Customer Segmentation Using ML

Divya Pachbhai

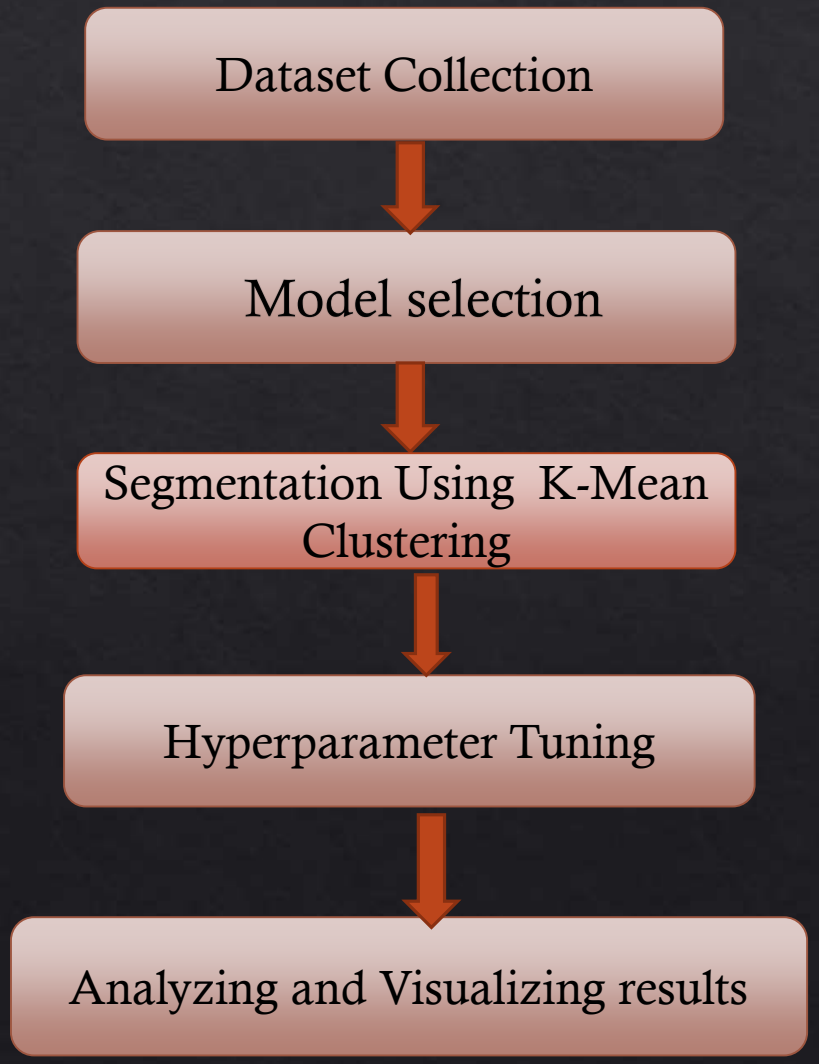
INTRODUCTION

- Everything can be customized these days. There isn't a method that works for everyone. However, this is actually a fantastic thing for business. It opens up a lot of room for constructive competition and presents businesses with chances to think outside the box when it comes to attracting and retaining customers.
- One of the fundamental steps towards better personalization is customer segmentation. This is where personalization starts, and proper segmentation will help you make decisions regarding new features, new products, pricing, marketing strategies.

Problem Statement

- Now a days we can see that companies spend a lot on marketing but still cannot reach target customers. To solve this problem we can group customers on the basis of demographical, behavioral, geographical or psychological differences this is called as customer segmentation.
- Segmentation helps companies to reach targeted customers with minimal marketing, launch a product which fulfill customer demand with proper pricing, and helps in developing proper roadmap.

