CPSC 392: ASSIGNMENT 3

DIYA KULKARNI

BEHAVIORAL CLUSTERING MODEL

INTRODUCTION

In the behavioral clustering model, I am utilizing the dataset “behavioral.csv”, which contains information about the media company customers' behavior on the site. The variables I have available are

- id: customer id
- gender: self-disclosed gender identity, male, female, nonbinary or other
- age: age in years
- current_income: self-reported current annual income in thousands
- time_spent_browsing: average number of minutes spent browsing website per month
- prop_ad_clicks: proportion of website ads that they click on (between 0 and 1)
- longest_read_time: longest time spent consecutively on website in minutes
- length_of_subscription: number of days subscribed to the magazine
- monthly_visits: average number of visits to the site per month

All of the variables are continuous/interval variables except for “gender”, which is categorical. By utilizing clustering algorithms, I will be able to create a model to help the media company understand their customers’ needs better, and put them into useful groups of similar clusters. This model will be impactful as it will enable the media company to adjust their marketing tactics according to the appropriate customer segments.

METHODS

After loading the data and all the necessary imports and checking for missing values, I created a list of predictors, omitting “gender” because it was a categorical variable and “id” because it was stated to remove it. I Z-scored all of the predictors before proceeding with creating a Gaussian Mixture Model to use as my clustering algorithm for my model.

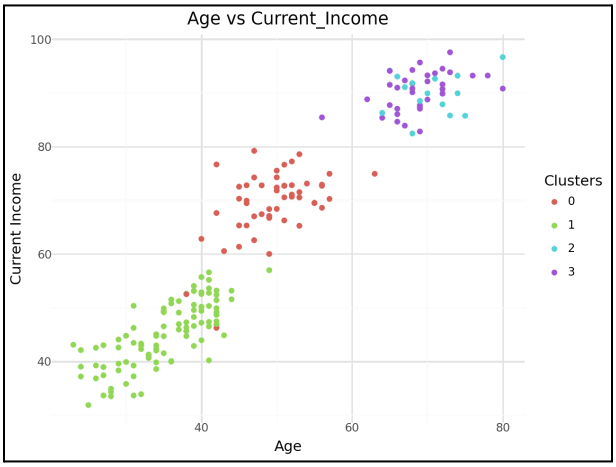
Pros and Cons

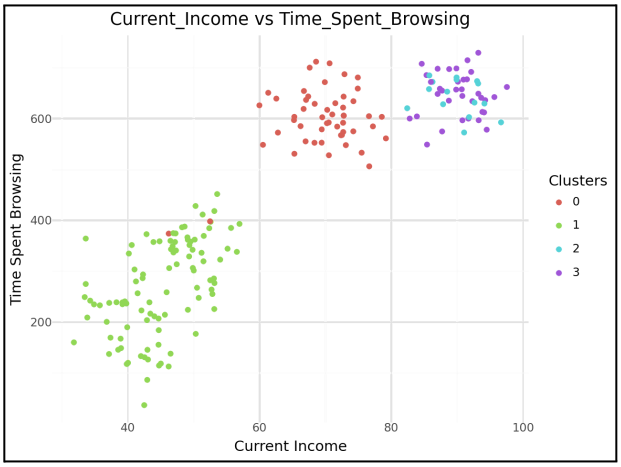
	Pros	**Cons**
K-Means data - numerical, well-defined spherical clusters, clusters are relatively well-separated	- Easy to understand - Adapts to new data - Ability to guarantee convergence.	- Need to pre-determine clusters. - Sensitivity to outliers - Potential of getting different results
Gaussian Mixture Models data - exhibits multiple, potentially overlapping	- Soft clustering method; data doesn’t have to belong to only one cluster	- Assumption of Gaussian (continuous) distribution - Risk of overfitting

