# Project 6 : Clustering

UNIVERSITÀ
DEGLI STUDI
DI MILANO

LE DELLIOU Julien

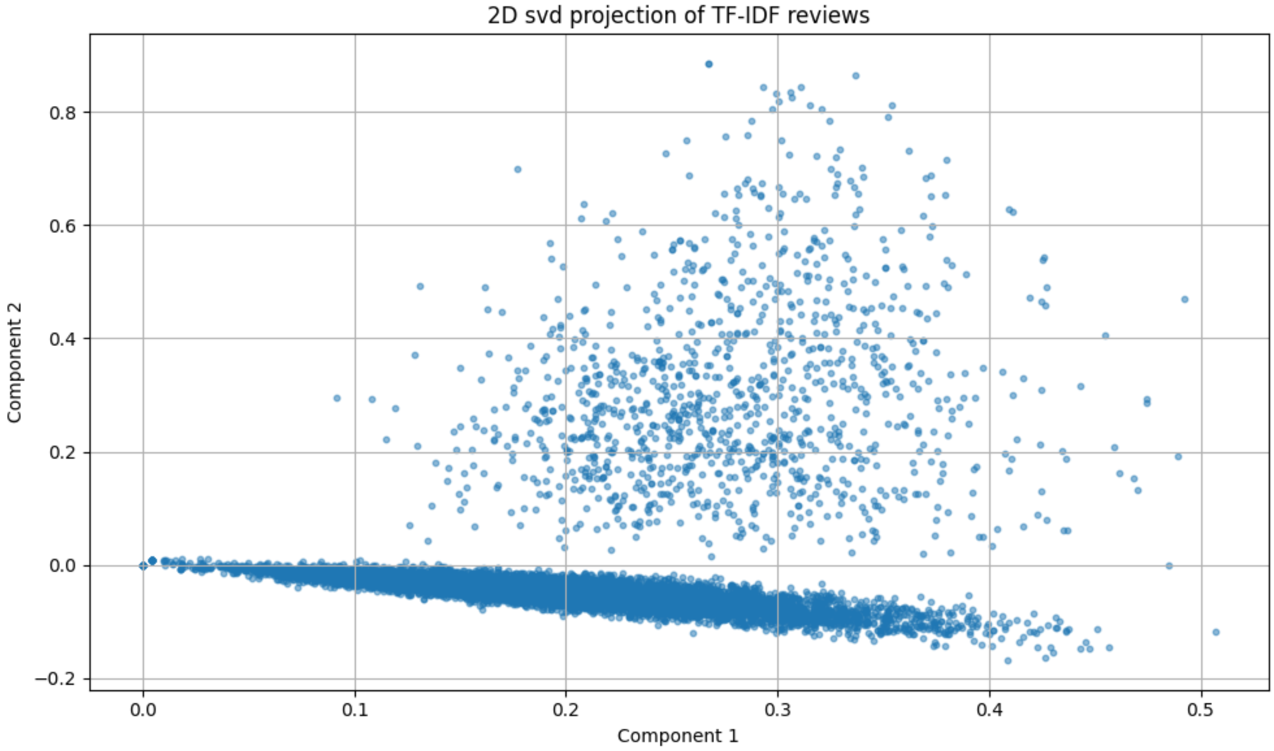
06/12/25

## Introduction :

First for this project I chose to focus on clustering text data. More specifically I focused on analyzing reader reviews. Each reader freely writes their review. This review can be of varying lengths. This will therefore involve unsupervised clustering. This technique aims to create coherent groupings.

The initial dataset is approximately 3 GB. This can be a large size for working on Google Collab.