Methodology



Dataset

Dataset consists 200 rows and five hyperparameters they are:

- Customer ID
- Gender
- Age
- Annual Income
- Spending score

```
In [3]: customer.head()
```

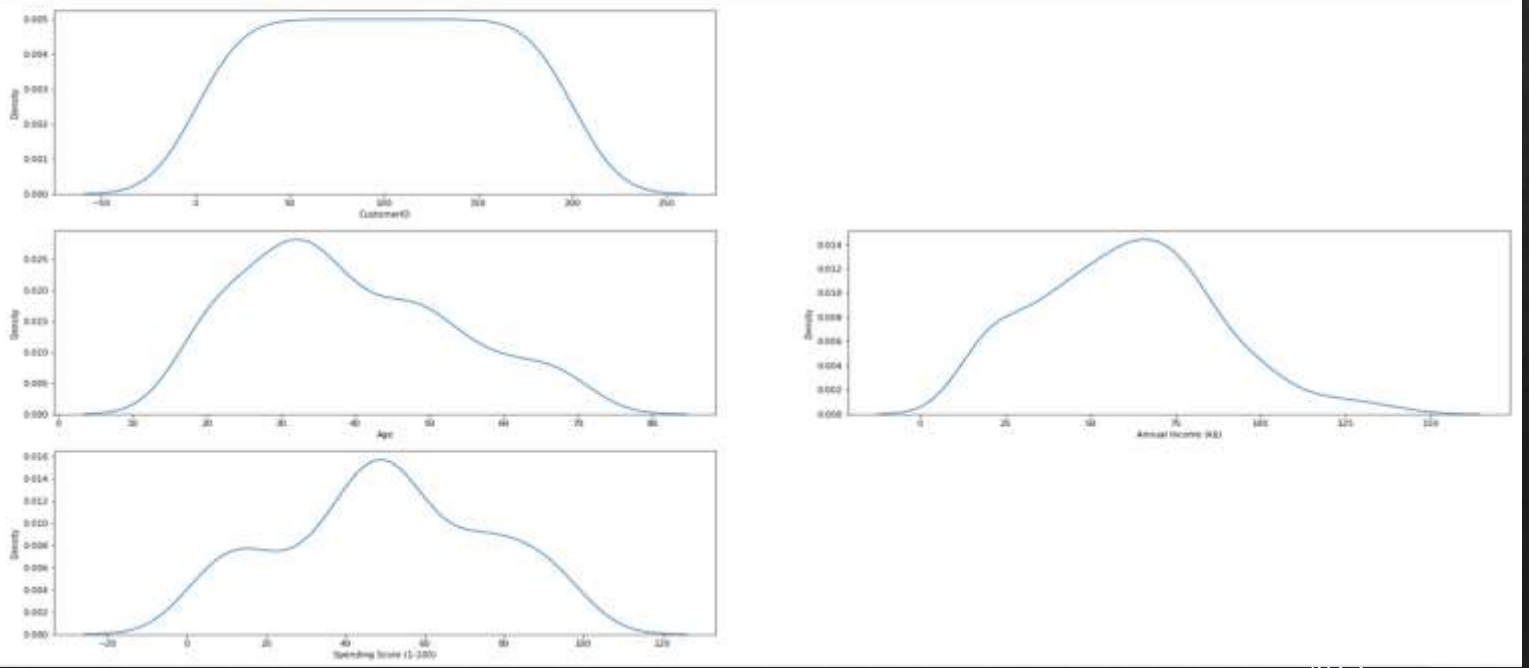
Out[3]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

Dataset

Density plots :

```
plt.figure(figsize=(30,45))
for i,col in enumerate(customer.columns):
    if customer[col].dtype != 'object':
        ax = plt.subplot(9, 2, i+1)
        sns.kdeplot(customer[col], ax=ax)
        plt.xlabel(col)

plt.show()
```



