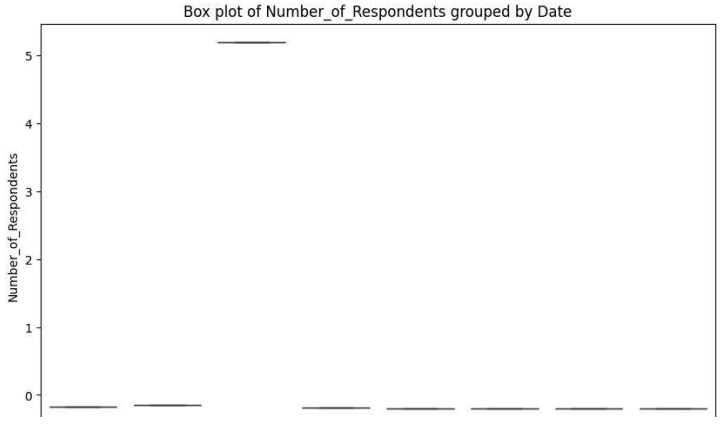

Correlation Heatmap of Numerical Features
Year
Number_of_Respondents
Very_Satisfied
Satisfied
Neutral
Dissatisfied
Very_Dissatisfied
Very_Satisfied_or_Satisfied
1 0.27 0.27 0.27 0.27 0.27 0.27 0.27 -0.97
0.27 1 1 1 1 1 1 1 -0.28
0.27 1 1 1 1 1 1 1 -0.27
0.27 1 1 1 1 1 1 1 -0.27
0.27 1 1 1 1 1 1 1 -0.27
0.27 1 1 1 1 1 1 1 -0.27
0.27 1 1 1 1 1 1 1 -0.27
0.27 1 1 1 1 1 1 1 -0.27
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

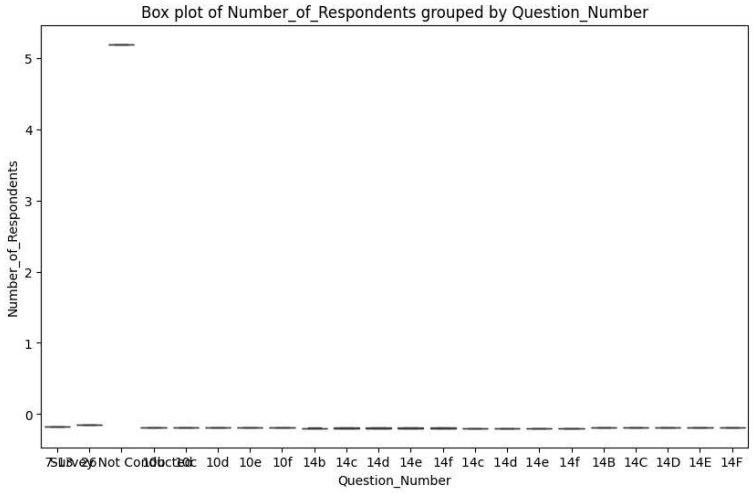

ObjectId
-0.97
-0.28
-0.27
-0.27
-0.27
-0.27
-0.27
-0.27
1
Year
Number_of_Respondents
Very_Satisfied
Satisfied
Neutral
Dissatisfied
Very_Dissatisfied
Very_Satisfied_or_Satisfied
ObjectId

