9. Topic Models, Nonnegative Matrix Factorization, Hidden Markov Models, and Graphical Models

Presented by: Jincheng Pang, Quan Xiao

2019.05.12

Focus

a. Topic Modeling

a.1. Topic Models
a.2. An Idealized Model
a.3. Nonnegative Matrix Factorization - NMF
a.4. NMF with Anchor Terms
a.5. Hard and Soft Clustering
a.6. The LDA Model for Topic Modeling
a.7. The Dominant Admixture Model
a.8. Formal Assumptions
a.9. Finding the Term-Topic Matrix
a.10. Hidden Markov Models

Topic Models

- Idea: There exist r ”topics”, that each given document is a mixture of these topics, which determine the probabilities of different words appearing in the document.
- Topic: A set of word frequencies.
  The word frequencies in document are convex combinations of word frequencies in topics.
- Document: A bag of words(terms).
  We disregard the order and context each word occurs in the document, only list the frequency of occurrences of each word.

Matrix Representation

- A: $d \times n$ *term − document* **matrix. One column per document and one row per term. Each document is a vector with d components where d is the total number of different terms; each component is the frequency of a particular term.**

- B: $d \times r$ *term − topic* **matrix. Each column of B is a vector corresponding to one topic; it is the vector of expected frequencies of terms in that topic.**

- C: $r \times n$ *topic − document* **matrix. Each vector with r nonnegative components summing to one is associated with each document, telling the fraction of the document that is on each of the topics.**

- P: $d \times n$ **matrix with column $P(:, j)$ denoting the expected frequencies of terms in document j. Then,**

$$P = BC$$

Document Generation Process

Topic Models generate documents according to the frequency matrix P above. p_{ij} is the probability that a random term of document j is the ith term in the dictionary. Assume that terms in a document are drawn independently. In general, assume B to be a fixed matrix, whereas C is random.
So, the process to generate n documents, each containing m terms, is the following:

Definition 1. *Let $\mathcal{D}$ be a distribution over a mixture of topics. Let B be the term-topic matrix. Create a $d \times n$ term-document matrix A as follows:*

1. *Initialize $a_{ij} = 0$ for $i = 1, 2, \dots, d; j = 1, 2, \dots, n$.*
2. *For $j = 1, 2, \dots, n$*
   (a) *Pick column j of C from distribution $\mathcal{D}$. This will be the topic mixture for document j, and induces $P(:, j) = BC(:, j)$.*
   (b) *For $t = 1, 2, \dots, m, do:$*
      i. *Generate the tth term x_t of document j from the multinomial distribution over $\{1, 2, \dots, d\}$ with probability vector $P(:, j)$ i.e., $\text{Prob}\,(x_t = i) = p_{ij}$.*
      ii. *Add $1/m$ to $a_{x_t,j}$.*

```
    iii. End for
  (c) End for
```

Often we are given fewer terms of each document than the number of terms. Even though

$$E\left(a_{ij}|P\right) = p_{ij},$$

the variance is high. For example, for the case when $p_{ij} = 1/d$ for all i with m much less than $\sqrt{d}$, $A(:, j)$ is likely to have $1/m$ in a random subset of m coordinates since no term is likely to be picked more than once. Thus,

$$\|A(:, j) - P(:j)\|_1 = m\left(\frac{1}{m} - \frac{1}{d}\right) + (d - m)\left(\frac{1}{d}\right) \approx 2$$

This says that in l_1 norm, which is the right norm when dealing with probability vectors, the ”noise” $\mathbf{a}_{\cdot,\mathbf{j}} - \mathbf{p}_{\cdot,\mathbf{j}}$ is likely to be larger than $\mathbf{p}_{\cdot,\mathbf{j}}$. Write

$$A = BC + N,$$

N stands for noise, which can have high norm. The l_1 norm of each column of N could be as high as that of BC.

Computational Difficulty

Two main ways of tackling the computational difficulty of finding B and C from A.