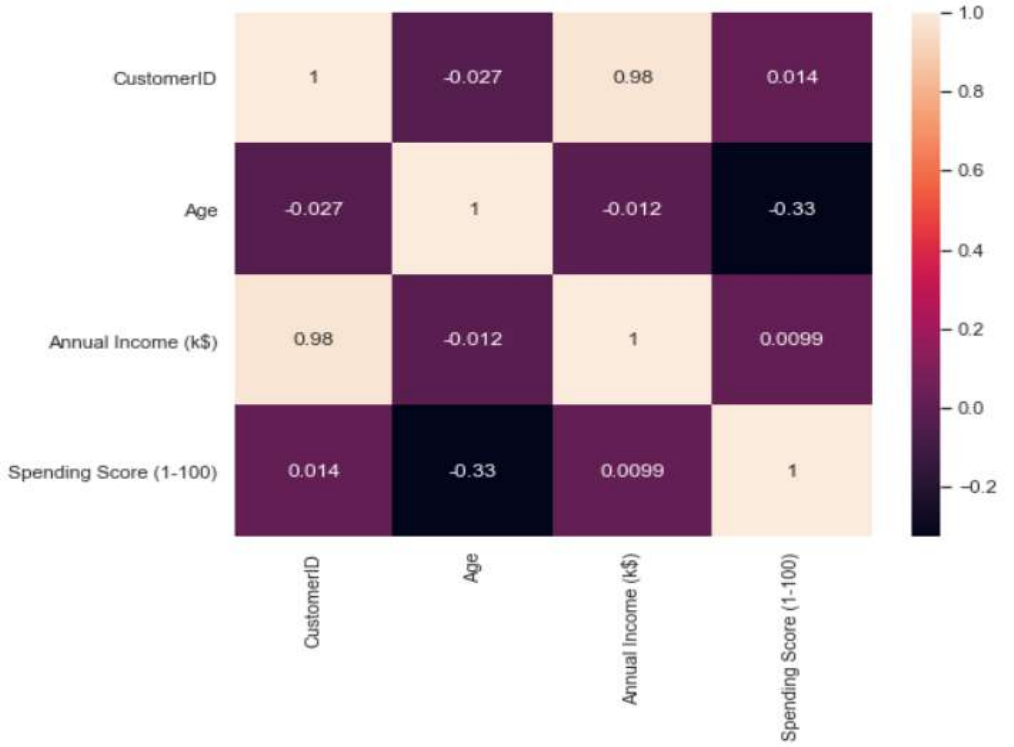
Dataset

Heatmap:

- This heatmap shows that there is strong negative correlation between age of customer and spending score.
- We can also see that there is positive correlation between income and Spending score.
- So we will take this three parameters to group the customers.

Plotting heatmap to understand correlation between attributes

```
In [41]: plt.figure(figsize=(8,6))
         sns.heatmap(customer.corr(), annot=True)
         plt.show()
```



K-Mean Clustering

Unsupervised learning algorithm K-Means Clustering divides the unlabeled dataset into several clusters on the basis of similar characteristics. Here, K specifies how many pre-defined clusters must be produced as part of the process. Advantages of K-Mean clustering are:

- ✓ Implementation is easy
- ✓ handles huge data collections at large scale
- ✓ provides for convergence
- ✓ able to reset the centroids' locations
- ✓ takes fresh examples in stride
- ✓ generalizes to a variety of cluster types, including elliptical clusters

Hyperparameter Tuning

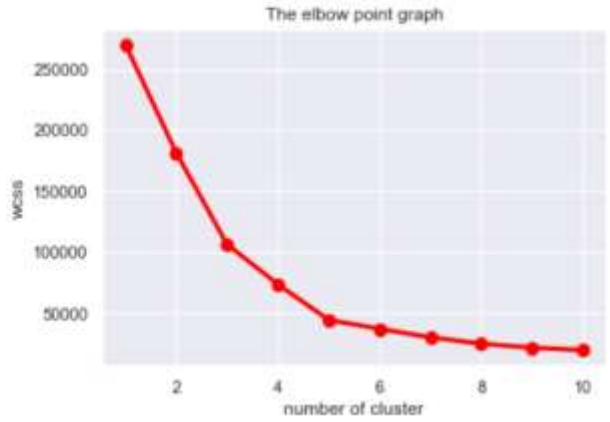
- In the Elbow method, we are varying the number of clusters (K) from 1 – 10. For each value of K, we are calculating WCSS (Within-Cluster Sum of Square) value.
- WCSS is the sum of squared distance between each point and the centroid in a cluster.
- When we plot the WCSS with the K value, the plot looks like an Elbow.
- As the number of clusters increases, the WCSS value will start to decrease.
- When we analyze the graph we can see that the graph will rapidly change at a point and thus creating an elbow shape.
- From this point, the graph starts to move almost parallel to the X-axis. The K value corresponding to this point is the optimal K value or an optimal number of clusters.

