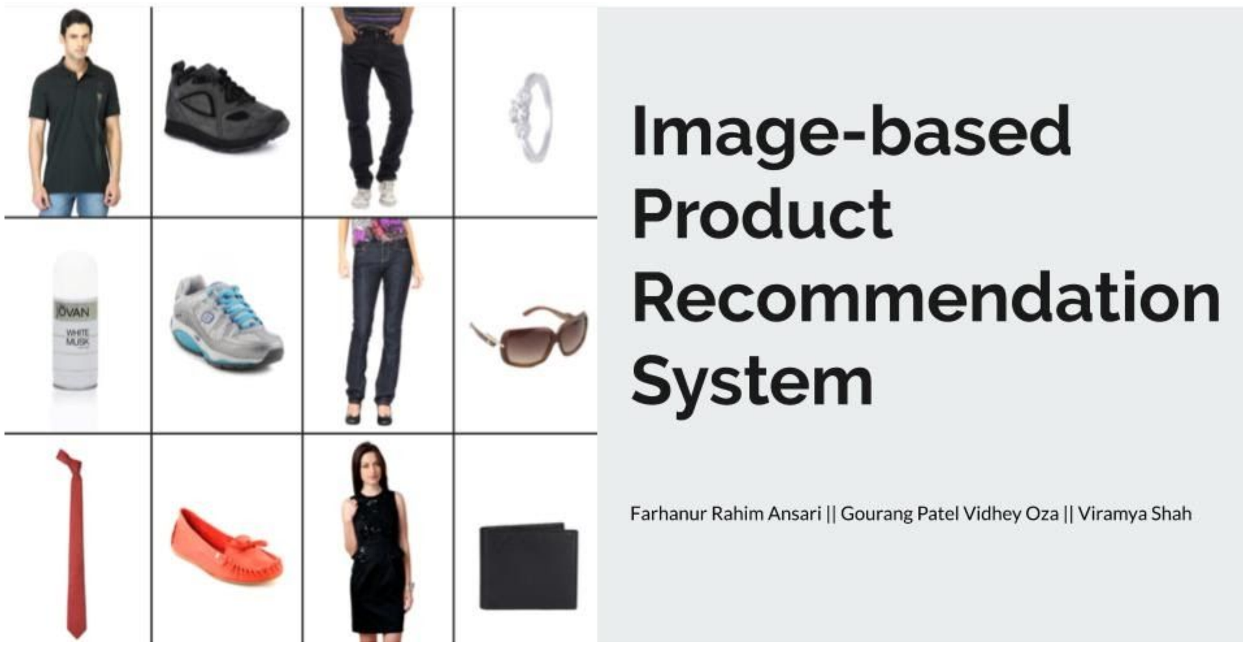
Image-based Product Recommendation System