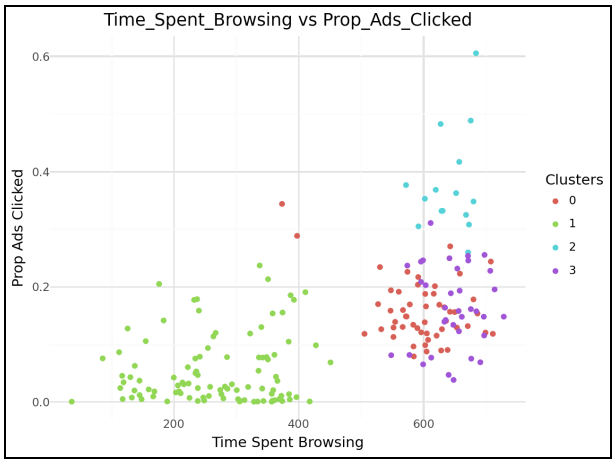
distributions	- Allows variance to be different	
DBSCAN data - irregular shapes, varying densities, and significant noise	- Makes no assumption about shape of clusters - Does not require every data point to belong to a cluster - Does not predefine the # of clusters - they are calculated along the way - Focuses on density > cohesion → it doesn't matter if data points are close within the cluster. It matters more if the clusters are close to each other	- Won't work well with overlapping clusters - Less effective with high dimensional data - Suboptimal when clusters have different densities
Hierarchical Clustering data - natural hierarchical structure	- Flexible # of clusters - Hierarchical relationship - Flexibility with linkage criteria	- Very slow - Clusters cannot unmerge once they are merged.

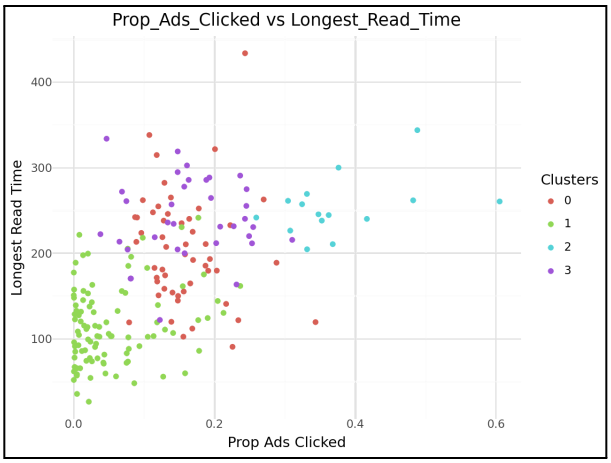
I chose the Gaussian Mixture Model (clustering algorithm) because it is able to group data into different clusters and works well on data with multiple, potentially overlapping distributions. I also found the approach to GMM to be one I was familiar with. I believe that the data in this case does not have a natural hierarchical relationship, and also has the potential to overlap (for example, "time_spent_browsing" and "longest_read_time"). Due to this, DBSCAN and Hierarchical Clustering would not be efficient algorithms. Additionally, the uncertainty with the K-Means clustering algorithm made me believe that GMM was the best fit for this dataset.

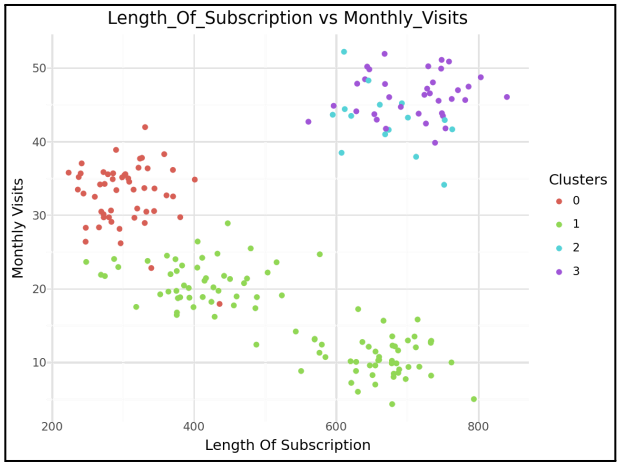
Scatterplots for Features (while the code contains all of the combinations of features, I have only included a few here for convenience purposes):











Chosen Model Details

When creating my Gaussian Mixture model, I used Bayesian Information Criterion to determine the number of clusters to fit in my model. The BIC measures how well fit your model is, where lower values of BIC are better. When plotting the different BIC values for different “K”s, where K is the number of parameters in the model, I found that a K value of 4 equated to the lowest value of BIC. Upon determining this, I created and fit the Gaussian Mixture Model with 4 clusters.