Taking Two parameters (Income and Spending Score)

Elbow point graph

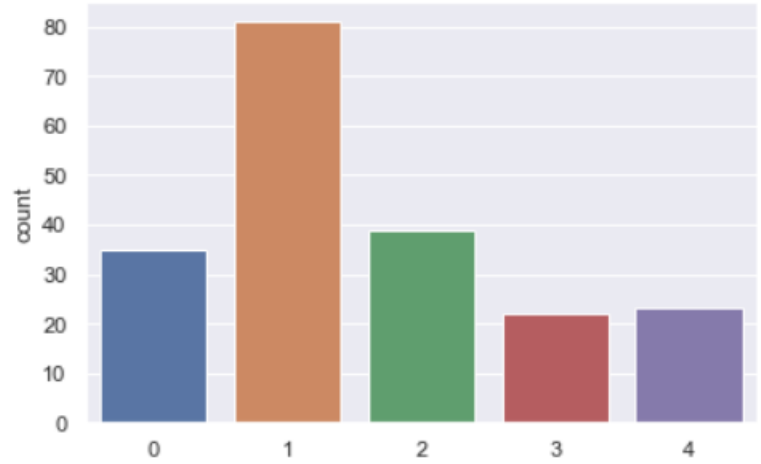
Plotting elbow graph

```
In [12]: sns.set()
         plt.plot(range(1,11),wcss,linewidth=3,markersize=8,marker='o',color='red')
         plt.title("The elbow point graph")
         plt.xlabel('number of cluster')
         plt.ylabel('wcss')
         plt.show()
```