Farhanur Rahim Ansari, Gourang Patel, Vidhey Oza, Viramya Shah

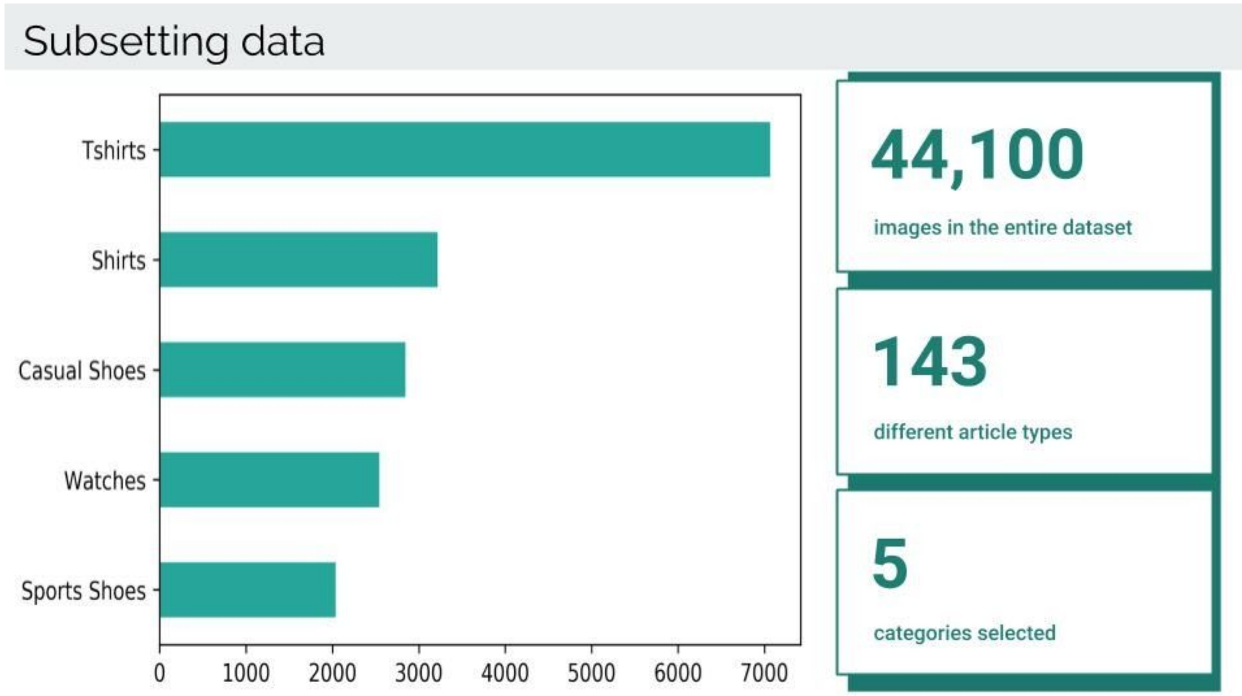


Idea

Current implementations of image recommendations are used by giant e-commerce portals like Amazon, Alibaba, etc but still stand inefficient in the sense they require user text input. We, on the other hand, intend to optimize the whole system by taking images as input from the user. This would enhance the customer experience and fill up the gaps faced in text searches

Problem Statement

Develop an image-based product recommendation system used primarily in the e-commerce domain. This engine will take as input an image of a certain product and recommend 5 similar products that resemble closely to the input image. We aim to achieve this goal by implementing a combination of dimensionality reduction, clustering, modelling and recommendation system techniques.



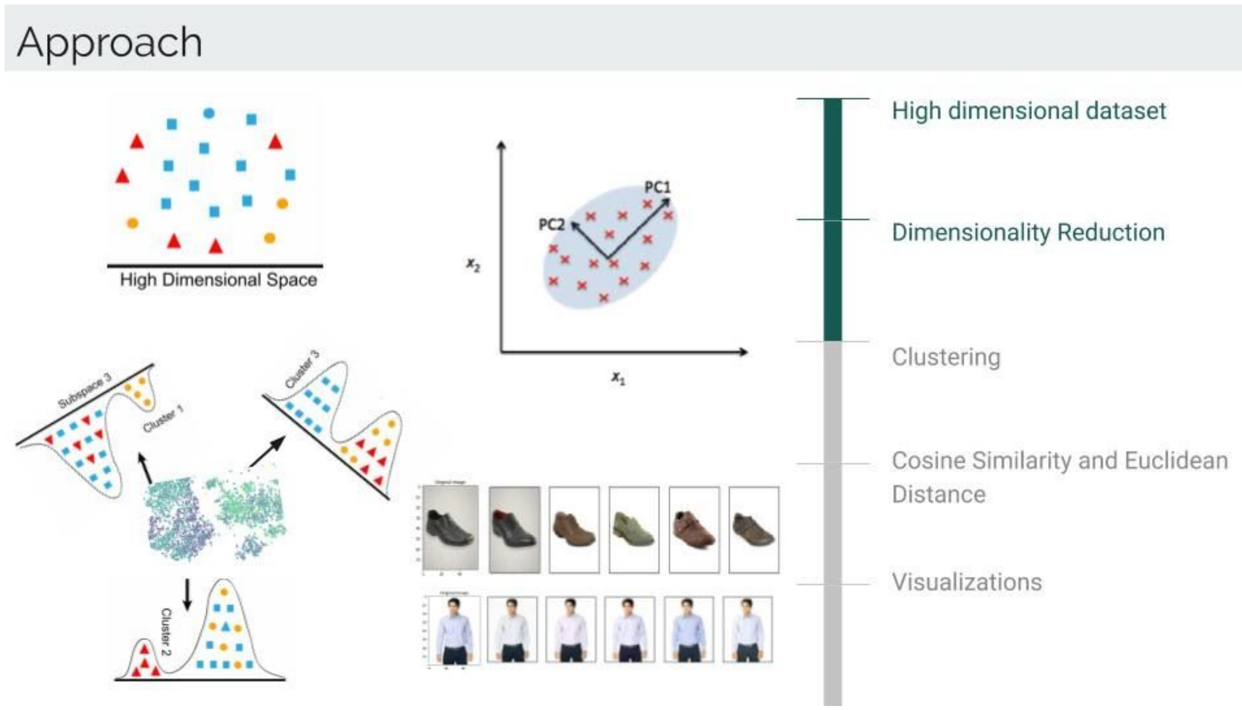
Dataset

The fashion industry is one of the most prominent parts of the e-commerce industry and has tons of data publicly available. We decided to use the **Fashion Product Image dataset** for our project. It's a rich dataset of product features and images spread across 7 master categories, 49 subcategories, and 143 article types. Each product has a unique id to distinguish them.