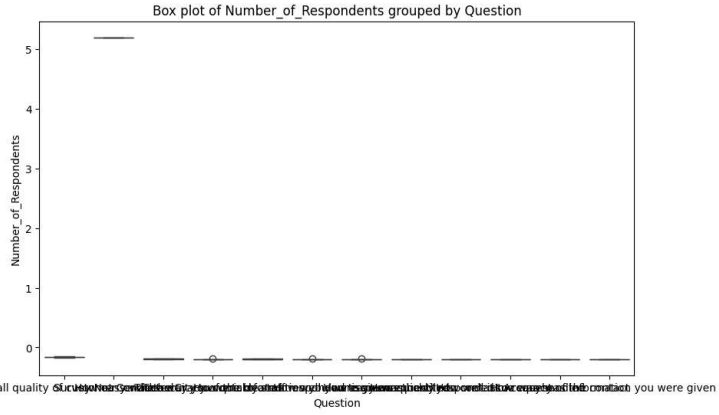

Box plot of Year grouped by Date
Year
Date

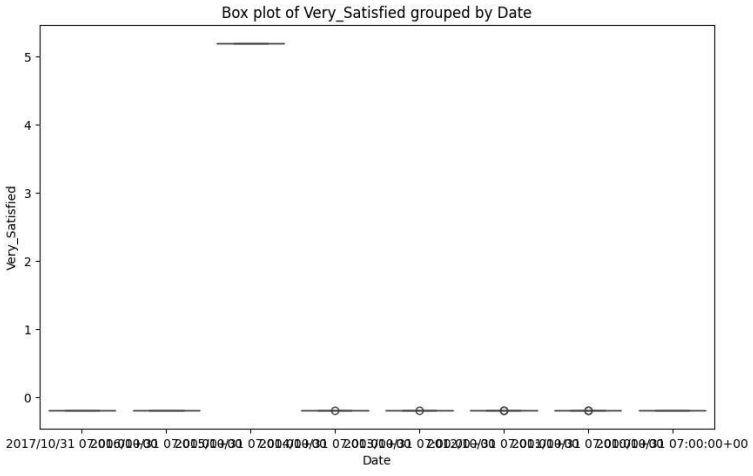

Box plot of Year grouped by Question_Number
Year
Question_Number

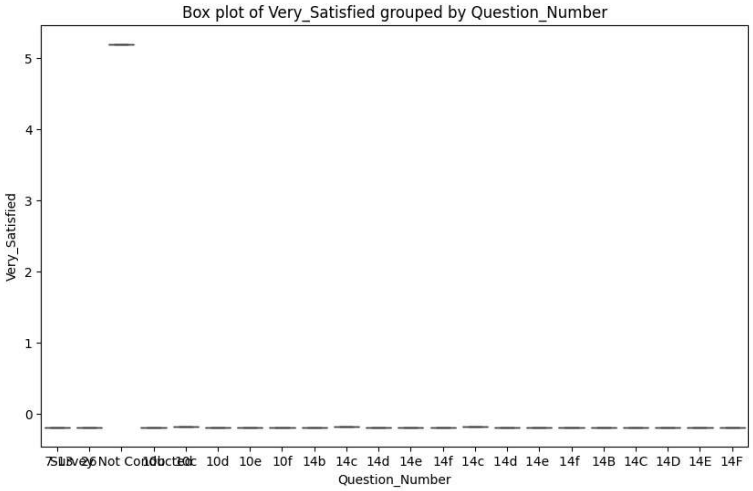

Box plot of Year grouped by Question
Year
Question


