Model Based Document Classification and Clustering

Members :
Ashwin Pathak
Alakh Desai
Mohit Chandra
Vidit Jain

TA Mentor:
Aryaman Gupta

Aim

Primary Objectives

- ➔ The primary focus of this paper is topic detection, i.e. assigning the documents in a collection C to “topics”.
- ➔ In this project we focus on the case when the documents are assigned only one topic..
- ➔ We also touch upon “topic tracking” , i,e to assign a new document to one of the topics detected in C.

Detection vs Tracking

Detection is a clustering problem

while

Tracking is a classification problem

Clustering and classification methods play a central role in the reduction of both the number of operations needed for document classification, and the retrieval time.

Hence, we need to discover a map from documents to feature vectors that eases the task for the clustering method

Approach

Document dimensionality reduction is based on the K- nearest-neighbor classifier. We choose to work with this classifier because of its simplicity and lack of assumptions on the distributional properties of the documents.

GMDC

Gaussian Mixture Document Clustering

- Each of the components in the mixture distribution is assumed to be a multivariate Gaussian distribution with uncorrelated components.
- The model is used to build clusters based on the likelihood of the data, and to classify documents according to Bayes rule.
- Main advantage is the ability to automatically estimate the number of clusters (topics) present in the document collection via Bayes factors.

The Effect of Feature Selection on Document Classification

Converting documents into vectors

- ➔ If total number of words (terms) are p, then each document can be represented as a p-dimensional vector and n documents as nxp dimensional vectors.
- ➔ However, it gives a sparse matrix, hence, we apply transformations like square-root and logarithm.
- ➔ This reduces the influence of high counts.

- ➔ Note: We can also choose binary counts,i.e., 1 if the word is present in the document, otherwise, 0.

Weighting the terms

To account for this disparity between terms, several global weighting schemes can be used.. They are global in the sense that the weights reflect the distribution of terms over the entire document collection

Some proposed choices for the weight assigned to the j-th term are:

Identity: $w_j = 1$

Normal: $w_j = 1/\sqrt{\sum_i f_{ij}^2}$

Global frequency Inverse document frequency (GfIdf):

$w_j = \sum_i f_{ij} / \sum_i I(f_{ij} > 0)$, where I denotes the indicator function.

Inverse document frequency: $w_j = \log(n / \sum_i I(f_{ij} > 0))$

Entropy: $w_j = 1 + \sum_i p_{ij} \log p_{ij} / \log n$, with $p_{ij} = f_{ij} / \sum_i f_{ij}$.

Normalization and Representation

We will evaluate all 15 combinations : for each weighting choices, 3 types of transformations (untransformed, square-root and logarithm).

Final Matrix

The transformed and weighted term frequencies are given by :

$$x_{ij} = \frac{w_j \times g(f_{ij})}{\sqrt{\sum_k (w_k \times g(f_{ik}))^2}},$$

where w_j is the weight associated with the j-th term and g(.) is the term transformation.

Dimensionality reduction

➔ Representing each document by a p-dimensional vector has following disadvantages :

- It is costly : Due to sparsity of the vector. PCA can help in representing data in lower dimensions efficiently.

- Representing documents by high dimensional term frequency vectors might even be detrimental to performance as in the case of lexical matching.

Principal Component Analysis

Reduces the dimensionality of the vector :

$$\Sigma = 1/n\,\tilde{X}^t\tilde{X}, \quad \text{where}$$
$$\tilde{X} = \tilde{X} = (I - 1/n\mathbf{1}\mathbf{1}^t)X$$

is obtained from X by mean centering the columns. Here $\mathbf{1} = (1, \ldots, 1)$. Let

$$\Sigma = A\Lambda A^t$$

Alternatively,

$$\tilde{X}^t\tilde{X} = \tilde{V}\tilde{\Phi}^2\tilde{V}^t$$

Latent Semantic Indexing

Latent semantic indexing reduces dimensionality by projecting the document vectors onto the closest q-dimensional linear subspace, whereas principal component analysis projects them onto the closest affine subspace.