The dataset can broadly be divided into image data and style data. The data is described as follows:

1. Image data: Each image has a unique id and is of size 2400*1800*3. All the images have a clear white background with the object in the foreground.
2. Style data: The styles.csv contains metadata about the image. Analysis of various fields like 'gender', 'masterCategory', 'subCategory', 'articleType' tells us about the distribution of the data.

The dataset is highly imbalanced across the different article types. To overcome this shortcoming, we started our implementation by taking into account the top 5 article type and verifying it through a thorough analysis of the product features. The article types which we finalized are: T-shirts, Shirts, Casual Shoes, Watches and Sports Shoes. We used 1000 images of each article type. Considering the problem statement and the practical nature of the project, we implemented an Image Data Generator to perform image augmentation by rotating and capturing images through 4 different angles. The new augmented dataset consisted of 5000 images for each of the 5 article types.



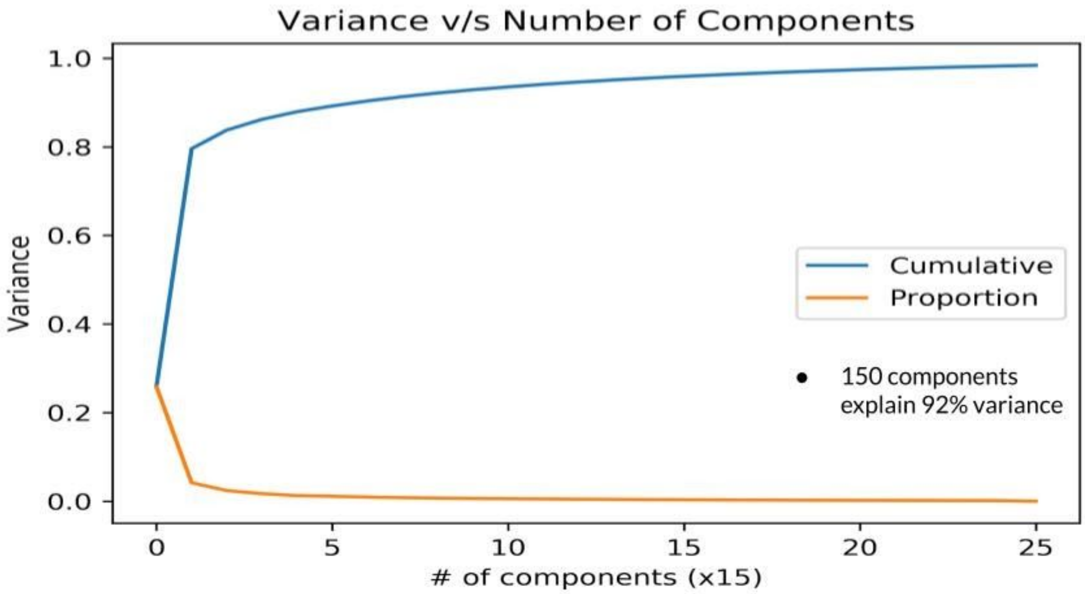
The approach we used is that on the complete dataset we employed the dimensionality reduction techniques and then used them as input in the clustering methods. Thus, we form significant clusters of different products we have.

For the given test image by the user as input on the e-commerce platform, we first employed dimensionality reduction and then predicted the cluster to which the image belongs to. For cluster prediction we employed and selected Agglomerative Clustering with linkage ward. We then employ similarity matrices like cosine and spatial distance to recommend similar fashion products to the user.

Both the techniques perform exceptionally well in recommending products similar to the input product. On an average, 80% of the recommended images are similar between the two approaches. While in extreme cases we see few differences in the output of both the techniques. In general, we observe that spatial distance technique is quite useful in capturing the aesthetics like the color and shades of the input image. While on the other hand, we observed that cosine similarity performs quite well in capturing the architecture, geometry and shape of the input product.

Thus, we recommend images similar to the user input and give users the choice to select desired similar products on an e-commerce website.