Box plot of Number_of_Respondents grouped by Date
Number_of_Respondents

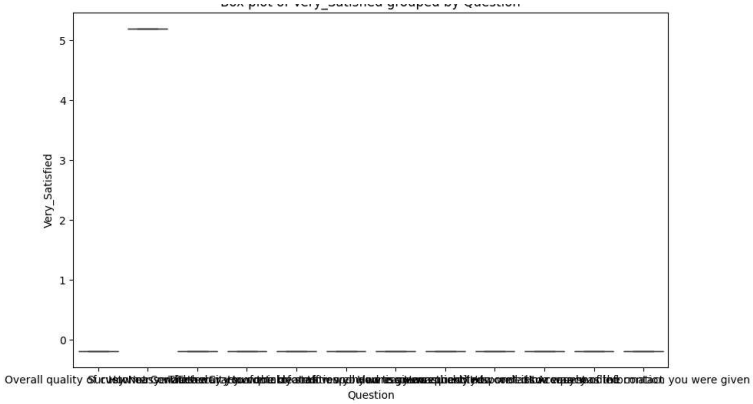


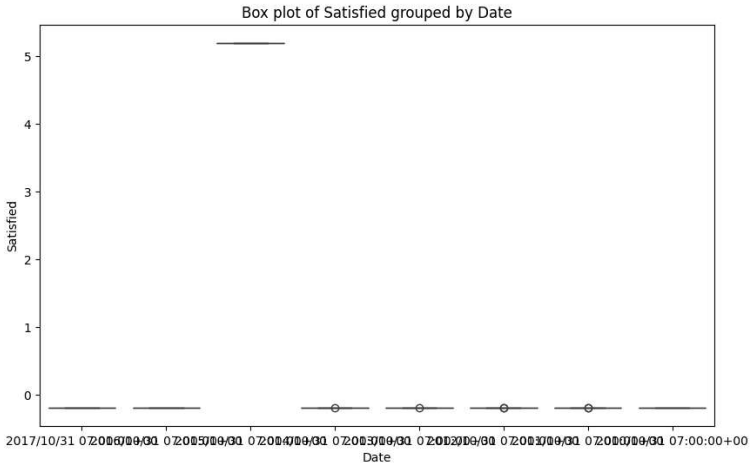


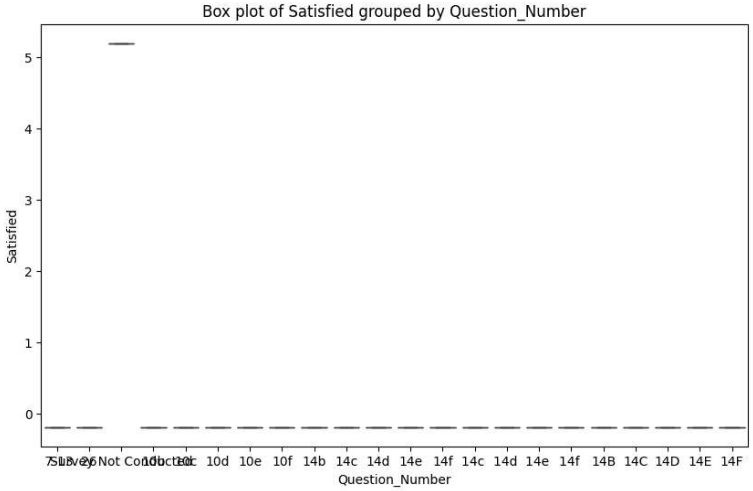


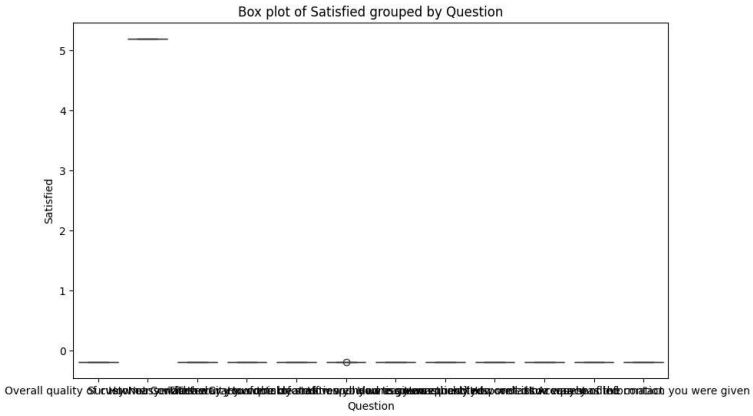


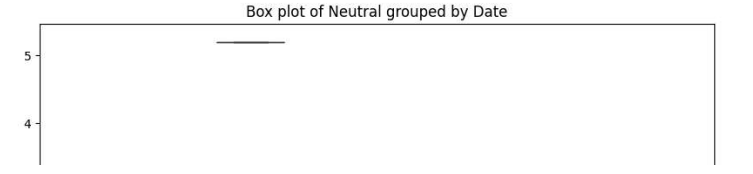
Box plot of Very_Satisfied grouped by Question

