**Netflix Movies and TV Shows clustering**

**Shubham Chandrakar**

**Data science trainees,**

**AlmaBetter, Bangalore**

**Abstract:**

The dataset provided to us by Flexible, a third party Netflix search engine. It consists of tv shows and movies available on netflix as of 2019. In 2019 Flexible released the report stating that the number of TV shows on Netflix has nearly tripled since 2010. While the number of movies on Netflix has decreased by 2,000 titles.

Our aim is to perform exploratory data analysis (EDA), understand which type of content is available in which country, to verify whether netflix is increasingly focusing on tv shows in spite of movie titles, and perform clustering of similar content by matching text based features.

***Keywords:machine learning, unsupervised machine learning, EDA, Netflix movies and tv show clustering, VIF analysis.***

**1. Problem Statement**

In this project, the main objective is to perform exploratory data analysis (EDA), understand which type of content available in which country, to verify whether netflix is increasingly focusing on tv shows in spite of movie titles, and perform clustering of similar content by matching text based features

Various features provided in the dataset were as follows:

1. show\_id : Unique ID for every Movie / Tv Show
2. type : Identifier - A Movie or TV Show
3. title : Title of the Movie / Tv Show
4. director : Director of the Movie
5. cast : Actors involved in the movie / show
6. country : Country where the movie / show was produced
7. date\_added : Date it was added on Netflix
8. release\_year : Actual Release Year of the movie / show
9. rating : TV Rating of the movie / show
10. duration : Total Duration - in minutes or number of seasons
11. listed\_in : Genre
12. description: The Summary description

We need to ascertain the effect of these features on the other features.

**2. Introduction**

The dataset provided to us by Flexible, a third party Netflix search engine. It consists of tv shows and movies available on netflix as of the year 2019. In the year 2019,Flexible released the report stating that the number of TV shows on Netflix has nearly tripled since the year 2010, while the number of movies on Netflix has decreased by 2,000 titles.

### Our goal is to build a predictive model, which could help us predict the actual number of clusters to categories the contents.

## **4. EDA relevant findings**

## **Runtime for movies is normally distributed and TV shows have less than 5 seasons on average**

It is found that the runtime of movies follows normal distribution. Mean is around 90-110 minutes. Most of the movie's length is less than 180 minutes.

Most TV shows have a single season only. And most of the TVshows have less than 5 seasons.

## **After 2017 there is decline in growth of new movie title**

There was a rising trend in growth in the number of both Movie and TV show titles before 2017. But after 2017 it is found that there is a decline in growth of the number of Movies in which there is a rise in the number of TV series.

## **Drama, Comedy and Documentaries are most popular genres**

Drama is the most popular genre, followed by Comedy, Documentaries , Action and adventure, TV dramas. Most of the content in netflix is based on these genres.

## **Most of the content has positive and neutral sentiment**

It is found that in any year most of the content has positive content followed by neutral content.

## **USA and India are the countries creating most of the movies for netflix**

It is seen that the USA is the top country in terms of production of movies followed by India. Other countries in the top five are the UK , Canada , France and Spain respectively.

## **USA and UK are the countries creating most of the TV shows for netflix**

It is seen that the USA is the top country in terms of production of TV series followed by the UK. Other countries in the top five are Japan , South Korea and Canada respectively.

**6. Steps involved:**

* **Data Wrangling**

After loading the dataset these step we performed these steps:

1. Checking duplicate row
2. Checking Null/Nan Values
3. Feature engineering.
4. Checking each individual feature.

* **Exploratory Data Analysis**

We performed this method by comparing one feature with another..This process helped us figuring out various aspects and relationships among the variables. It gave us a better idea of which feature behaves in which manner compared to the target variable.

* **Data preparation for clustering**

In these steps we removed punctuations from the document, after that we removed stopwords, followed by stemming.

* **Topic modeling**

In these step we preprocess the data and performed vectorization using CountVectorizer( )

* **Fitting different models**

For modeling we tried various regression algorithms like:

1. Clustering using Latent Dirichlet Allocation (LDA).
2. Clustering using K-means clustering algorithm.
3. Clustering using Agglomerative clustering algorithm.