Dimensionality Reduction

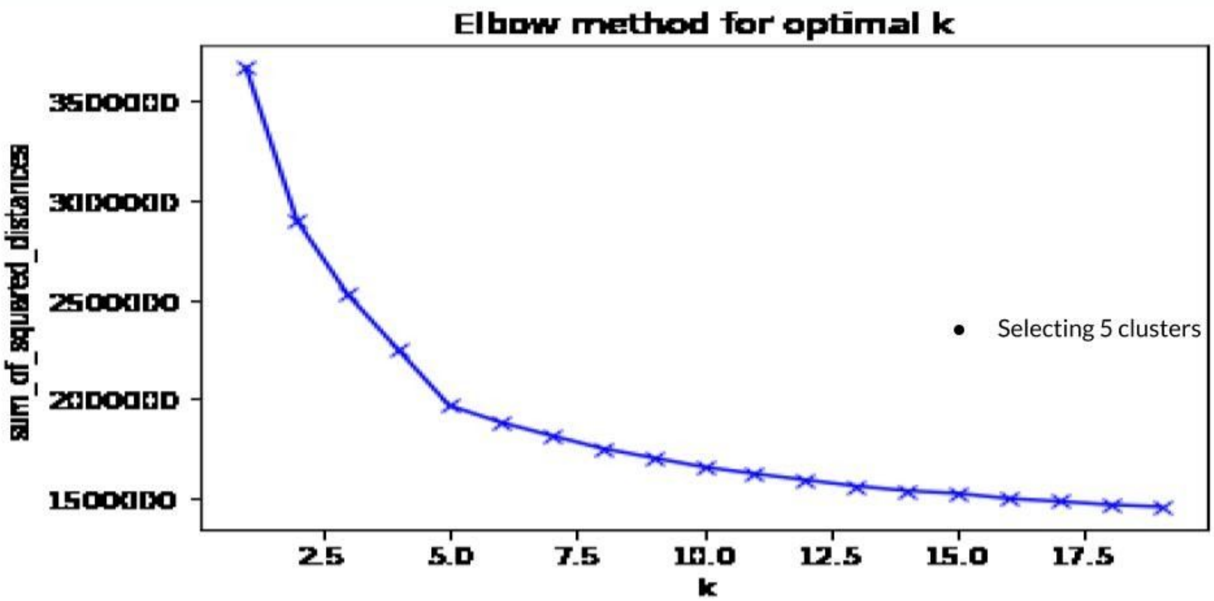


Dimensionality Reduction

Our dataset consisted of high-quality fashion product colour images. There are a total of 129600 (2400x1800x3) components in each image. Using all the dimensions and computing distances and comparing an input image with all the images available in the dataset would incur a lot of cost and slow retrieval to any e-commerce. Therefore, we employ dimensionality reduction to the complete dataset so as to get the lower subspace representation of an image and then using it further for clustering and recommendation.

Applying PCA on a subset of data (top categories as seen in the earlier slide) grouped by the 'articleType'. Promising results were observed. The results (as seen in the graph below) show that ~150 components (out of 129600) were able to capture 92% variation in the entire dataset.

Cluster selection

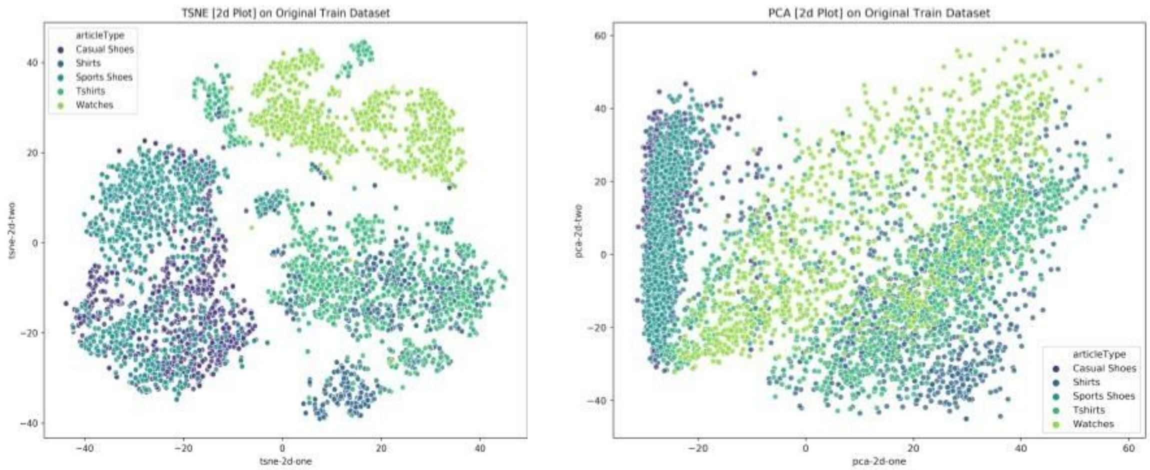


We employed cluster selection i.e identifying the optimum number of clusters for implementing clustering techniques using the Elbow Method for Optimal K**.** We calculated the sum of squared distances for different values of K and plotted it against the number of clusters K.