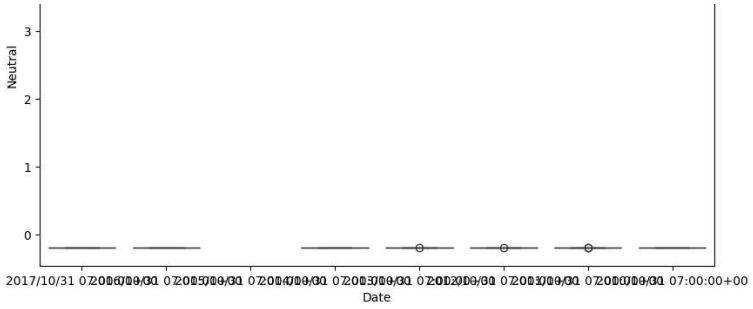

Very_Satisfied
Question

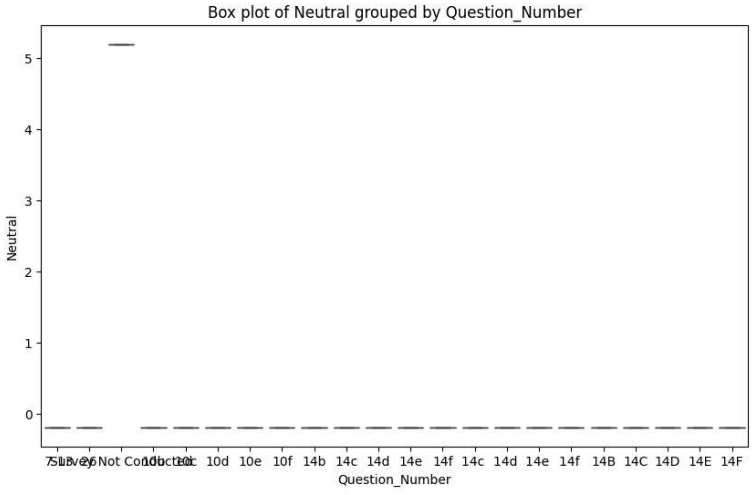

Box plot of Satisfied grouped by Date
Satisfied
Date

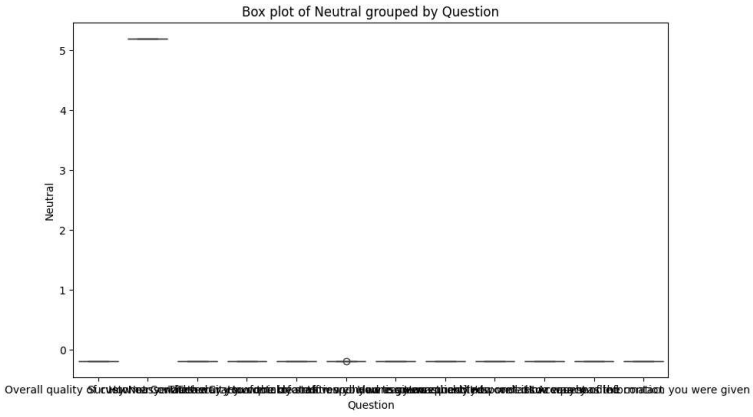

Box plot of Satisfied grouped by Question_Number
Satisfied
Question_Number

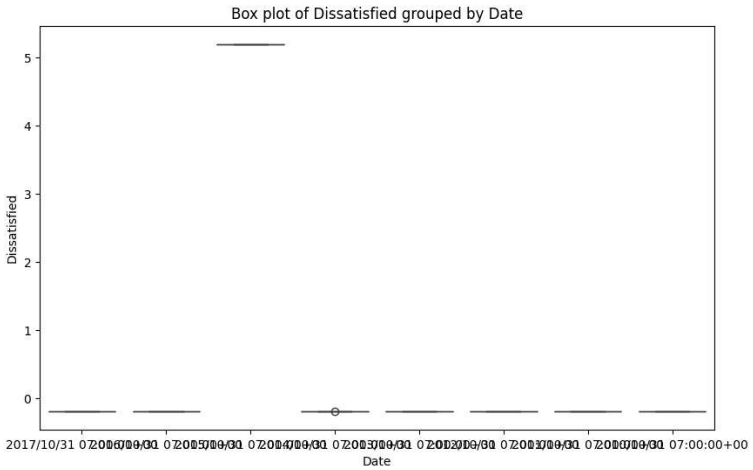

Box plot of Satisfied grouped by Question
Satisfied
Question

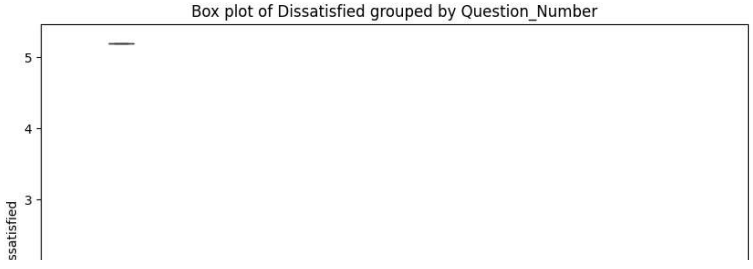

Box plot of Neutral grouped by Date

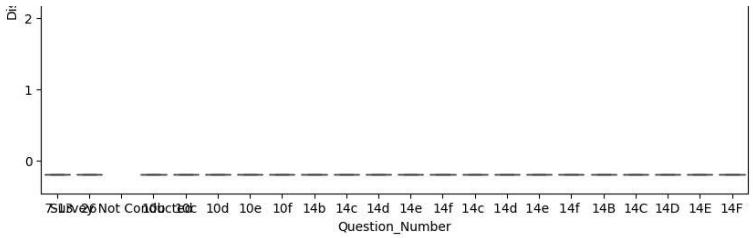

Neutral
Date