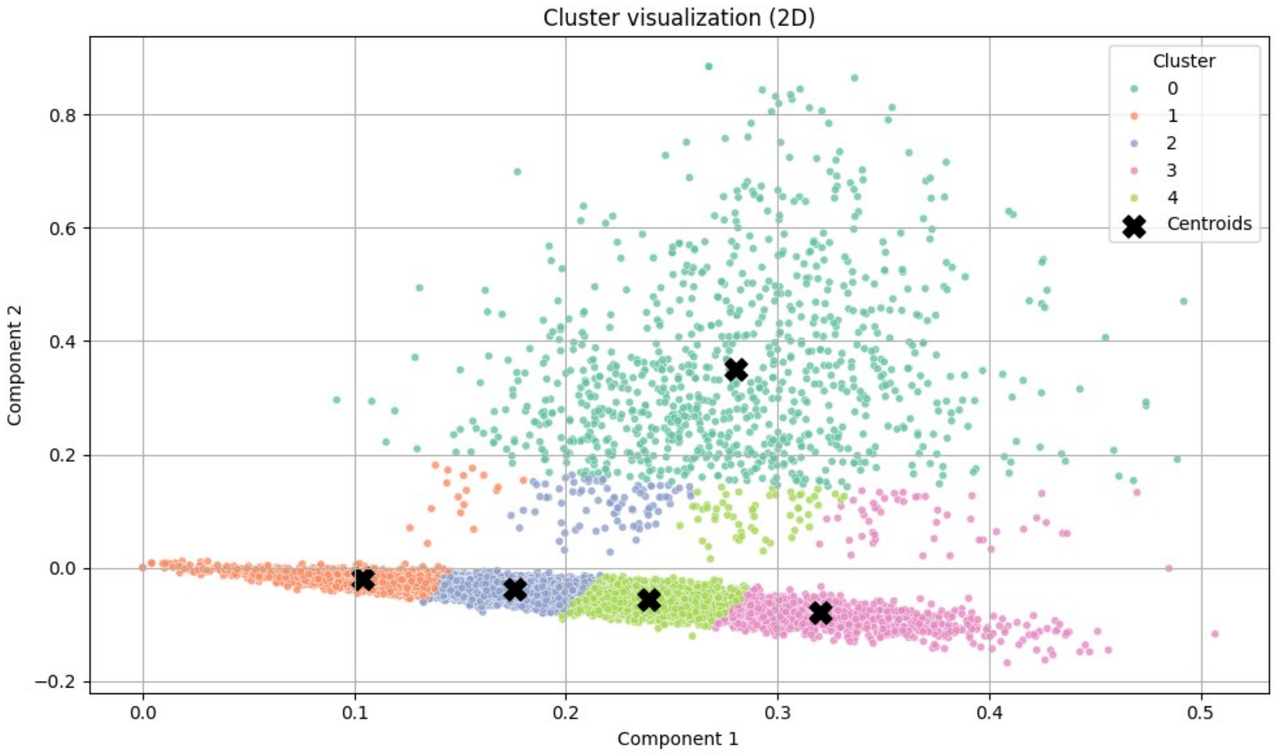
I first implemented the k-means algorithm, following the steps covered in class. I then evaluated the quality of the clustering using two methods. The first is the elbow method and the second is the silhouette score. Next I implemented the BFR (Bradley-Fayyad-Reina) algorithm. The latter allows the data to be processed in successive batches. This approach allowed me to maintain the scalability of the analysis.



This figure illustrates the initial dispersion of opinions after 2-dimensional SVD projection

## Data description :

The file this project is based on is called Books_rating. It contains several hundred thousand reviews. The dataset weighs over 3GB. This file size takes a long time to load into Google Collab. Therefore, I split this file into 12 equal pieces. Each file weighs about 250MB. For this project, I focused on the first file, which I called Books_rating_part1. However, all the code I wrote can be reused on the other parts.



This image illustrates the organization of data after vectorization and projection into the reduced space.

## Cleaning and vectorization :

Before applying a clustering algorithm, it was necessary to convert each text review into a numerical representation. To do this I lowercase all the texts to avoid capitalization issues, meaning that two words would be considered different simply because they had a capital letter or not. I also removed stopwords that didn't provide any information.

Next, I used a TD-IDF vectorizer to transform the texts into vectors. I limited the vocabulary to the 1 000 most relevant words. The goal was to reduce the size of the matrix and retain only the most informative terms.