From the plot we can see that the sum of squared distance decreases with the increasing value of K which is logical as the least error will be observed when k will be equal to the number of images. However we can observe that at K = 5 we observe a sharp decrease similar in sum of squared distances. This implies that K = 5 will be the optimal number of clusters for the given dataset.

This value of K is also in accordance with the number of categories which we have in the dataset.

PCA & t-SNE plots

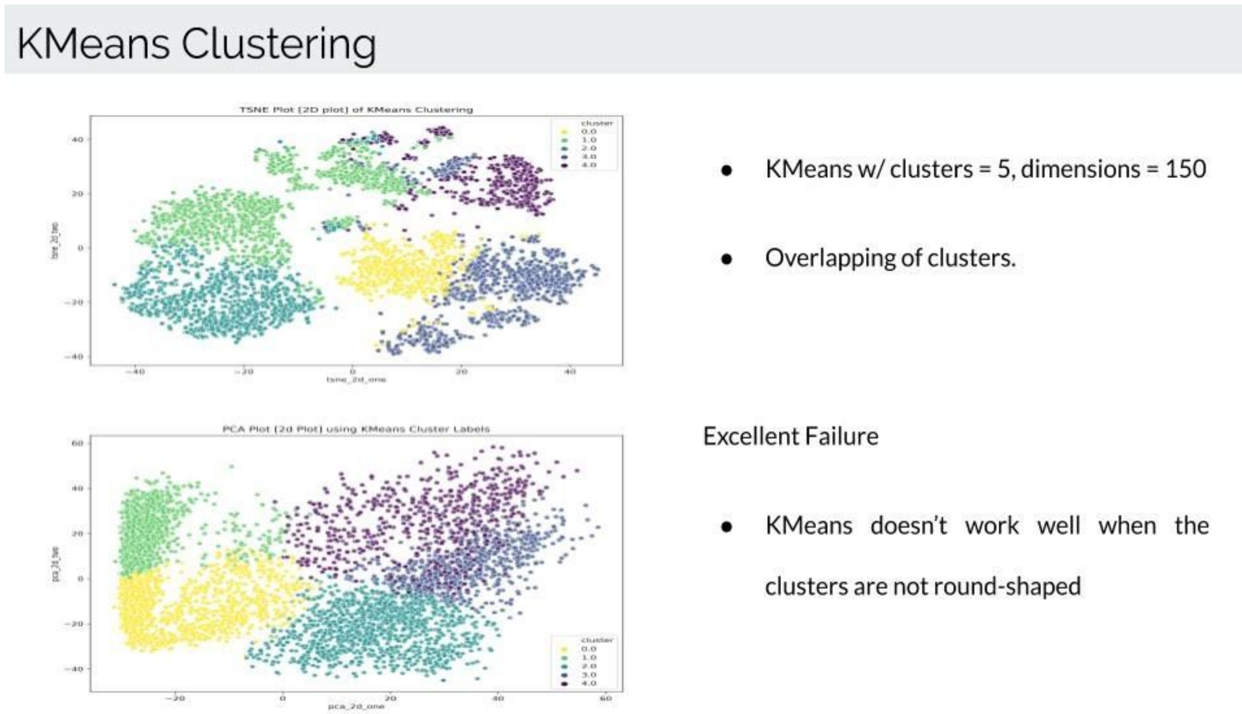


2D visualizations of images using t-SNE can be used to identify clusters of different products.

The train data consists of original labels i.e the articleType. We have used the subset of 5000 images from the dataset with the five distinct articleType which are {'Casual Shoes', 'Shirts', ' Sports Shoes', 'Tshirts', 'Watches'}. As we apply dimensionality reduction techniques mainly PCA and t-SNE. We observed that approximately 150 components capture the 92% variance in the data**.**

Using the first two components i.e PCA-1 and PCA-2 we visualized the data in 2D dimension for PCA. We observed that the clusters are not significant with the original labels. Hence, 2D visualization of images using PCA can't be used to identify clusters of different products**.**