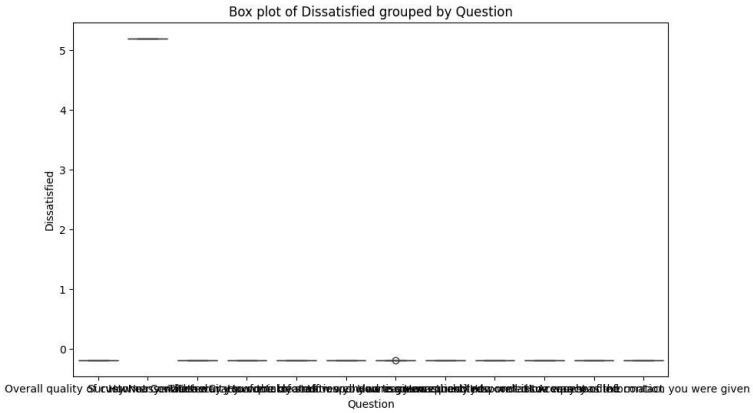

Box plot of Neutral grouped by Question_Number
Neutral
Question_Number

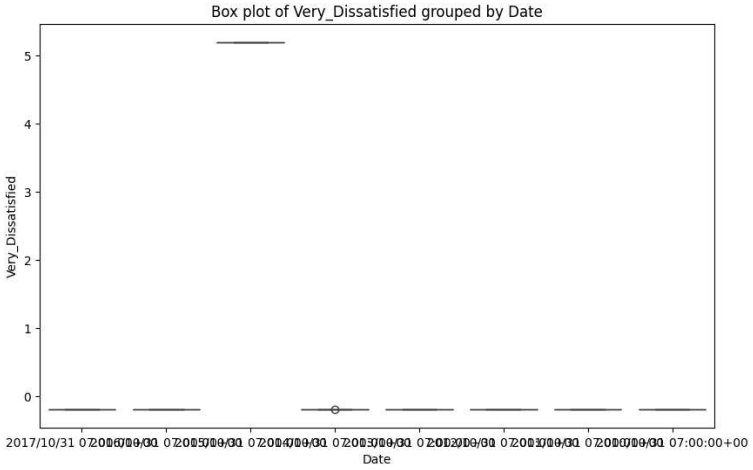

Box plot of Neutral grouped by Question
Neutral
Question

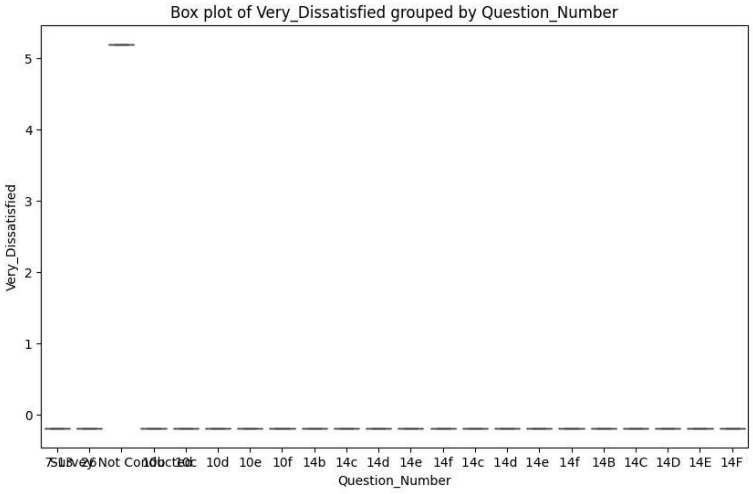

Box plot of Dissatisfied grouped by Date
Dissatisfied
Date

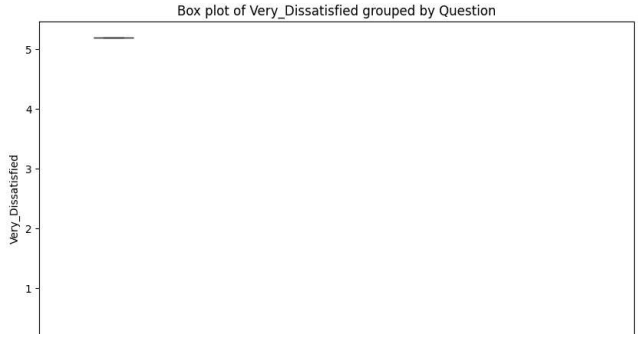

Box plot of Dissatisfied grouped by Question_Number

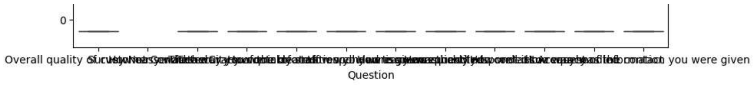

Di
2
1
0
Survey
Not Conducted
10d
10e
10f
14b
14c
14d
14e
14f
14c
14d
14e
14f
14B
14C
14D
14E
14F
Question_Number