* **Tuning the hyperparameters for better accuracy**

Tuning the hyperparameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting.

Method used for hyper parameter tuning : GridSearchCV

It is found that hyperparameter tuning can identify the best parameter and it results in boosting accuracy over the baseline model.

**7. Algorithms:**

**7.1 LDA (Linear Dirichlet Allocation):**

Latent Dirichlet allocation (LDA) is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words.

LDA assumes the following generative process for each document w in a corpus D:

1. Choose N ∼ Poisson(ξ).

2. Choose θ ∼ Dir(α).

3. For each of the N words wn:

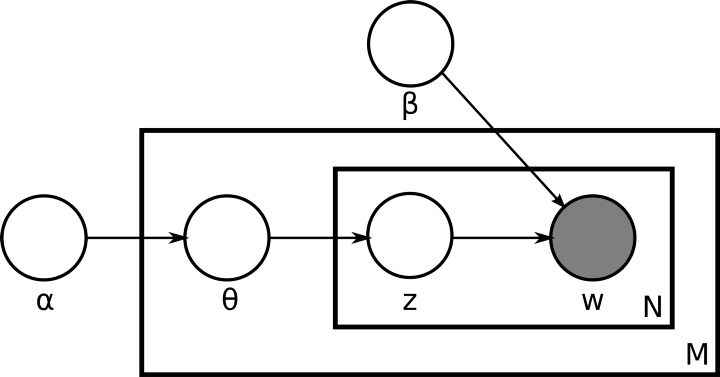
(a) Choose a topic zn ∼ Multinomial(θ).

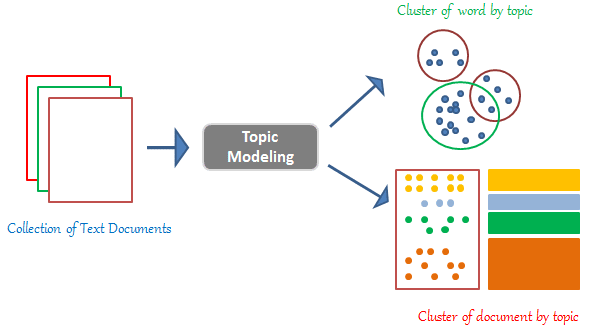
(b) Choose a word wn from p(wn |zn,β), a multinomial probability conditioned on the topic zn.

Several simplifying assumptions are made in this basic model, some of which we remove in subsequent sections. First, the dimensionality k of the Dirichlet distribution (and thus the dimensionality of the topic variable z) is assumed known and fixed. Second, the word probabilities are parameterized by a k ×V matrix β where βi j = p(wj = 1|zi = 1), which for now we treat as a fixed quantity that is to be estimated. Finally, the Poisson assumption is not critical to anything that follows and more realistic document length distributions can be used as needed. Furthermore, note that N is independent of all the other data generating variables (θ and z). It is thus an ancillary variable and we will generally ignore its randomness in the subsequent development.

A k-dimensional Dirichlet random variable θ can take values in the (k −1)-simplex (a k-vector θ lies in the (k−1)-simplex if θi ≥ 0, ∑k i=1 θi = 1), and has the following probability density on this simplex: p(θ|α) = Γ ∑k i=1αi ∏k i=1 Γ(αi) θα1−1 1 ···θαk−1 k , (1) where the parameter α is a k-vector with components αi > 0, and where Γ(x) is the Gamma function. The Dirichlet is a convenient distribution on the simplex — it is in the exponential family, has finite dimensional sufficient statistics, and is conjugate to the multinomial distribution. In Section 5, these properties will facilitate the development of inference and parameter estimation algorithms for LDA. Given the parameters α and β, the joint distribution of a topic mixture θ, a set of N topics z, and a set of N words w is given by:

p(θ, z,w|α,β) = p(θ|α) N ∏n=1 p(zn |θ)p(wn|zn,β)





.

**7.2 K-Means Algorithm:**

The K-means clustering algorithm computes centroids and repeats until the optimal centroid is found. It is presumptively known how many clusters there are. It is also known as the flat clustering algorithm. The number of clusters found from data by the method is denoted by the letter ‘K’ in K-means.