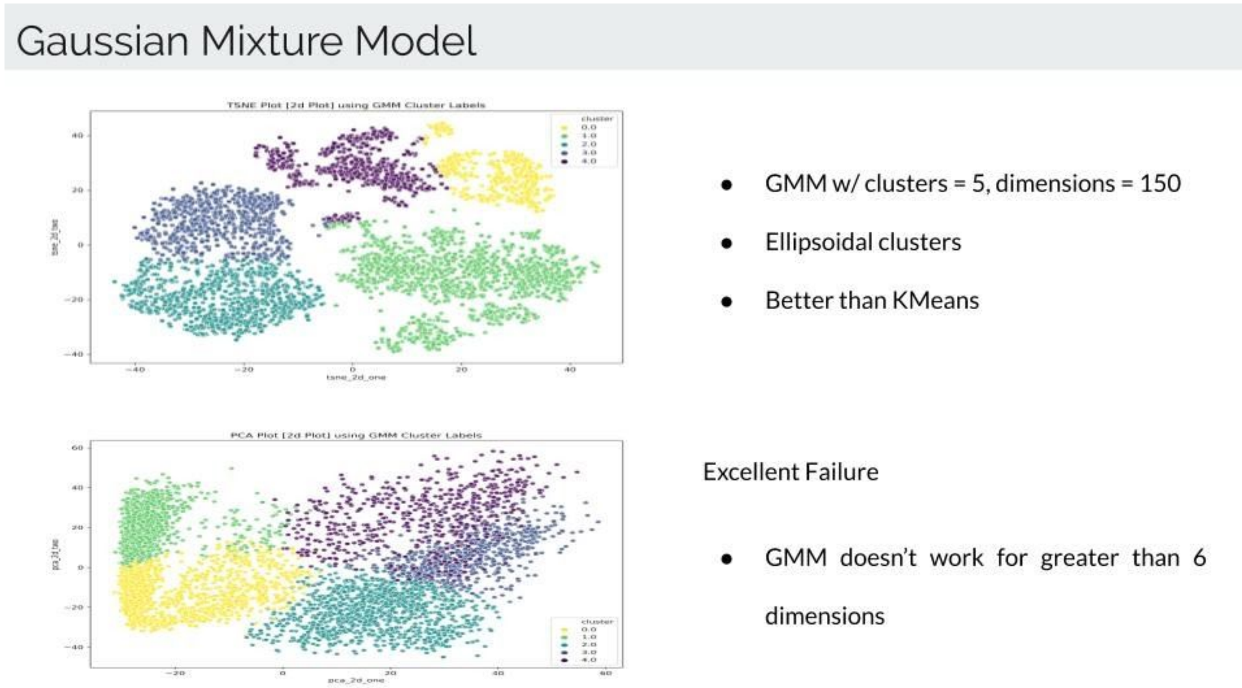
Similarly, we used ~150 components from PCA and employed t-SNE technique for dimensionality reduction. The perplexity parameter was set to 30 while plotting t-SNE. We observed that the 2D visualization of images using t-SNE can be used to identify clusters of different products as the clusters were quite significant and separated.



KMeans Clustering

KMeans is a distance-based approach to employ clustering. The n_clusters parameter was chosen to be 5, as we derived 5 to be the optimum cluster value using the sum of squared error vs n_cluster plot.

KMeans clustering was employed on the reduced dimensions image dataset to visualize clusters with both PCA and t-SNE components. We observe overlapping of clusters in both 2D plots for both PCA and t-SNE. The overlapping explains that the products are not segregated well and thus, we can't use the clusters to recommend similar products to the user as it would lead to an erroneous recommendation.
The clusters are not segregated well. As KMeans works well with clusters with round-shape. The reason can be explained as in KMeans we calculate the distance of each point from the centroids which is the basically the centre of the cluster.