- **One is to make assumptions on the matrices B and C discussed in Section 9.2.**
- **The other way is to restrict N. An idealized way would be to assume $N = 0$ which leads to what is called the Non-negative Matrix Factorization(NMF)(Section 9.3) problem. With a further restriction on B called Anchor terms(Section 9.4), there is a polynomial time algorithm to do NMF.**
- **The most common approach to topic modeling makes an assumption on the probability distribution of C that the columns of C are independent Dirichlet distributed random vectors. This is called the Latent Dirichlet Allocation model(Section 9.6). We show that the Dirichlet distribution leads to many documents having a ”primary topic”, whose weight is much larger than average in the document.**

An Idealized Model

Assumptions:

- **Pure Topic Assumption: Each document is purely on a single topic. Each column j of C has a single entry equal to 1, and the rest of the entries are 0.**
- **Separability Assumption: The sets of terms occurring in different topics are disjoint. For each row i of B, there is a unique column l with $b_{il} \neq 0$.**

Under these assumptions, the data matrix A has a block structure. Let T_l denote the set of documents on topic l and S_l the set of terms occurring in topic l. Rearrange columns and rows so that A looks like:

$$A = \begin{pmatrix} * & * & * & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & * & * & * & 0 & 0 & 0 \\ 0 & 0 & 0 & * & * & * & 0 & 0 & 0 \\ 0 & 0 & 0 & * & * & * & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & 0 & 0 & * & * & * \\ 0. & 0 & 0 & 0 & 0 & 0 & * & * & * \end{pmatrix}$$

If we can partition the documents into r clusters, T_1 ,T_2 ,...,T_r, one for each topic, we can take the average of each cluster and that should be a good approximation to the corresponding column of B
Note that under the Pure Topics Assumption, the distribution $\mathcal{D}$ over columns of C is specified by the probability that we pick each topic to be the only topic of a document. Let $\alpha_1, \alpha_2, \ldots, \alpha_r$ be these probabilities.

Document Generation Process under Pure Topics Assumption

1. **Initialize all a_{ij} to zero.**
2. **For each document do:**
   (a) **Select a topic from the distribution given by $\{\alpha_1, \alpha_2, \ldots, \alpha_r\}$.**
   (b) **Select m words according to the distribution for the selected topic.**
   (c) **For each selected word add $1/m$ to the $document - term$ entry of the matrix A.**

Definition 2 (**Clustering Problem**). *Given A generated as above and the number of topics r, partition the documents $\{1, 2, \ldots, n\}$ into r clusters $T_1, T_2, \ldots, T_r$ each specified by a topic.*
Approximate Version*: Partition the documents into r clusters, where at most εn of the $j \in \{1, 2, \ldots, n\}$ are misclustered.*

How to solve the Clustering Problem?

We can find the term clusters S_l, which then can be used to solve the Clustering Problem.

Then, we can have an approximation to the matrix B.

Graph Method

- **Idea: Construct a graph G on d vertices, with one vertex per term, and put an edge between two vertices if they co-occur in any document.**
- **By separability assumption, there are no edges between vertices belonging to different S_l.**
- **Goal: For a specific topic l, figure out how many documents n_l we need so that with high probability, S_l is a connected component.**
- **One annoyance is that some words may have very low probability and not become connected to the rest of S_l . On the other hand, words of low probability can't cause much harm.**

Formal Argument

- **Setting: Let $\gamma < 1/3$ and define $\varepsilon = \gamma^m$.**
  Consider a partition of S_l into two subsets of terms W and $\overline{W}$ that each have probability mass at least γ in the distribution of terms in topic l.

Claim 1. *For every such partition, if there is at least one edge between W and $\overline{W}$, the largest connected component $\hat{S}_l$ in S_l must have probability mass at least $1 - \gamma$.*

Since $\text{Prob}(\hat{S}_l) \geq 1 - \gamma$, the probability that a new random document of topic l contains only words not in $\hat{S}_l$ is at most $\gamma^m = \varepsilon$.
To prove the statement about partitions, note that the probability that m words are all in W or $\overline{W}$ is at most $\text{Prob}(W)^m + \text{Prob}(\overline{W})^m$.
Thus the probability that none of n_l documents creates an edge between W and $\overline{W}$ is

$$\begin{aligned}\left(\text{Prob}(W)^m + \text{Prob}(\overline{W})^m\right)^{n_l} &\leq (\gamma^m + (1-\gamma)^m)^{n_l} \\ &\leq ((1-\gamma/2)^m)^{n_l} \\ &\leq e^{-\gamma m n_l/2}\end{aligned}$$