Like PCA, it also reduces dimensions

It finds the singular value decomposition $X=U\Phi Vt$ of X— the columns of X are not mean-centered. Dimensionality reduction is achieved as in principal component analysis: $yi=Vtqxi$, where Vq is the matrix consisting of the leading q columns of V

GMDC

Gaussian Mixture Model

- For feature vectors $x_1, x_2, \ldots x_n$ are sampled as: $f(x) = \sum_{g=1}^{G} p_g \mathcal{N}_q(x; \mu_g, \Sigma_g),$

- where, G is number of components in the mixture. q is the dimension of reduced feature vector. {p} are mixture proportions and N denotes the q-variate normal density with mean u and covariance Σ_g
- However, covariance matrix can become too large, hence, we use only diagonals : $\Sigma_g = \mathbf{diag}(\sigma_{1g}^2, \ldots, \sigma_{qg}^2).$

Selecting the Number of Clusters

Advantage of GMDC is its ability to compute the likelihood of the model through Bayes factors.

Let D denote the data and M_1 and M_2 be two different mixture models. The Bayes factor for model M_2 against M_1 is the ratio

$$P(D / M_2) / P(D / M_1);$$

BIC (Bayes Information Criterion) :
when the prior on the parameters is a multivariate normal distribution with mean equal to the MLE of the parameters, and variance-covariance matrix equal to the inverse of the Fisher information matrix given by the model, the likely model is equivalent to choosing max(BIC).

$$\text{BIC} = 2 \text{ log-likelihood} - r \log(n),$$

n : |D|,
r: no. of parameters

Assigning Documents to clusters

Assignment is simply given as

$$g = \arg \max_{g'=1,\ldots,G} p_{g'} \mathcal{N}_q(x; \mu_{g'}, \Sigma_{g'}).$$

Empirical Measures : Fowlkes-Mallows-Wallace Index

Topics	Apparent Partition A_1	A_2	...	A_G	Total
T_1	n_{11}	n_{12}	$\cdots$	n_{1G}	$n_{1\cdot}$
T_2	n_{21}	n_{22}	$\cdots$	n_{2G}	$n_{2\cdot}$
...	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
T_J	n_{J1}	n_{J2}	$\cdots$	n_{JG}	$n_{J\cdot}$
Total	$n_{\cdot 1}$	$n_{\cdot 2}$	$\cdots$	$n_{\cdot G}$	n

FMW Index is given by $\sum_{i,g} \binom{n_{ig}}{2} \Big/ \sqrt{\sum_i \binom{n_{i\cdot}}{2} \sum_g \binom{n_{\cdot g}}{2}}$.