Regarding the sparse matrix, each row represents a review and each column represents a word. Most of the cells contain zeros because the word didn't appear in the review.

Dimension reduction :

Even limited to 1000 dimensions the TF-IDF matrix remains difficult to exploit visually. I therefore applied a dimension reduction via the Truncated SVD algorithm which is a variant of PCA. This technique is more suited to sparse matrices. In addition it reduced the computation time. K-means algorithms converge faster on a reduced space.

Implementation of k-means clustering :

Once the reviews were vectorized and dimensionally reduced, I was able to implement the k-means algorithm. The algorithm proceeds as follows :

initialization : selection of k initial centroids, randomly placed in the reduced 20 dimensional vector space (via SVD)
assignment : each review is assigned to the nearest centroid, based on the Euclidean distance
update : the centroids are recalculated as the average of the points associated with them
convergence : I repeat steps 2 and 3 until the centroids move very little, or until the maximum number of iterations is reached

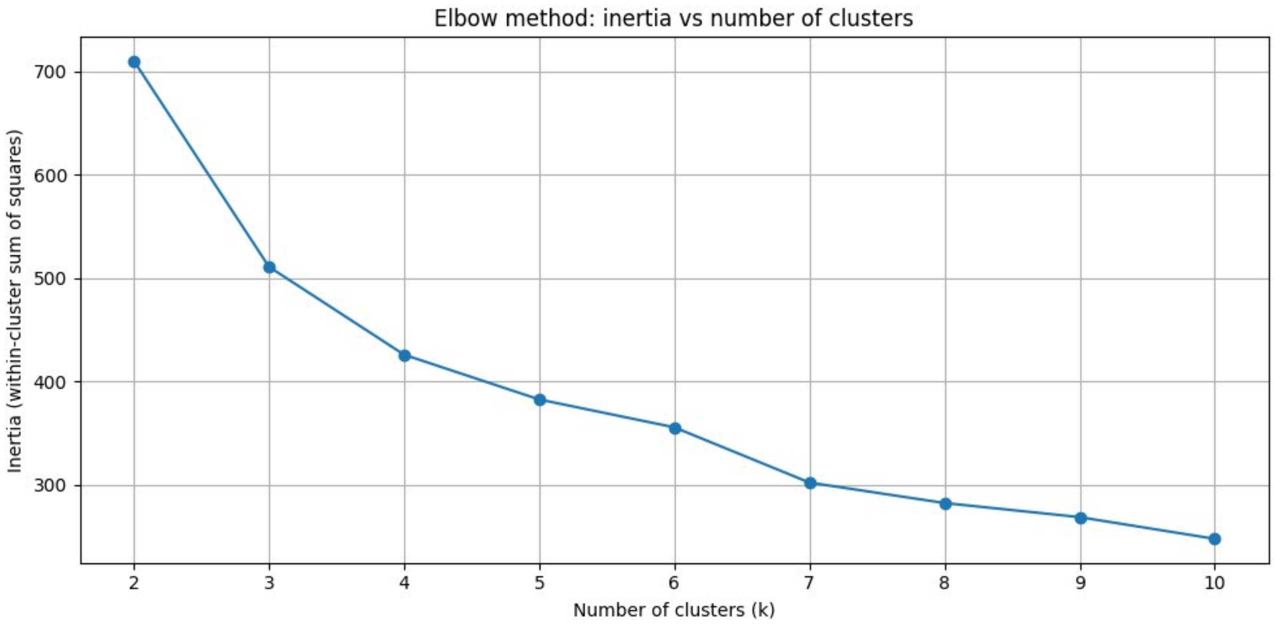
I tested several values of k between 2 and 10, to analyze the stability of the clusters formed and estimate their quality.

Evaluation of results :

Once I had implemented k-means I had to choose the number of clusters. For this I used two methods which are Elbow method and silhouette score.