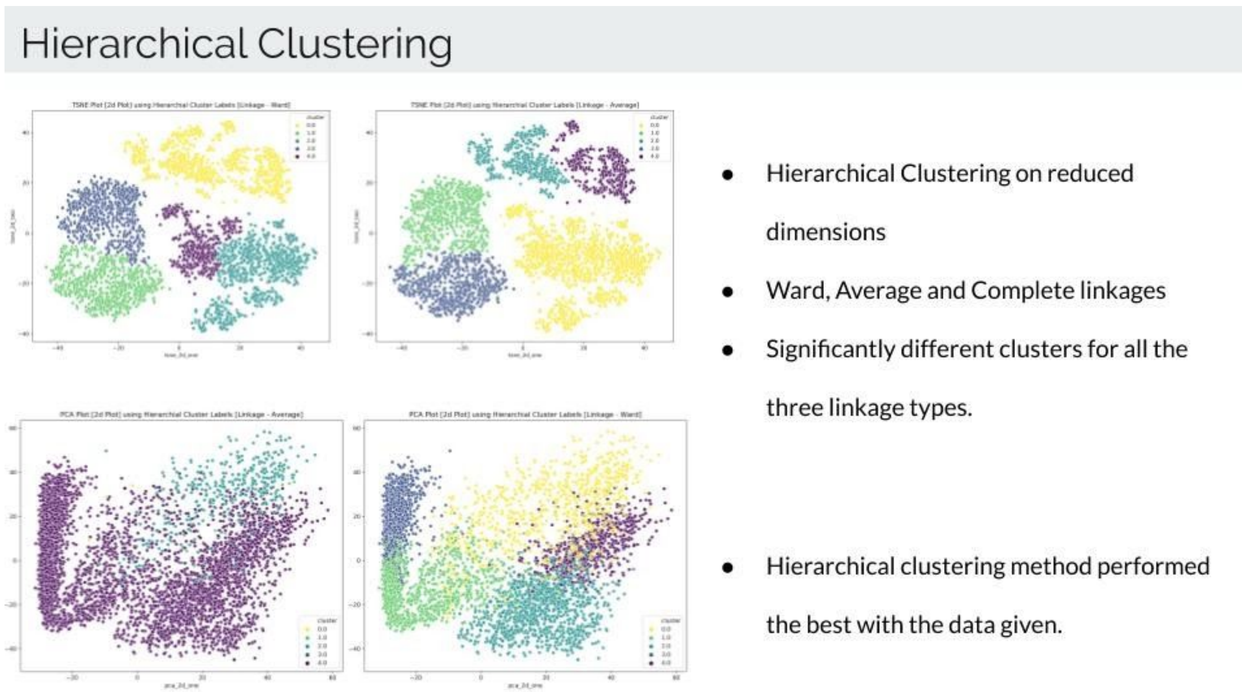


Gaussian Mixture Model

The distance approach used in KMeans Clustering for the fashion-based image dataset didn't perform well. Therefore, we employed the Gaussian Mixture Model (GMM) Clustering method on the dataset with reduced dimensions. Similar to KMeans here also the n_components parameter was set to 5, as we have 5 different article types in our dataset.

With GMM, the output clusters formed were significantly better. Even though the clusters show an overlap in the PCA reduced image, still the segregations are quite better than the previous KMeans approach. The possible overlap may be implying more realistic proximity of the article categories. This can be verified through the images considered in the dataset.

The main limitation with GMM clustering is that it doesn't work well for higher dimensions (especially for the dimension of data greater than 6). Considering the high dimensionality within our dataset it is required that we continue with clustering and try a hierarchical approach.

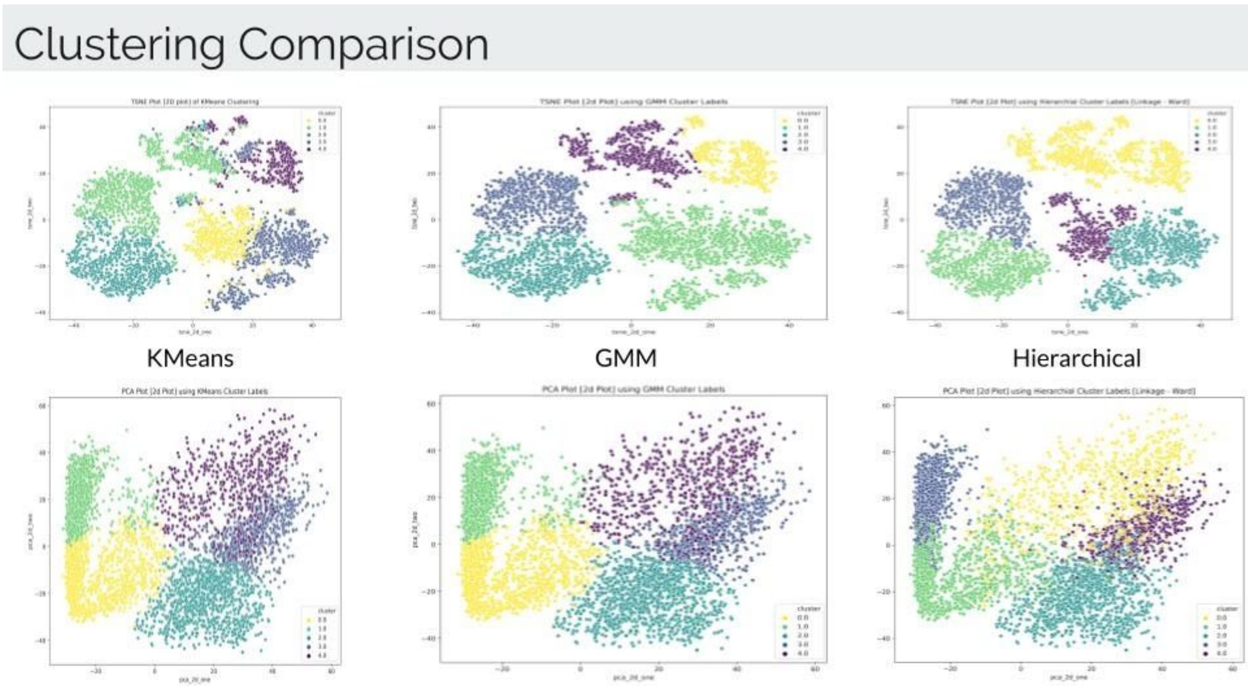


Hierarchical Clustering

KMeans failed as the clusters were not spherical in shape and GMM failed as the dimension of the dataset was quite large and GMM doesn't work well with higher dimensions. Therefore, we employed Hierarchical Clustering with two types of linkages - 'ward' and 'average'.
Hierarchical Clustering with 'average' linkage didn't perform very well. As we can observe overlapping of clusters in t-SNE and PCA both.