Since there are at most 2^d different possible partitions of S_l into W and $\overline{W}$, the union bound ensures at most a δ probability of failure by having

$$2^d e^{-\gamma m n_l/2} \leq \delta,$$

in turn

$$mn_l \geq \frac{2}{\gamma}\left(d\ln 2 + \ln\frac{1}{\delta}\right)$$

Lemma 1. *If* $mn_l \geq \frac{2}{\gamma}\left(d\ln 2 + \ln\frac{1}{\delta}\right)$*, then with probability at least* $1-\delta$*, the largest connected component in* S_l *has probability mass at least* $1-\gamma$*. This in turn implies that the probability to fail to correctly cluster a new random document of topic* l *is at most* $\varepsilon = \gamma^{1/m}$.

Nonnegative Matrix Factorization - NMF

Write

$$A = BC + N$$

N stands for noise. In topic modeling, N can be high. Let's first look at the problem when there is no noise, which can be thought of as the limiting case as the number of words per document goes to infinity.

NMF Problem

Exact equations

$$A = BC$$

A is the given matrix with non-negative entries and all column sums equal to 1. Given the number of topics r, can we find B and C such that $A = BC$ where B and C have nonnegative entries?
In topic modeling, we have additional constraints:

1. **$A = BC$.**
2. **The entries of B and C are all nonnegative.**
3. **Columns of both B and C sums to one.**

Lemma 2. *Let A be a matrix with nonnegaitve elements and columns summing to one. The problem of finding a factorization BC of A satisfying the three conditions above is reducible to the NMF problem of finding a factorization BC satisfying conditions* (1) *and* (2)*.*

The NMF problem as formulated above is a nonlinear problem in $r(n+d)$ unknowns(the entries of B and C).

Lemma 3. *If A has rank r, then the NMF problem can be formulated as a problem with $2r^2$ unknowns. Using this, the exact NMF problem can be solved in polynomial time if r is constant.*

Proof. From $A = BC$, we know that each row of A is a linear combination of the rows of C. So the space spanned by the rows of A is contained in the space spanned by the rows of the $r \times n$ matrix C. The latter space has dimension at most r, while the former has dimension r by assumption. So they must be equal. Thus every row of C must be a linear combination of the rows of A. Choose any set of r independent rows of A to form a $r \times n$ matrix A_1. Then $C = SA_1$ for some $r \times r$ matrix S. By analogous reasoning, if A_2 is a $d \times r$ matrix of r independent columns of A, there is a $r \times r$ matrix T such that $B = A_2 T$. Now we can easily cast NMF in terms of unknowns S and T:

$$A = A_2 T S A_1 \quad ; \quad (SA_1)_{ij} \geq 0 \quad ; \quad (A_2 T)_{kl} \geq 0 \quad \forall i, j, k, l$$

It remains to solve the nonlinear problem in $2r^2$ variables. ■

NMF with Anchor Terms

Anchor Terms

- **An anchor term for a topic is a term that occurs in the topic and does not occur in any other topic.**
- **Assumption: Each topic has an anchor term.**
- **This assumption is weaker than the separability assumption of Section 9.2, which says that all terms are anchor terms.**

Definition 3 (**Anchor Term**). *For each $l = 1, 2, \ldots r$, there is an index i_l such that*

$$b_{i,l} \neq 0 \quad \textit{and} \quad \forall l' \neq l \quad b_{i_l,l'} = 0.$$

It is easy to see that each row of the $topic - document$ **matrix** C **has a scalar multiple of it occurring as a row of the given** $term - document$ **matrix** A**.**

$$\begin{pmatrix} 0.3 \times \mathrm{c}_4 \\ \\ A \\ \\ 0.2 \times \mathrm{c}_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0.3 \\ & & & \\ & & B & \\ & & & \\ 0 & 0.2 & 0 & 0 \end{pmatrix} \begin{pmatrix} \leftarrow \mathbf{c}_1 \rightarrow \\ \leftarrow \mathbf{c}_2 \rightarrow \\ \\ \leftarrow \mathbf{c}_4 \rightarrow \end{pmatrix}$$

