

# **Customer Segmentation with K-Nearest Neighbors**

SELVARAJ.X(2577127)

31-OCT-2021

# Introduction

- ❖ Our goal is to effectively segment customers based on their purchase behavior using the K-Nearest Neighbors (KNN) algorithm.
- ❖ Customer segmentation is crucial in the retail industry for several reasons.
- ❖ It allows for personalized marketing, leading to improved customer satisfaction and increased sales.
- ❖ Customer segmentation is a fundamental strategy for retail success.
- ❖ KNN is a powerful tool that, when used appropriately, can help retailers better understand and engage with their customers.

# Data Exploration

- ❖ The dataset contains various features, including purchase history, age, and income.
- ❖ Our target variable will be customer segments.
- ❖ Importance of Customer Segmentation
  - Customer segmentation is crucial in the retail industry for several reasons.
  - It allows for personalized marketing, leading to improved customer satisfaction and increased sales.

```
data = pd.read_csv('data/customer_data.csv')
data.head()
```

	name	age	gender	education	income	country	purchase_frequency	spending
0	Teresa Williams MD	42	Female	High School	53936	Slovenia	0.9	13227.120
1	Christine Myers	49	Female	Master	82468	Aruba	0.6	12674.040
2	Dwayne Moreno	55	Male	Bachelor	56941	Cyprus	0.3	5354.115
3	Amy Norton	24	Female	Bachelor	60651	Palau	0.2	2606.510
4	Tonya Adams	64	Male	Master	81884	Zambia	0.9	18984.780

```
x = data[['age', 'income', 'purchase_frequency']]
```

# Data Preprocessing

- ❖ Before applying KNN, we need to preprocess the data.
- ❖ Scaling: We've normalized numerical features to ensure they have the same scale.
- ❖ Handling Missing Values: Missing data was addressed to prevent bias.
- ❖ Encoding Categorical Variables: We transformed categorical data into numerical form for analysis.

```
# create segments for each customer...

from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

# Scale the features
scaler = StandardScaler()
x_scaled = scaler.fit_transform(x)

# Create an instance of the KMeans clustering algorithm
kmeans = KMeans(n_clusters=3, random_state=0) # Choosing 3 clusters for de

# Fit the KMeans model
kmeans.fit(x_scaled)

# Get the cluster labels for each customer
data['segment'] = kmeans.labels_
```

# Implementing KNN

- ❖ The K-Nearest Neighbors (KNN) algorithm is a powerful tool for customer segmentation.
- ❖ We selected an appropriate number of neighbors (K) to enhance the accuracy of our model.

```
knn_model = KNeighborsClassifier(n_neighbors=3)

print('<-----KNN Classifier model----->')
knn_model.fit(xtrain,ytrain)
ypred = knn_model.predict(xtest)

# performance metric

print("Accuracy score :\t",accuracy_score(ytest,ypred))
print()
print('Confusion Martix :\n',confusion_matrix(ytest,ypred))
print()
print('classification Report :\n', classification_report(ytest,ypred))
```

# Model Training

- ❖ To create customer segments, we trained the KNN model on our preprocessed dataset.
- ❖ Customers are grouped based on purchase behavior, age, and income.
- ❖ This provides valuable insights for targeted marketing.
- ❖ The distance metric we used to find nearest neighbors is crucial, as it determines the similarity between customers.



# Hyperparameter Tuning

- ❖ The hyperparameter K plays a vital role in KNN.
- ❖ We used strategies to select the optimal value of K, and it significantly impacted our segmentation results.

```
# Hyperparameter Tuning

knn_model = KNeighborsClassifier(n_neighbors=17)

print('<-----KNN Classifier model after tuning----->')
knn_model.fit(xtrain,ytrain)
ypred = knn_model.predict(xtest)

print("Accuracy score :\t",accuracy_score(ytest,ypred))
print()
print('Confusion Martix :\n',confusion_matrix(ytest,ypred))
print()
print('classification Report :\n', classification_report(ytest,ypred))
```

# Model Evaluation with Hyperparameter Tuning

<-----KNN Classifier model-----

Accuracy score : 0.976

Confusion Martix :

```
[[101  2  1]
 [  1 72  0]
 [  2  0 71]]
```

classification Report :

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

0	0.97	0.97	0.97	104
1	0.97	0.99	0.98	73
2	0.99	0.97	0.98	73

accuracy			0.98	250
macro avg	0.98	0.98	0.98	250
weighted avg	0.98	0.98	0.98	250

<-----KNN Classifier model after tuning-----

Accuracy score : 0.98

Confusion Martix :

```
[[101  1  2]
 [  0 73  0]
 [  2  0 71]]
```

classification Report :

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

0	0.98	0.97	0.98	104
1	0.99	1.00	0.99	73
2	0.97	0.97	0.97	73

accuracy			0.98	250
macro avg	0.98	0.98	0.98	250
weighted avg	0.98	0.98	0.98	250

# Real-World Application

- ❖ Customer segmentation has practical applications in the retail industry.
- ❖ It can lead to improved customer engagement, personalized marketing campaigns, and ultimately increased sales.

## Model Limitations

- ❖ While KNN is a powerful tool, it has limitations.
- ❖ It may not perform well in scenarios with high-dimensionality or imbalanced data.

# Conclusion

- ❖ Customer segmentation is a fundamental strategy for retail success.
- ❖ The role of KNN in data-driven marketing cannot be underestimated.
- ❖ KNN is a powerful tool that, when used appropriately, can help retailers better understand and engage with their customers.