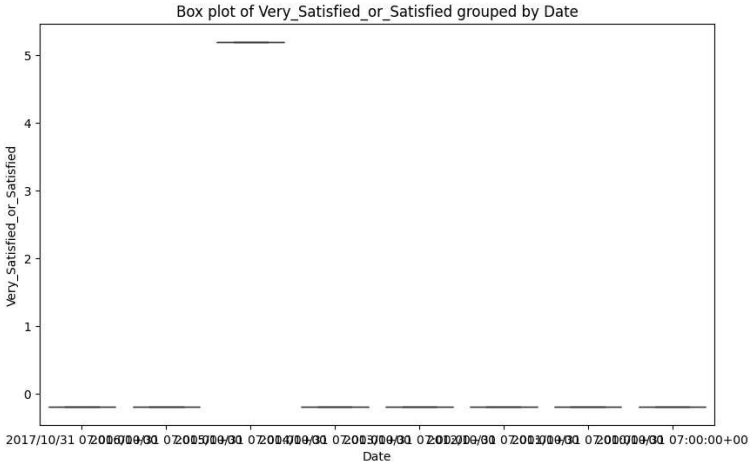

Box plot of Dissatisfied grouped by Question
Dissatisfied
5
4
3
2
1
0
Overall quality
you were given
Question

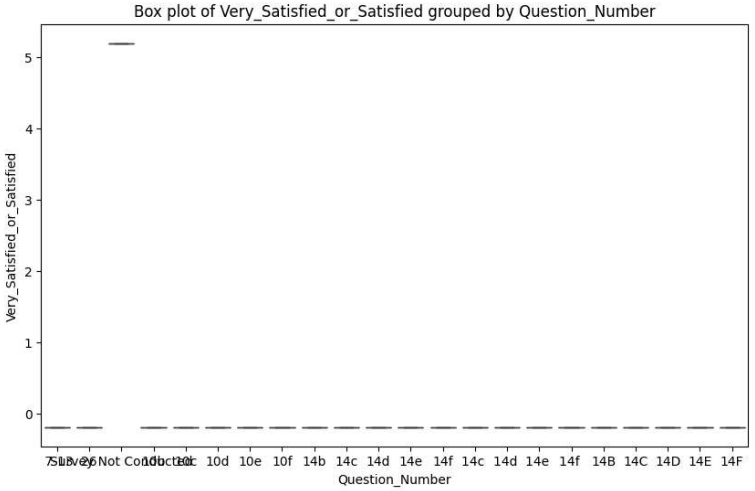

Box plot of Very_Dissatisfied grouped by Date
Very_Dissatisfied
5
4
3
2
1
0
2017/10/31 07:00:00+00
Date

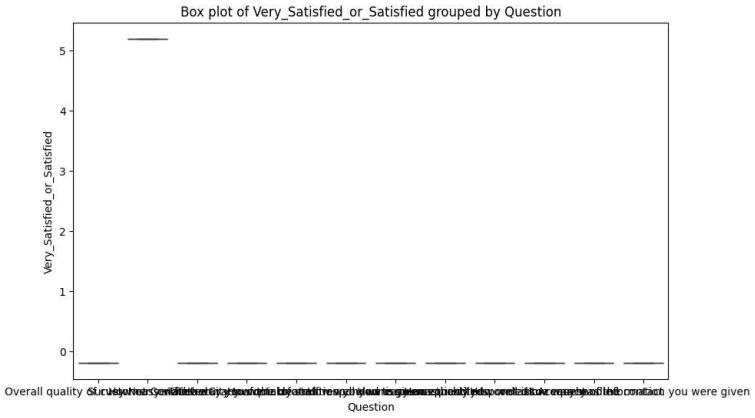

Box plot of Very_Dissatisfied grouped by Question_Number
Very_Dissatisfied
5
4
3
2
1
0
Survey
Not Conducted
10d
10e
10f
14b
14c
14d
14e
14f
14c
14d
14e
14f
14B
14C
14D
14E
14F
Question_Number