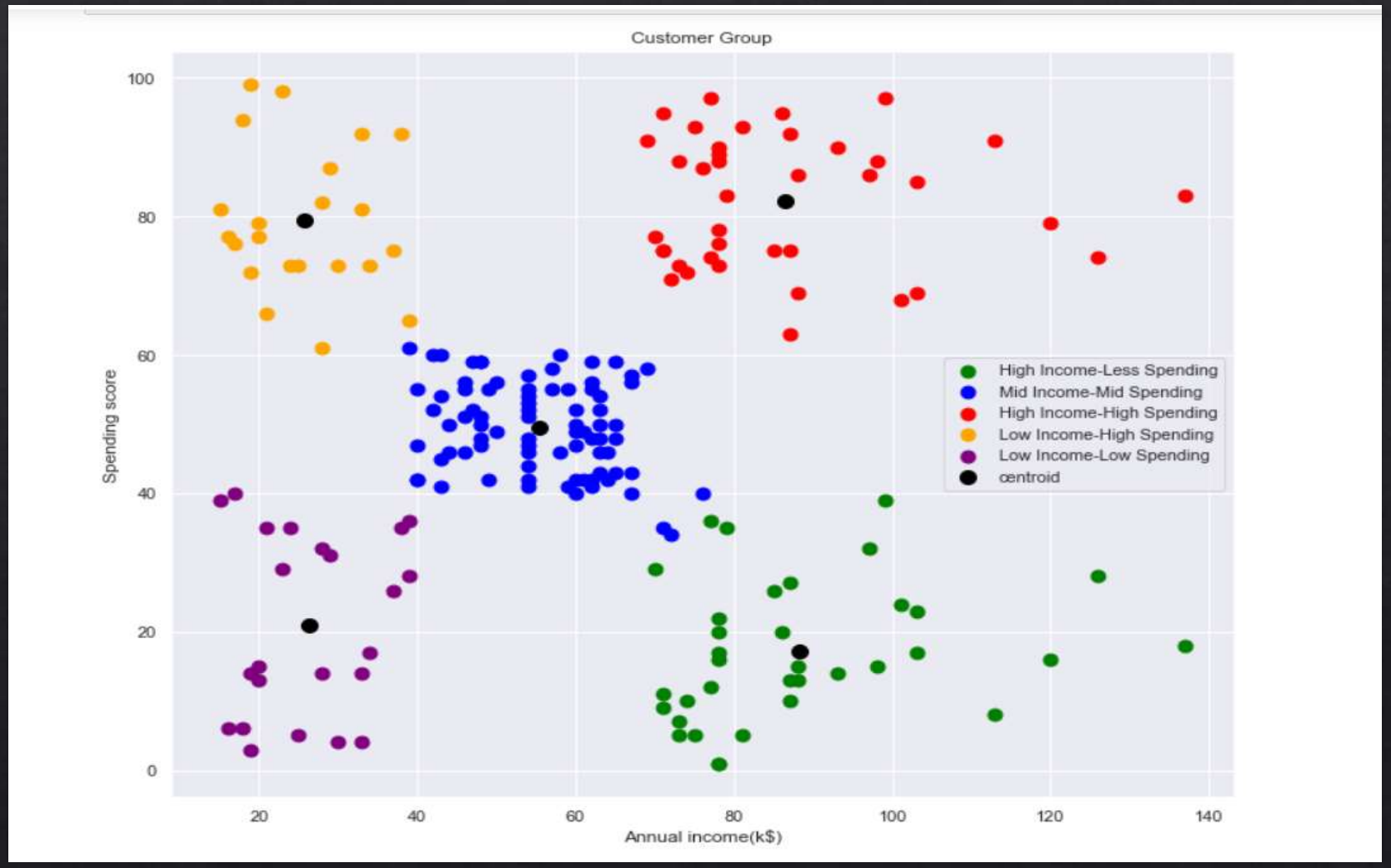


Count of customers in each segment

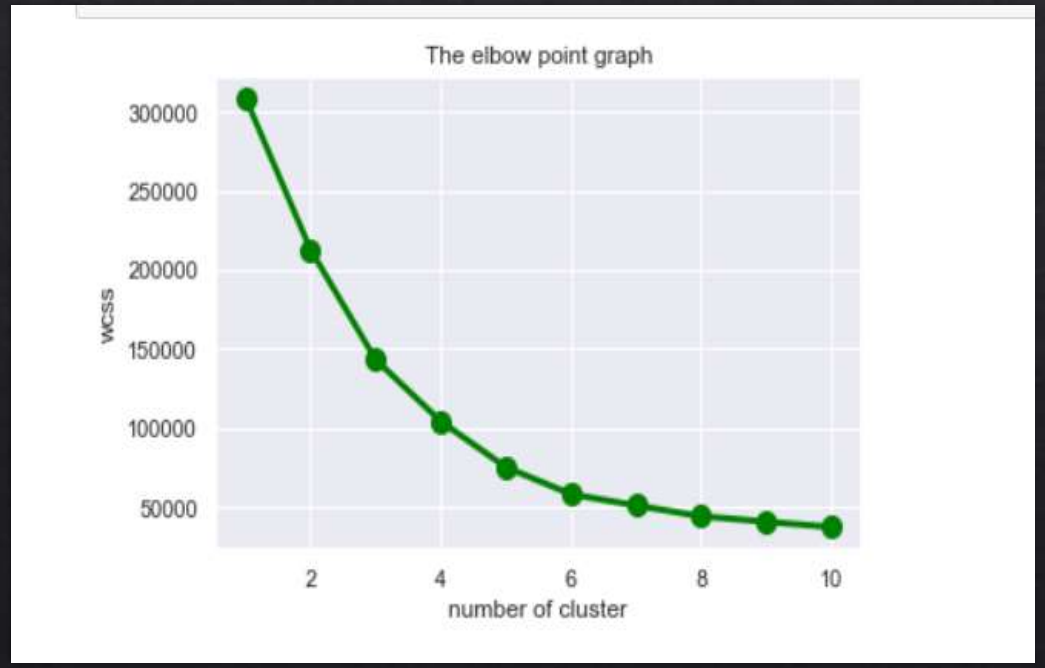
```
In [15]: # counting number of customers in each segment
         sns.countplot(labels)

Out[15]: <AxesSubplot:ylabel='count'>
```