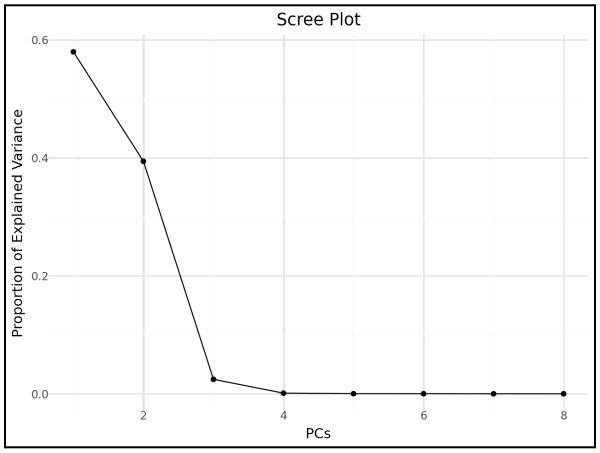
I decided to fit a model to investigate the impact of “Current Income” on “Length of Subscription”. I wanted to dive deeper into the relationship between income and media subscription services.

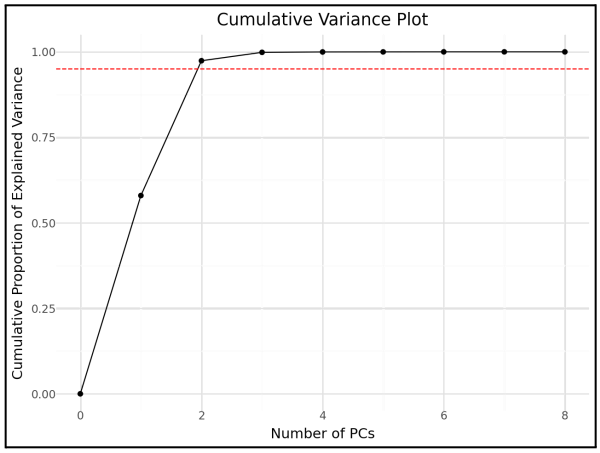
RESULTS

Upon visual inspection, the plot indicates that the clusters are fairly well-separated, with minimal overlap between the groups. This is a good sign that the GMM has performed well in differentiating customer segments within the data.

When plotting the Principal Component Analysis, it takes 3 PCs for the cumulative proportion of explained variance to reach 1.00, i.e., for all of the variance to be explained. Considering GMM benefits

from dimensionality reduction, the low number of PCs indicates that the clustering is effective and computationally efficient. The scree and cumulative variance plots are displayed below:





Following the PCA, I created a summary table to better understand the types of customers within each clusters based on the different predictors. This table includes key statistical measures for each predictor, such as the mean, standard deviation, median, minimum, and maximum, providing deeper insight into the variation and central tendencies within each cluster. (Below is a more detailed breakdown; the code includes a more condensed version).

Mean:

	age	current_income	time_spent_browsing	prop_ads_clicked	longest_read_time	length_of_subscription	monthly_visits
	mean	mean	mean	mean	mean	mean	mean
clusters							
0	49.500000	69.437692	596.525800	0.158204	204.024657	302.904621	32.767196
1	35.275510	45.217449	268.743973	0.054014	115.500054	535.630950	15.784370
2	70.400000	89.937333	642.464733	0.377371	253.531493	671.403067	42.866101
3	68.942857	90.240857	647.054062	0.164419	240.256097	710.267499	46.230638