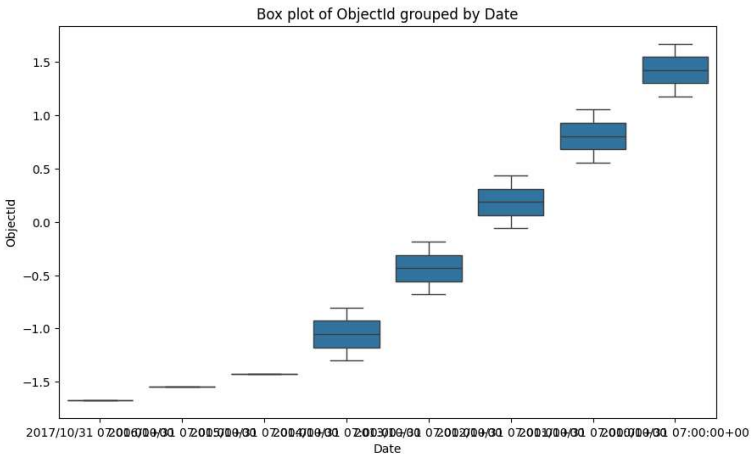

Box plot of Very_Dissatisfied grouped by Question
Very_Dissatisfied
5
4
3
2
1



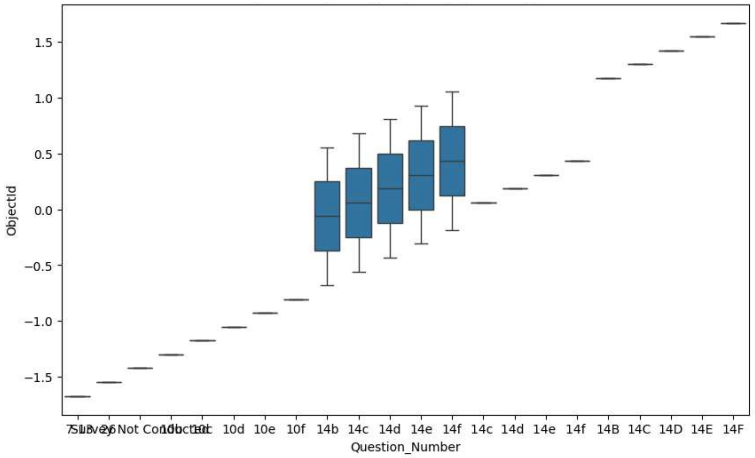


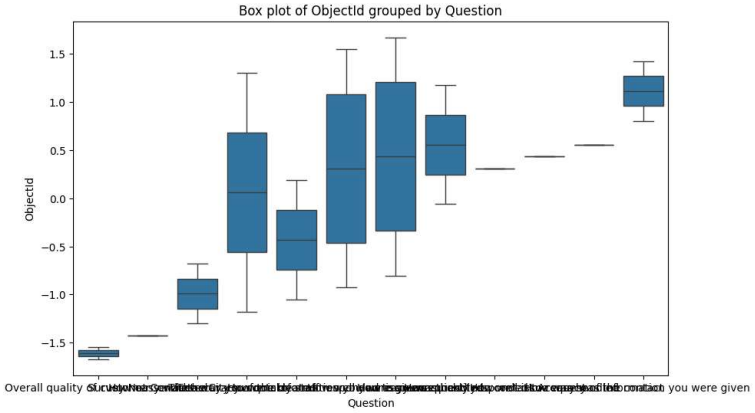






Box plot of ObjectId grouped by Question_Number


ObjectId
Question_Number
10d 10e 10f 14b 14c 14d 14e 14f 14c 14d 14e 14f 14B 14C 14D 14E 14F


Box plot of ObjectId grouped by Question
ObjectId
Question