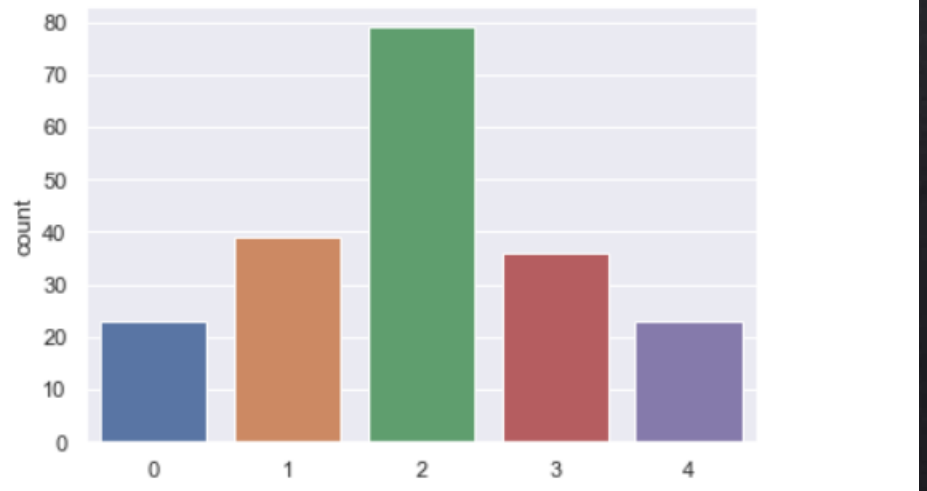
Visualization



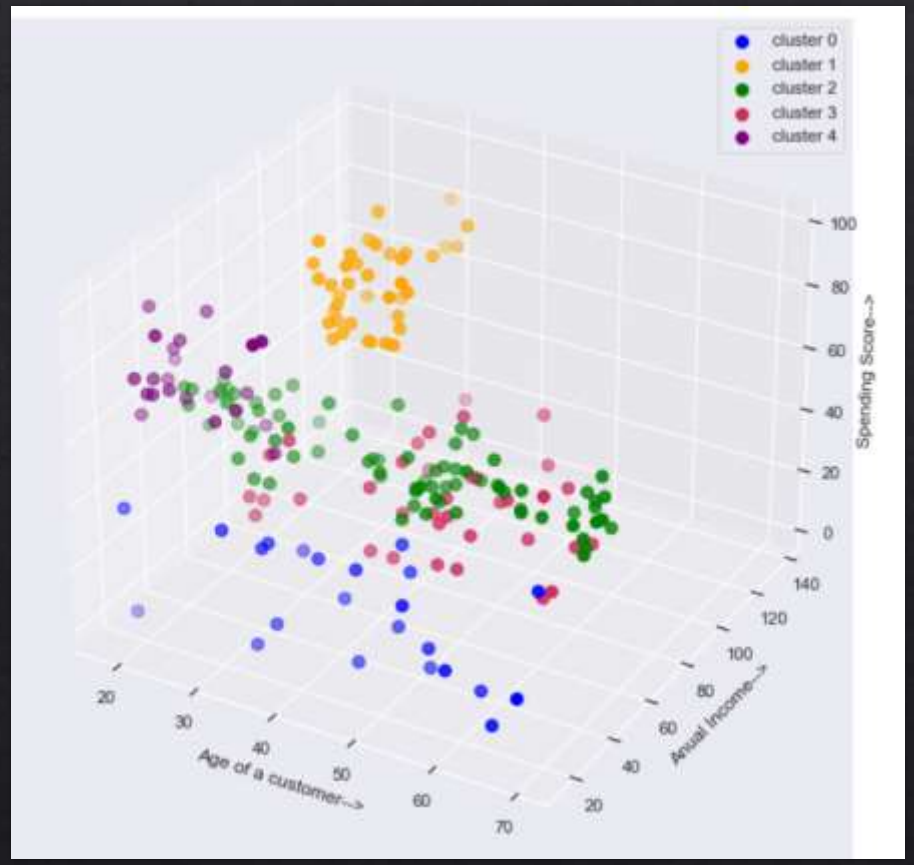
Taking 3 parameters (Age, Income and Spending Score)