In this method, data points are assigned to clusters in such a way that the sum of the squared distances between the data points and the centroid is as small as possible. It is essential to note that reduced diversity within clusters leads to more identical data points within the same cluster.

Working of K-Means Algorithm

The following stages will help us understand how the K-Means clustering technique works-

Step 1: First, we need to provide the number of clusters, K, that need to be generated by this algorithm.

Step 2: Next, choose K data points at random and assign each to a cluster. Briefly, categorize the data based on the number of data points.

Step 3: The cluster centroids will now be computed.

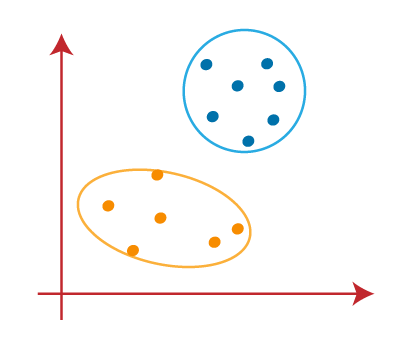
Step 4: Iterate the steps below until we find the ideal centroid, which is the assigning of data points to clusters that do not vary.

4.1 The sum of squared distances between data points and centroids would be calculated first.

4.2 At this point, we need to allocate each data point to the cluster that is closest to the others (centroid).

4.3 Finally, compute the centroids for the clusters by averaging all of the cluster’s data points.

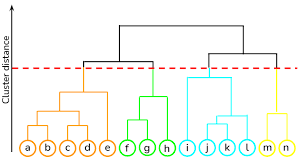
K-means implements the Expectation-Maximization strategy to solve the problem. The Expectation-step is used to assign data points to the nearest cluster, and the Maximization-step is used to compute the centroid of each cluster.



**7.3 Hierarchical Clustering:**

Hierarchical clustering,also known as hierarchical cluster analysis*,* is an algorithm that groups similar objects into groups called clusters. The endpoint is a set of clusters*,* where each cluster is distinct from each other cluster, and the objects within each cluster are broadly similar to each other.

Hierarchical clustering starts by treating each observation as a separate cluster. Then, it repeatedly executes the following two steps: (1) identify the two clusters that are closest together, and (2) merge the two most similar clusters. This iterative process continues until all the clusters are merged together. This is illustrated in the diagrams below.



**8. Model performance:**

Model can be evaluated by various metrics such as:

**1. Elbow method**

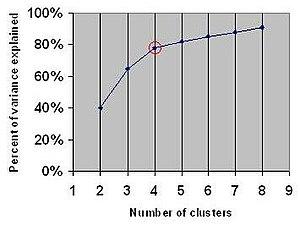
In cluster analysis, the elbow method is a heuristic used in determining the number of clusters in a data set. The method consists of plotting the explained variation as a function of the number of clusters and picking the elbow of the curve as the number of clusters to use. The same method can be used to choose the number of parameters in other data-driven models, such as the number of principal components to describe a data set.

The method can be traced to speculation by Robert L. Thorndike in 1953.

Using the "elbow" or "knee of a curve" as a cutoff point is a common heuristic in mathematical optimization to choose a point where diminishing returns are no longer worth the additional cost. In clustering, this means one should choose a number of clusters so that adding another cluster doesn't give much better modeling of the data.

The intuition is that increasing the number of clusters will naturally improve the fit (explain more of the variation), since there are more parameters (more clusters) to use, but that at some point this is over-fitting, and the elbow reflects this. For example, given data that actually consist of *k* labeled groups – for example, *k* points sampled with noise – clustering with more than *k* clusters will "explain" more of the variation (since it can use smaller, tighter clusters), but this is over-fitting, since it is subdividing the labeled groups into multiple clusters. The idea is that the first clusters will add much information (explain a lot of variation), since the data actually consist of that many groups (so these clusters are necessary), but once the number of clusters exceeds the actual number of groups in the data, the added information will drop sharply, because it is just subdividing the actual groups. Assuming this happens, there will be a sharp elbow in the graph of explained variation versus clusters: increasing rapidly up to *k* (under-fitting region), and then increasing slowly after *k* (over-fitting region).