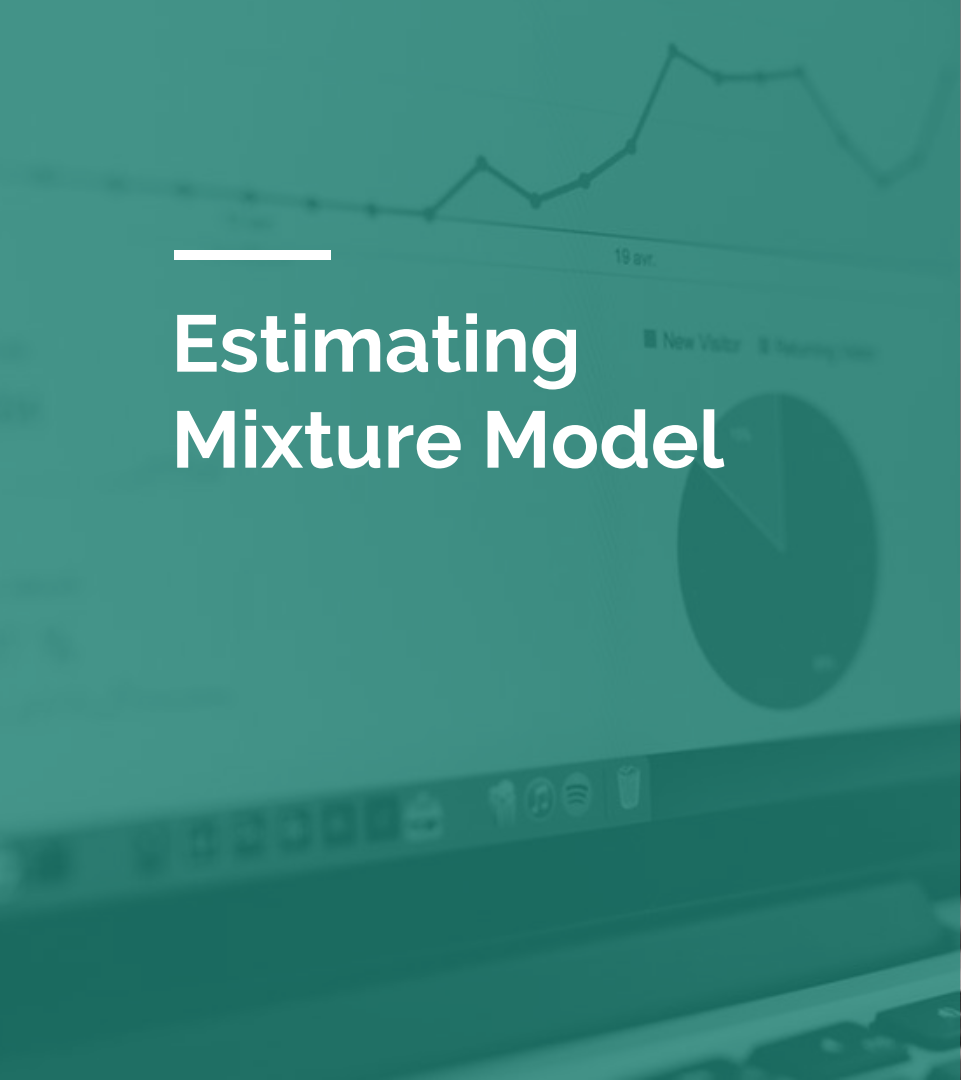
Empirical Measures : F1 Index

F1 Index is given as :

$$F_1(d) = 2\frac{p(d)r(d)}{p(d)+r(d)} = \{\frac{1}{2}(\frac{1}{r(d)} + \frac{1}{p(d)})\}^{-1}$$

giving rise to the F_1 average

$$F_1 = \sum_{d \in \text{ data collection}} F_1(d) \times \frac{1}{n} = 2\sum_{i,g} \frac{n_{ig}^2}{n_{i\cdot} + n_{\cdot g}} \frac{1}{n}.$$



Estimating Mixture Model

We make use of Expectation-Maximization Algorithm to estimate the parameters of GMDC.

EM gives explicit iterative updating formulas for the parameters associated to Gaussian mixtures for most constrained structures of variance-covariance matrices.

EM Updating Formulas

Let theta be parameters, and D = {x} be data. Then log-likelihood is given as :

$$z_{ig} = \begin{cases} 1 & \text{if } g \text{ is the cluster containing data item } x_i, \\ 0 & \text{otherwise,} \end{cases}$$

$i = 1, \ldots, N$, $g = 1, \ldots, G$. Then the complete log-likelihood of the model given $\{(x_i, z_i)\}_1^n$ is

$$l(\theta|(x_1, z_1), \ldots, (x_n, z_n)) = \sum_{i=1}^{n} \sum_{g=1}^{G} z_{ig} \log(p_g \mathcal{N}_q(x_i; \mu_g, \Sigma_g)). \tag{3.2}$$

with proper substitutions of mean and p, we get :

$$\begin{aligned} \ell(\theta|(x_1, z_1), \ldots, (x_n, z_n)) &= \sum_{g=1}^{G} q n_g \log \lambda_g + \sum_{g=1}^{G} \sum_{i=1}^{n} \frac{1}{\lambda_g} z_{ig} (x_i - \hat{\mu}_g)^t D_g^{-1} (x_i - \hat{\mu}_g) \\ &= \sum_{g=1}^{G} \{q\, n_g \log \lambda_g + \frac{1}{\lambda_g} \text{trace}\,(W_g D_g^{-1})\}, \end{aligned}$$

Final Results

Final Result is given as :

$$\lambda_g = \frac{1}{n_g} |\mathbf{diag}\,(W_g)|^{1/q}, \qquad D_g = \frac{\mathbf{diag}\,(W_g)}{|\mathbf{diag}\,(W_g)|^{1/q}}.$$

Initialization Step

- Using K-means and running EM algorithm after each merging step.
  - OR
- Using Agglomerative Hierarchical Clustering method and forming a tree structure.

- Note: we plan to use K-means as our initialization algorithm.