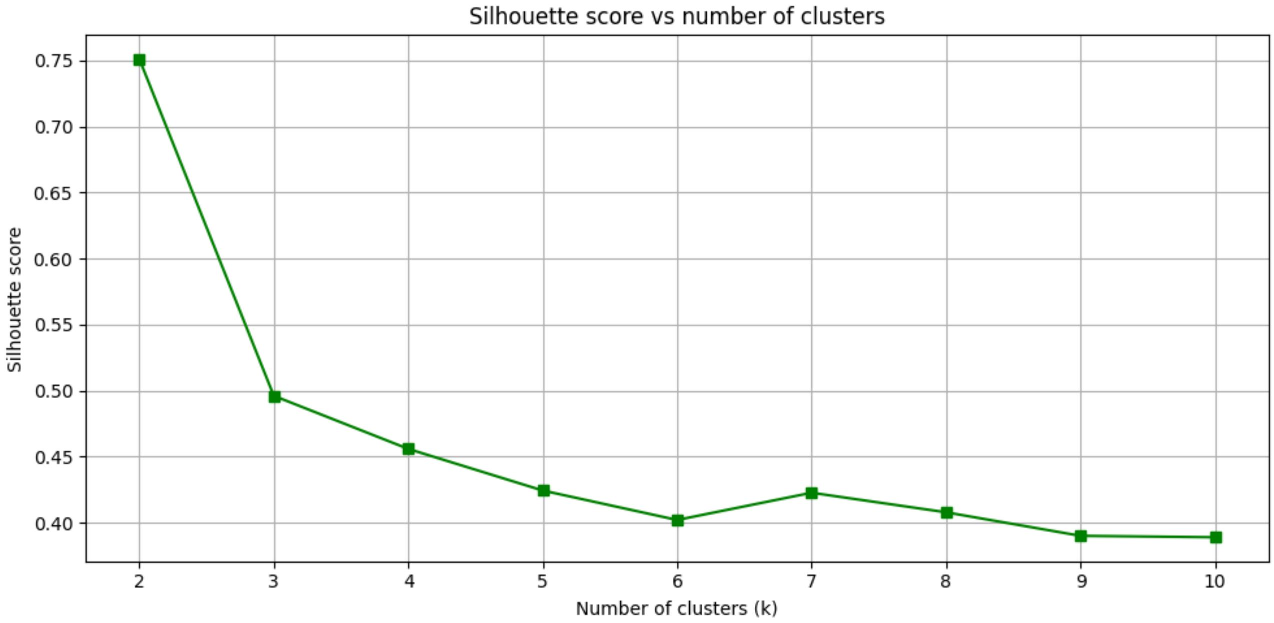
For the Elbow method :

I started by running K-Means for different values of k ranging from 2 to 10. At each iteration I retrieved the intra-cluster inertia, that is the sum of the distances between each point and its centroid. I then plotted the inertia as a function of k. the point where this curve forms the elbow often used to estimate an optimal k. In my case this was at k=5. So I kept this value.



For Silhouette score :

In my case, the highest score was obtained for k = 2 with a value close to 0.75. This suggests very well-defined two-group clusters. However, I chose to retain k = 5, because even though the score is a little lower (around 0.42) it is still reasonably high for text data.



Cluster visualization

After performing the clustering I wanted to visualize the results. Since the TF-IDF data remains in 20 dimensions, it was impossible to represent it directly. So I applied PCA to project the points into a 2D space.

Implementation of BFR :

As mentioned earlier, the 3GB file took a long time to load on Google Collab. Rather than limiting myself to a single sample, I wanted to implement a method adapted to massive data. Indeed, the BFR algorithm studied in progress is adapted. It allows clustering in batches. This therefore allows working on large volumes without saturating the memory.

Conclusion :

This project allowed me to put into practice several concepts covered in class. It allowed me to compare theoretical notions with practical notions. I was able to implement the k-means algorithm. To go further, I also implemented the BFR algorithm.

Declaration :

«  I/We declare that this material, which I/We now submit for assessment, is entirely my/our own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my/our work. I/We understand that plagiarism, collusion, and copying are grave and serious offences in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion or copying. This assignment, or any part of it, has not been previously submitted by me/us or any other person for assessment on this or any other course of study. No generative AI tool has been used to write the code or the report content »