

```
In [1]: import pandas as pd

# Load the dataset
file_path = '/Users/macbookpro/Desktop/Data for K-Means Clustering.xlsx'
data = pd.read_excel(file_path)

# Display the first few rows of the dataset
data.head()
```

Out[1]:

	UNIQUE ID	Average Quarter	Average Excess	Average Revenue
0	40	2.58	-2233.544367	180698.335602
1	41	2.58	-3990.718000	327293.764314
2	44	2.58	-1433.183006	142466.330093
3	45	2.58	-618.200356	116086.854261
4	46	2.58	-4155.915719	319407.803656

```
In [2]: import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
```

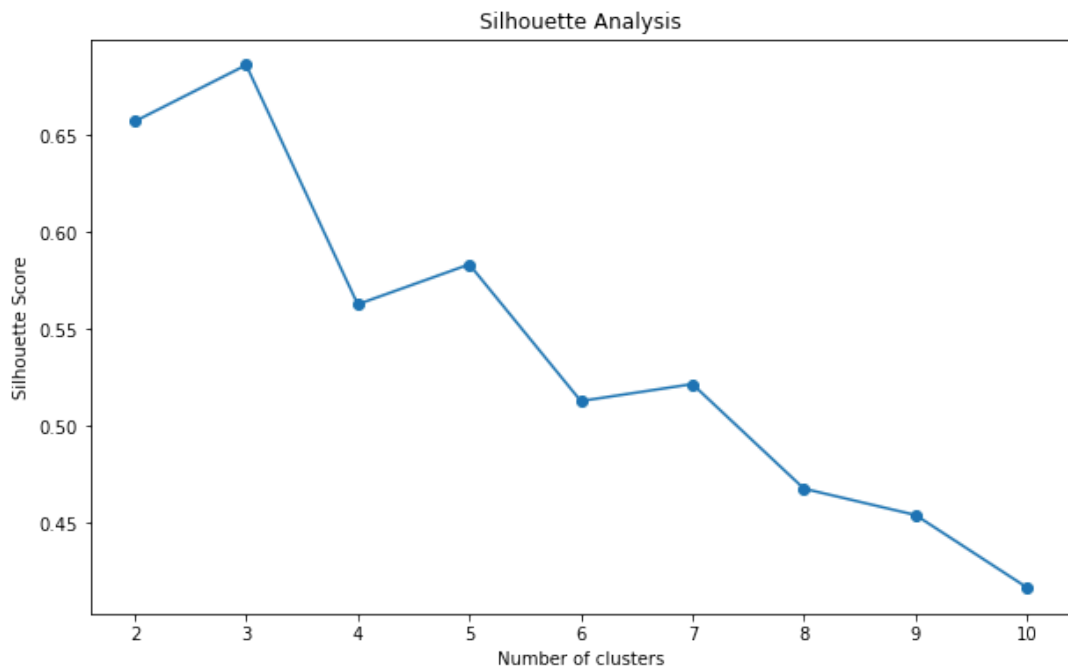
```
In [3]: # Extract the relevant columns for clustering
X = data[['Average Quarter', 'Average Excess', 'Average Revenue']]
```

```
In [4]: # Standardize the data
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

```
In [5]: from sklearn.metrics import silhouette_score

# Calculate silhouette scores for different values of K
silhouette_scores = []
for i in range(2, 11):
    kmeans = KMeans(n_clusters=i, random_state=42)
    labels = kmeans.fit_predict(X_scaled)
    silhouette_scores.append(silhouette_score(X_scaled, labels))

# Plot the silhouette scores to find the optimal K
plt.figure(figsize=(10, 6))
plt.plot(range(2, 11), silhouette_scores, marker='o')
plt.title('Silhouette Analysis')
plt.xlabel('Number of clusters')
plt.ylabel('Silhouette Score')
plt.show()
```



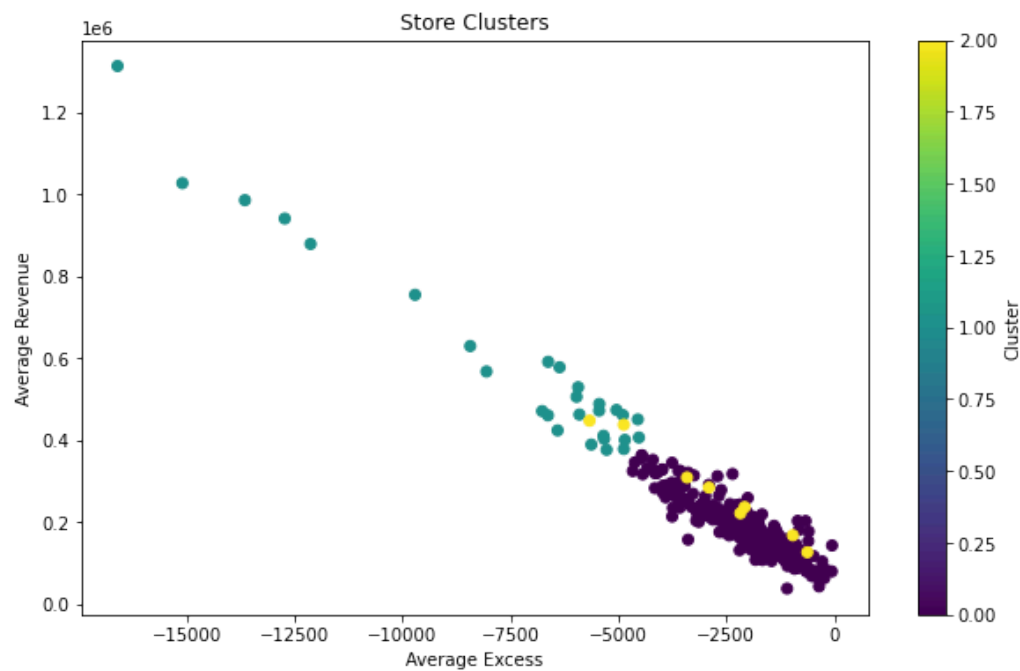
```
In [6]: # Apply K-Means with 5 clusters
kmeans = KMeans(n_clusters=3, random_state=42)
data['Cluster'] = kmeans.fit_predict(X_scaled)
```

```
In [7]: import os
```

```
In [8]: # Save the Excel File with the added column indicating which cluster a store is assigned to
export_dataset_path = os.path.join('/Users/macbookpro/Desktop', 'Clustered_Store_Data.xlsx')
export_plot_path = os.path.join('/Users/macbookpro/Desktop', 'Store_Clusters.pdf')

# Save the updated dataset with cluster assignments
data.to_excel(export_dataset_path, index=False)
```

```
In [9]: # Visualize the clusters (Note: Does not give full information on the Average Quarter)
plt.figure(figsize=(10, 6))
plt.scatter(data['Average Excess'], data['Average Revenue'], c=data['Cluster'], cmap=cm.viridis)
plt.xlabel('Average Excess')
plt.ylabel('Average Revenue')
plt.title('Store Clusters')
plt.colorbar(label='Cluster')
plt.savefig(export_plot_path, format='pdf')
plt.show()
```



```
In [10]: # Display the first few rows with cluster assignments
data.head()
```

Out[10]:

	UNIQUE ID	Average Quarter	Average Excess	Average Revenue	Cluster
0	40	2.58	-2233.544367	180698.335602	0
1	41	2.58	-3990.718000	327293.764314	0
2	44	2.58	-1433.183006	142466.330093	0
3	45	2.58	-618.200356	116086.854261	0
4	46	2.58	-4155.915719	319407.803656	0