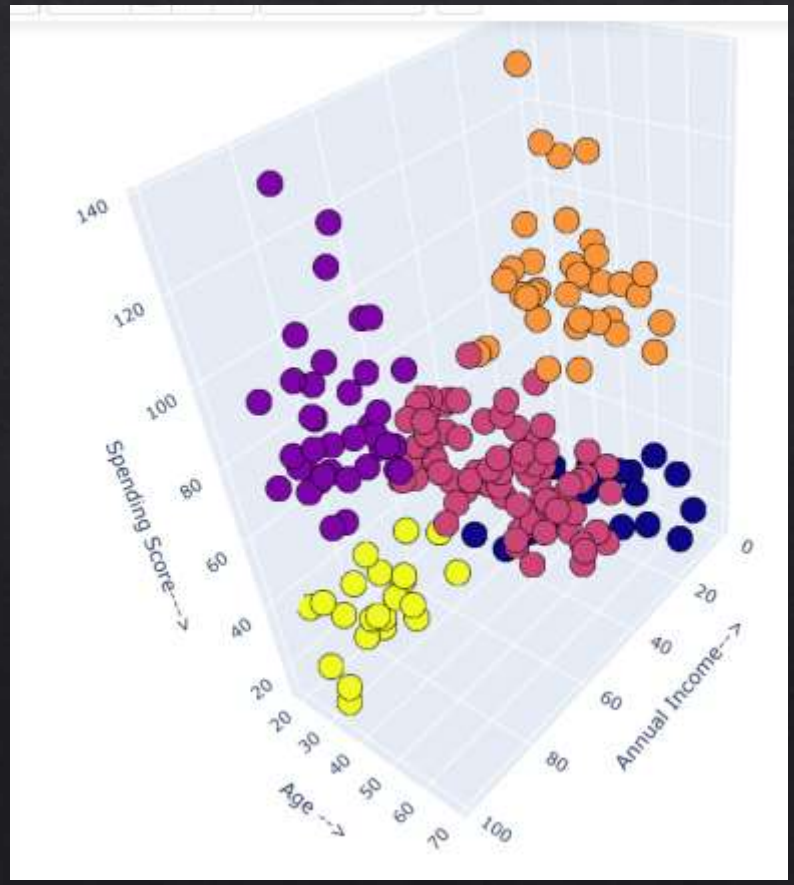



```
In [27]: sns.countplot(y_clusters)
Out[27]: <AxesSubplot:ylabel='count'>
```



Visualization





Results

Sr No.	Parameters	accuracy
1.	Income and Spending Score	93.34%
2.	Income, Age, and Spending Score	96.67%

References

Dataset: https://www.kaggle.com/datasets/kandij/mall-customers

Literature: Customer Segmentation Using Machine Learning by prof. Nikhil Patankar Soham Dixit, Akshay Bhamare , Ashutosh Darpel and Ritik Raina Customer_Segmentation_Using_Machine_Learning

THANK YOU