Hierarchical Clustering with 'ward' linkage gave the best results with t-SNE and significant results with PCA as well. We can further use the clusters from this technique and employ recommendation based on the cluster predicted of any input from the user. As the clusters were well separated and significant.

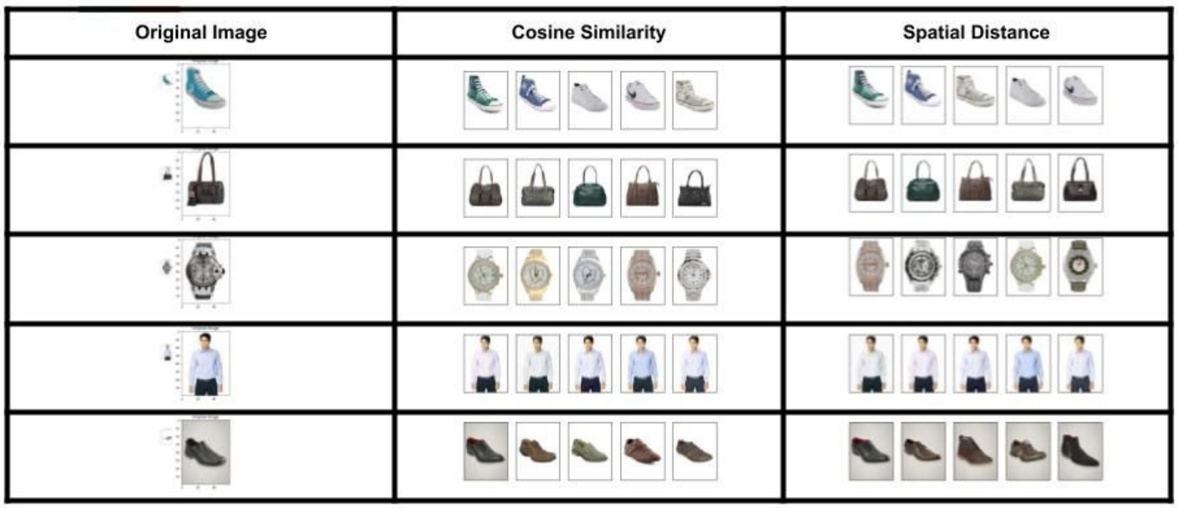
We will be further using results from Hierarchical Clustering- with linkage 'ward' for computing distances within a cluster and thus recommending top-5 similar images to the user-input.



Clustering Results Comparison

For the sake of completeness, here are all the clusters at a glance. To recap, we have Tshirts, Shirts, Casual Shoes, Sports Shoes, and Watches in the sub-dataset. One astute observation, for example, in Hierarchical clustering would be the isolated yellow cluster. Whereas the purple-cyan and green-blue would share boundaries. We can infer the yellow cluster as watches and the remaining 2 groups of clusters as either shirt-t shirts and casual/sports shoes.

Recommendation Results



Recommendation System

The input image is first dimensionally reduced to 150 components and is then sent to the recommendation engine which gives as output 5 similar images to the input image.For similar image recommendation, we employed 2 different techniques -

- Cosine Similarity:

$$Cos\theta = \frac{\vec{a} \cdot \vec{b}}{||\vec{a}|| \ ||\vec{b}||} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2} \sqrt{\sum_1^n b_i^2}}$$

where, $\vec{a} \cdot \vec{b} = \sum_1^n a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$ is the dot product of the two vectors.

- Spatial Distance:

$$||\boldsymbol{x}||_p = \left(\sum_i |x_i|^p \right)^{\frac{1}{p}}$$

Both the techniques perform exceptionally well in recommending products similar to the input product. On average, 80% of the recommended images are similar between the two approaches. While in extreme cases we see few differences in the output of both the techniques. In general, we observe that the cosine similarity technique is quite useful in capturing the aesthetics like the colour and shades of the input image. While on the other hand, we observed that spatial distance technique performs quite well in capturing the architecture, geometry and shape of the input product.