Likelihood based algorithm for merging

➔ After merging two clusters, the likelihood decreases by

$$\Delta(g_1, g_2) = q\,(n_{g_1} + n_{g_2}) \log \lambda_{\text{new}} + \frac{1}{\lambda_{\text{new}}} \text{trace}\,(W_{\text{new}} D_{\text{new}}^{-1})$$
$$- \left(q\, n_{g_1} \log \lambda_{g_1} + q\, n_{g_2} \log \lambda_{g_2} + \frac{1}{\lambda_{g_1}} \text{trace}\,(W_{g_1} D_{g_1}^{-1}) + \frac{1}{\lambda_{g_2}} \text{trace}\,(W_{g_2} D_{g_2}^{-1}) \right)$$
$$= q(n_{g_1} + n_{g_2}) \log \lambda_{\text{new}} - q n_{g_1} \log \lambda_{g_1} - q n_{g_2} \log \lambda_{g_2}$$

where the updating formulas are :

$$\hat{\mu}_{\text{new}} = \frac{n_{g_1}}{n_{g_1} + n_{g_2}} \hat{\mu}_{g_1} + \frac{n_{g_2}}{n_{g_1} + n_{g_2}} \hat{\mu}_{g_2},$$
$$W_{\text{new}} = W_{g_1} + W_{g_2} + n_{g_1} (\hat{\mu}_{\text{new}} - \hat{\mu}_{g_1})(\hat{\mu}_{\text{new}} - \hat{\mu}_{g_1})^t + n_{g_2} (\hat{\mu}_{\text{new}} - \hat{\mu}_{g_2})(\hat{\mu}_{\text{new}} - \hat{\mu}_{g_2})^t,$$
$$\lambda_{\text{new}} = \frac{1}{n_{g_1} + n_{g_2}} |\text{diag}\,(W_{\text{new}})|^{1/q}.$$

Functional merging algorithm

➔ It is the cosine between two gaussian mixture models $\cos(f_{g_1}, f_{g_2}) = \frac{\int f_{g_1} f_{g_2}}{\sqrt{\int f_{g_1}^2} \sqrt{\int f_{g_1}^2}}.$

In our case, it can be represented as follows and pair with maximum cosines are merged.

$$\frac{\lambda_{g_1}^{q/4} \lambda_{g_2}^{q/4}}{|\frac{1}{2}(\lambda_{g_1} D_{g_1} + \lambda_{g_2} D_{g_2})|^{1/2}} \exp\{-\frac{1}{2}(\mu_{g_1}^t \lambda_{g_1}^{-1} D_{g_1}^{-1} \mu_{g_1} + \mu_{g_2}^t \lambda_{g_2}^{-1} D_{g_2}^{-1} \mu_{g_2} - \mu_{\text{merge}}^t \lambda_{\text{merge}}^{-1} D_{\text{merge}}^{-1} \mu_{\text{merge}})\}$$

where

$$\lambda_{\text{merge}} = \lambda_{g_1} \lambda_{g_2} / |\lambda_{g_1} D_{g_1} + \lambda_{g_2} D_{g_2}|^{1/q}$$
$$D_{\text{merge}} = \{|\lambda_{g_1} D_{g_1} + \lambda_{g_2} D_{g_2}|^{1/q} (\lambda_{g_2} D_{g_1}^{-1} + \lambda_{g_1} D_{g_2}^{-1})\}^{-1}$$
$$\mu_{\text{merge}} = \lambda_{\text{merge}} D_{\text{merge}} (\lambda_{g_1}^{-1} D_{g_1}^{-1} \mu_{g_1} + \lambda_{g_2}^{-1} D_{g_2}^{-1} \mu_{g_2})$$

Workflow

- ➔ Feature vector representation.
- ➔ Reduction using PCA and latent semantic indexing.
- ➔ Applying GMDC
  - ◆ Initialization
  - ◆ EM likelihood merging and updation
  - ◆ Clustering formation and cross-checking using empirical measures
  - ◆ Classification
- ➔ Benchmarking the above process for all 15 combinations and analysis of the result.

Current Status

- ➔ Feature vector representation.
- ➔ Reduction using PCA.

Utility and future applications

- ➔ It finds its application in various fields of research for efficiently clustering documents.
- ➔ News Channels and other agencies such as Ads can retrieve knowledge and apply data analytics very easily.
- ➔ Easier way to store data in systematic manner even in your personal computers.

Thanks :)

Questions?