In practice there may not be a sharp elbow, and as a heuristic method, such an "elbow" cannot always be unambiguously identified.



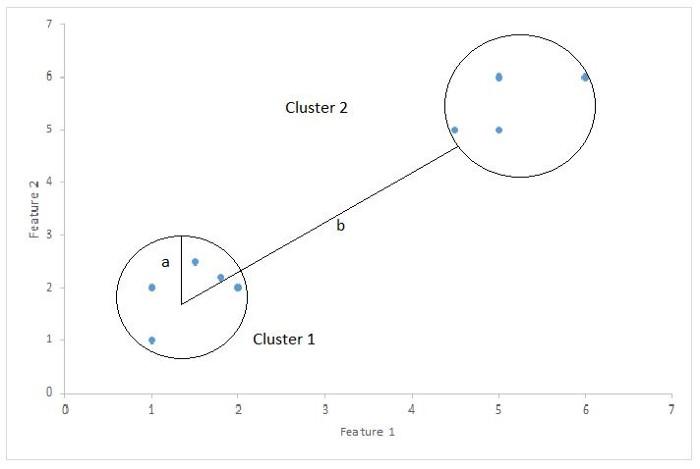
**2. Silhouette Coefficient/Score**

Silhouette Coefficient or silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1.

1: Means clusters are well apart from each other and clearly distinguished.

0: Means clusters are indifferent, or we can say that the distance between clusters is not significant.

-1: Means clusters are assigned in the wrong way.



Silhouette Score = (b-a)/max(a,b)

Where,

a= average intra-cluster distance i.e the average distance between each point within a cluster.

b= average inter-cluster distance i.e the average distance between all clusters.

But the problem is when we add an irrelevant feature in the dataset then at that time R2 sometimes starts increasing which is incorrect.

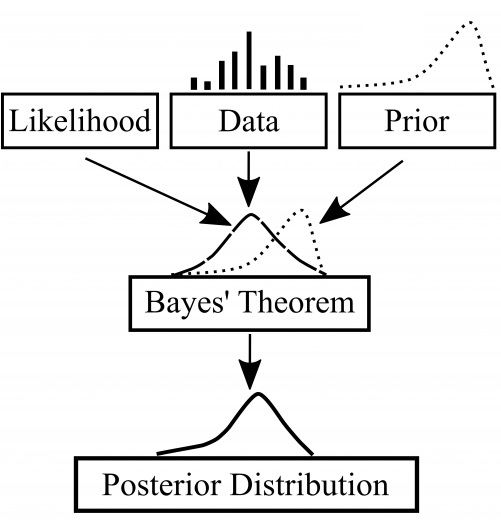
**9. Hyper parameter tuning:**

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used Grid Search CV, Randomized Search CV and Bayesian Optimization for hyperparameter tuning. This also results in cross validation and in our case we divided the dataset into different folds. The best performance improvement among the three was by Bayesian Optimization.

1. **Grid Search CV-**Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model’s performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.
2. **Randomized Search CV-** In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters, since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist’s control

# **Bayesian Optimization-** Bayesian Hyperparameter optimization is a very efficient and interesting way to find good hyperparameters. In this approach, in naive interpretation way is to use a support model to find the best hyperparameters.A hyperparameter optimization process based on a probabilistic model, often Gaussian Process, will be used to find data from data observed in the later distribution of the performance of the given models or set of tested hyperparameters.



As it is a Bayesian process at each iteration, the distribution of the model’s performance in relation to the hyperparameters used is evaluated and a new probability distribution is generated. With this distribution it is possible to make a more appropriate choice of the set of values that we will use so that our algorithm learns in the best possible way.

**8. Conclusion:**

It is found out that after 2017 there is decline in growth of new movie titles after 2017. Top 5 countries with the most movies on Netflix are the USA,India,UK, Canada and France . Top 5 countries with the most TV shows are the USA, UK, Japan, South Korea and Canada. As we performed topic modeling, LDA gives better results as data contains multiple topics. We can say that LDA with optimal number of clusters i.e. 2 is the best model.

**References-**

1. MachineLearningMastery
2. GeeksforGeeks
3. Analytics Vidhya