Standard Deviation:

	age	current_income	time_spent_browsing	prop_ads_clicked	longest_read_time	length_of_subscription	monthly_visits
	std	std	std	std	std	std	std
clusters							
0	4.816231	6.044273	63.769790	0.053902	65.444601	45.167411	4.228409
1	5.707258	6.019169	92.656064	0.059078	46.297467	143.424607	5.921563
2	4.339190	3.839007	35.052557	0.088517	34.251207	56.670825	4.282570
3	4.451985	3.580378	43.259062	0.070538	46.719921	62.330689	3.027729

Median:

clusters	age median	current_income median	time_spent_browsing median	prop_ads_clicked median	longest_read_time median	length_of_subscription median	monthly_visits median
0	50.0	70.405	602.509408	0.148271	195.806554	297.708877	33.538368
1	36.0	45.350	270.975815	0.029462	107.493287	546.907276	15.999800
2	70.0	89.930	652.357826	0.352661	245.354792	668.950295	43.238983
3	69.0	90.800	648.361853	0.157731	234.173768	728.063700	46.042560

Minimum:

clusters	age min	current_income min	time_spent_browsing min	prop_ads_clicked min	longest_read_time min	length_of_subscription min	monthly_visits min
0	38	46.24	373.624925	0.078870	90.475695	223.423411	17.927277
1	23	31.85	36.813198	0.000175	26.289526	248.108717	4.316451
2	64	82.43	572.157802	0.259527	204.450884	595.141761	34.111695
3	56	82.81	548.440006	0.038046	121.854154	560.927796	39.824416

Maximum:

clusters	age max	current_income max	time_spent_browsing max	prop_ads_clicked max	longest_read_time max	length_of_subscription max	monthly_visits max
0	63	79.20	711.215159	0.343746	433.521888	435.900052	41.948066
1	49	56.98	451.260283	0.236787	241.303454	793.832506	28.885762
2	80	96.67	684.254704	0.605402	343.640113	763.412996	52.186826
3	80	97.56	728.732816	0.310586	333.643622	840.325948	51.903795

From the data gathered in the summary tables, we can conclude the following about each cluster:

Cluster 0 - The average age in this cluster ranges from 45 to 53 years, with a self reported annual income of 39K to 51K. Cluster members spend 530 to 650 minutes browsing per month with their longest read time ranging from 140 to 270 minutes. On average, they click about 16% of ads and have been subscribed to the magazine for 260 to 340 days. They make an average of 33 monthly visits to the site per month.

Considering the longest read time within this cluster, it would be beneficial for the company to focus on in-depth topics. The age range indicates that this cluster group highly appreciates investigating topics for a longer period of time.

Cluster 1 - The average age in this cluster ranges from 30 to 40 years, with a self reported income of 63K to 75K. Cluster members spend 170 to 360 minutes browsing per month with their longest read time ranging from 70 to 160 minutes. On average, they click about 5% of ads and have been subscribed to the magazine for 390 to 680 days. They make an average of 16 monthly visits to the site per month.

Considering the younger age group as well as the low ad click rate, the company should focus on shorter & more relevant ads to what this cluster is interested in (namely career development, work life balance, etc.)

Cluster 2 - The average age in this cluster ranges from 66 to 74 years, with a self reported income of 86K to 92K. Cluster members spend 610 to 680 minutes browsing per month with their longest read time ranging from 220 to 280 minutes. On average, they click about 38% of ads and have been subscribed to the magazine for 615 to 730 days. They make an average of 43 monthly visits to the site per month.

Considering the high proportion of ads clicked, the company would benefit from creating ads catered to the cluster's tastes. Additionally, the older age group could appreciate being provided with more accessibility features.

Cluster 3 - The average age in this cluster ranges from 65 to 73 years, with a self reported income of 87K to 93K. Cluster members spend 600 to 690 minutes browsing per month with their longest read time ranging from 200 to 280 minutes. On average, they click about 17% of ads and have been subscribed to the magazine for 650 to 770 days. They make an average of 46 monthly visits to the site per month.

This cluster is very similar to the last in terms of the age group, income, and time on the site. However, they have roughly half the proportion of ads clicked. The company would do well by investigating the reason behind this.

—----------

ARTICLE CLUSTERING MODEL

INTRODUCTION

In the article clustering model, I am utilizing the dataset "HW3_topics.csv", which contains information about the number of articles customers read in each topic in the past 3 months. The variables I have available are:

- Stocks
- Productivity
- Fashion
- Celebrity
- Cryptocurrency
- Science
- Technology
- SelfHelp
- Fitness
- AI
- id

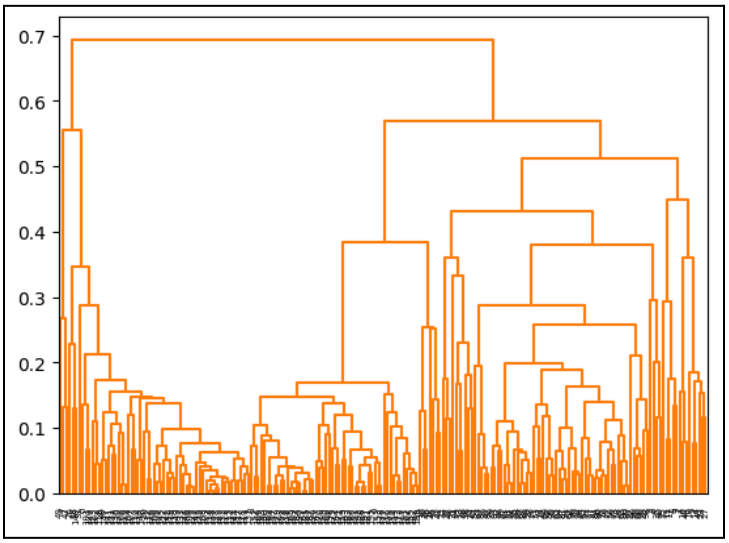
All of the variables correspond to the specific topics that customers are reading. The variables are all counts. By utilizing a Hierarchical Agglomerative Clustering model, I will be able to break up the customers into different customer groups and investigate the kinds of customers in each cluster. This

model is important for the company to gather information about their customers to best cater to their needs.

METHODS

After loading the data and all the necessary imports and checking for missing values, I created a list of features, including all the variables except “id” as it was not one of the article topics. I did not Z-score the variables as they were counts, and not continuous variables.

I started off by creating and fitting an empty HAC model with a cosine distance metric and average linkage but with the number of clusters not yet set. To figure out the number of clusters that I should use in my model, I plotted a dendrogram, using the “def plot_dendrogram()” function from sklearn, and calling the function with my empty HAC model. The dendrogram is displayed below:

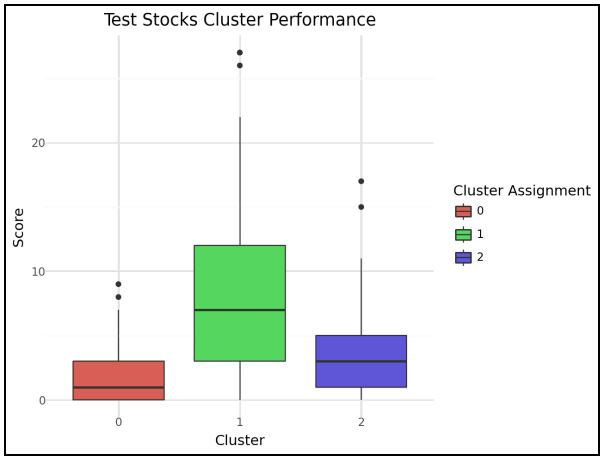


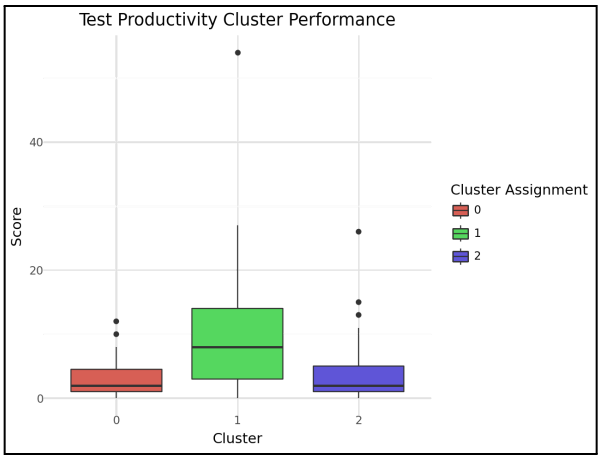
The dendrogram depicts that the data starts splitting off into more distinct clusters at about 0.4 - 0.5. When drawing a horizontal line around this range, I found that I was intersecting about 4 to 5 vertical lines, which indicates that the model would be best with somewhere from 3 to 5 clusters. However, after testing various numbers within the “num_clusters” part of my model and evaluating their silhouette scores, I found that 3 clusters yielded the highest silhouette score. So, I decided to continue the model by setting “num_clusters” equal to 3.

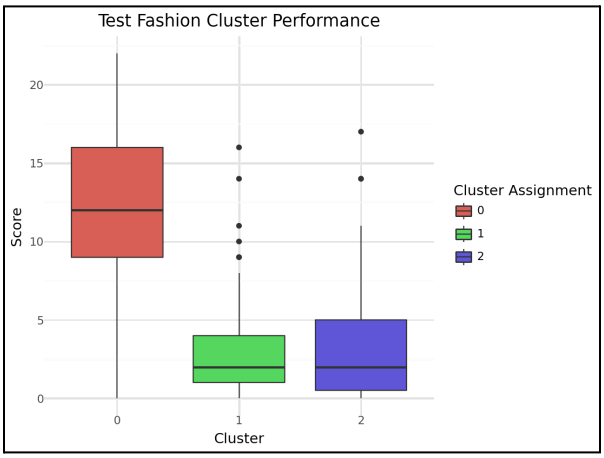
RESULTS

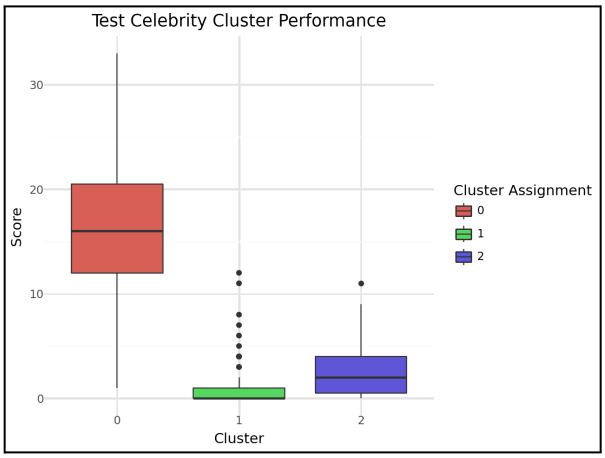
Upon refitting the HAC model with “num_clusters”, equal to 3, I calculated a silhouette score of 0.3166611891007902. While this score is relatively low, it is the highest among all the numbers in the range of (2,10). A silhouette score of 0.32 does not imply instability, but rather moderate separation. The clusters are distinguishable but not as consistent and may overlap slightly.

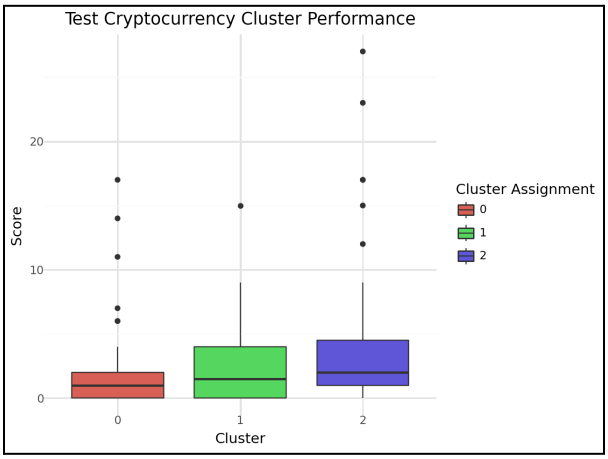
To test the cluster performance, I created ggplots for each feature:

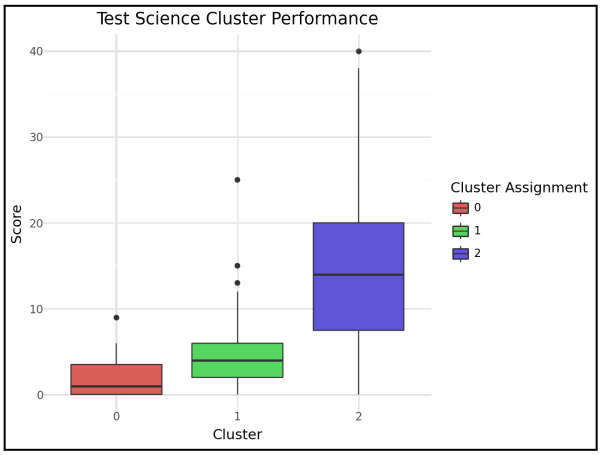

Test Stocks Cluster Performance
Score
Cluster
Cluster Assignment
0
1
2

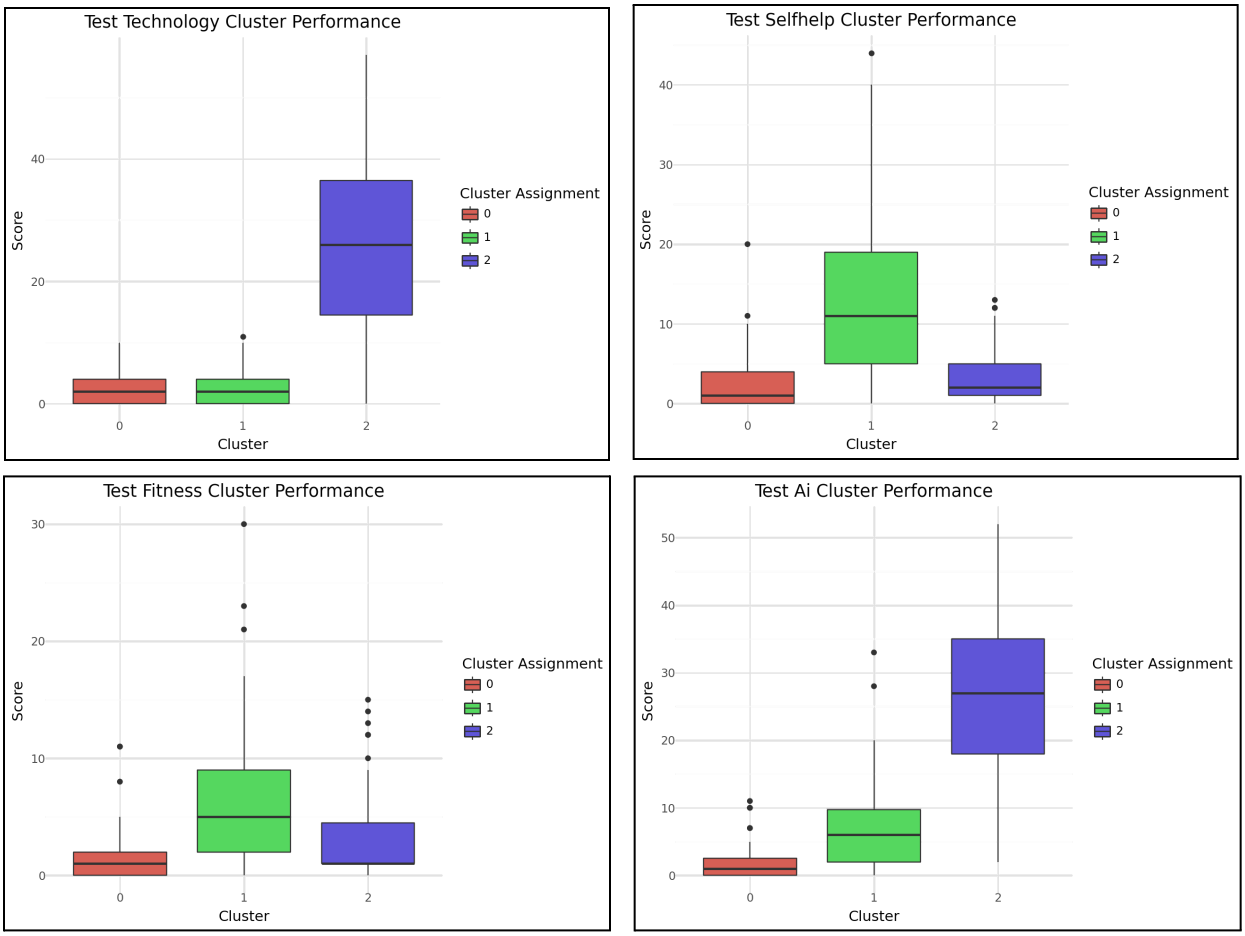

Test Productivity Cluster Performance
Score
Cluster
Cluster Assignment
0
1
2


Test Fashion Cluster Performance
Score
Cluster
Cluster Assignment
0
1
2


Test Celebrity Cluster Performance
Score
Cluster
Cluster Assignment
0
1
2


Test Cryptocurrency Cluster Performance
Score
Cluster
Cluster Assignment
0
1
2


Test Science Cluster Performance
Score
Cluster
Cluster Assignment
0
1
2



Additionally, to evaluate the kinds of customers in each cluster, I created a summary table with the means and standard deviations of the different clusters based on article topics.

	Stocks		Productivity		Fashion		Celebrity		Cryptocurrency	
	mean	std	mean	std	mean	std	mean	std	mean	std
cluster_3										
0	2.067797	2.455567	2.881356	2.792136	12.016949	5.447212	16.644068	6.310262	2.084746	3.265748
1	8.256098	6.670019	9.731707	8.805230	2.804878	3.233463	1.219512	2.444568	2.536585	2.986369
2	3.508475	3.701691	3.779661	4.548845	3.220339	3.868831	2.745763	2.844191	4.033898	5.558362

Science		Technology		SelfHelp		Fitness		AI	
mean	std	mean	std	mean	std	mean	std	mean	std
1.949153	2.160968	2.559322	2.608152	2.830508	3.714615	1.576271	2.422619	1.898305	2.637233
4.597561	4.174762	2.378049	2.406987	13.097561	9.911472	6.719512	5.940600	6.914634	6.369684
14.271186	8.882044	26.169492	14.283041	3.355932	3.633003	3.355932	3.929388	26.101695	12.793715

From the summary table, we can gather the most and least read topics in each cluster.

In cluster 0, Fashion has the highest average number of articles read at 16.64. The least read articles fall under AI, with an average of 1.90 articles read.

In cluster 1, Self Help has the highest average number of articles read at 13.1. The least read articles fall under Celebrity, with an average of 1.22 articles read.

In cluster 2, Science has the highest average number of articles read at 14.27. The least read articles fall under Celebrity, with an average of 2.75 articles read.

Using this information, the company can adjust the articles they are marketing to specific clusters.

—----------

DISCUSSION/REFLECTION

The analyses I performed allowed me to truly understand clustering algorithms and the steps that must be taken in order to create accurate models. I was able to become a lot more familiar with Gaussian Mixture Models, Principal Component Analysis, and Hierarchical Agglomerative Clustering. While it was difficult to understand how to best summarize and analyze the clusters, I believe that I was able to dive deep into the data and provide information that would allow the media company to implement specific and clear solutions. Something I would do differently is try out a different way to summarize the data at the end (instead of a summary table) or try a different clustering algorithm in the first question. I think my methods were very detailed and I was able to catch onto them quickly, but I would like to compare the methods with that of a different approach.