In fact, we have

$$\mathbf{a}_{i\cdot} = \sum_{k=1}^{r} b_{ik}\mathbf{c}_{i\cdot}$$

Claim 2. *If there is a NMF of A, there is one in which no row of C is a nonnegative linear combination of other rows of C.*

Proof. If some row of C is a nonnegative linear combination of the other rows of C, then eliminate that row of C as well as the corresponding column of B and suitably modify the other columns of B to maintain $A = BC$. ■

Claim 3. *If $A = BC$ is a NMF of A and there are rows in A that are positive linear combinations of other rows, the rows can be remove and the corresponding rows of B remove to give a NMF $\hat{A} = \hat{B}C$ where $\hat{A}$ and $\hat{B}$ are the matrices A and B with the removed rows. Similar as above.*

Claim 4. *$\hat{B}$ is a diagonal matrix.*

Proof. Since no row of C is a nonnegative linear combination of other rows of C, the r rows of C can be regarded as base vectors. Recall $\mathbf{a}_{i\cdot} = \sum_{k=1}^{r} b_{ik}\mathbf{c}_{i\cdot}$ Because of the existence of anchor terms, there are at least r rows of A to be the scalar multiples of rows of C. If $\hat{B}$ is not a diagonal matrix, we pick the row of B which has more than one nonzero elements. The corresponding row of $\hat{A}$ must be the linear combinations of other rows of $\hat{A}$. Thus, $\hat{B}$ is a diagonal matrix and the rows of A are scalar multiples of rows of C. ■

Algorithm 1. *First we have $d \times n$ term-document matrix A.*

1. *Obtain $\hat{A}$ by removing the rows from A that are positive linear combinations of other rows.*
2. *Let $C = \hat{A}$ and $\hat{B} = I$. We have $\hat{A} = I\hat{A}$ as $\hat{A} = \hat{B}C$.*
3. *Restore the rows to $\hat{B}$ to get B such that $A = BC$.*

To remove rows of A that are positive linear combinations of other rows in polynomial time, check if there are real numbers $x_1, x_2, \dots x_{i-1}, x_{i+1}, \dots x_n$ such that

$$\sum_{j \neq i} x_j \mathbf{a_j} = \mathbf{a_i} \quad x_j \geq 0$$

This is a linear program and can be solved in polynomial time.

Hard and Soft Clustering

Hard Clustering

When each document is purely on one topic and each term occurs in only one topic, approximately finding B was reducible to clustering documents according to their topic. We call this *hard clustering*, meaning each data point is to be assigned to a single cluster.

Soft Clustering

When each document has a mixture of several topics, we may still view each topic as a cluster and each topic vector, i.e., each column of B, as a ”cluster center”. We may then view $P(:,j) = BC(:,j)$ as the ”cluster center” for document j. The document vector $A(:,j)$ is its cluster center plus a noise $N(:,j)$.

The Latent Dirichlet Allocation Model for Topic Modeling

Introduction

- **In this model, the topic weight vectors of the documents, the columns of C, are picked independently from Dirichlet distribution.**
- **The $term - topic$ matrix B is fixed.**

The Dirichlet distribution has a parameter μ called the "concentration parameter", which is a real number in $(0, 1)$, typically set to $1/r$.
For each vector v with r nonnegative components summing to one,

$$Prob\ density\ (column\ j\ of\ C = \mathbf{v}) = \frac{1}{g(\mu)} \prod_{l=1}^{r} v_l^{\mu-1},$$

where $g(\mu)$ is the normalizing constant so that the total probability mass is one.
Once C is generated, the LDA model hypothesizes that the matrix

$$P = BC$$

acts as the probability matrix for the data matrix A, namely,

$$E(A|P) = P$$

Thus, A can be generated.

Some properties of the Dirichlet distribution

Lemma 4. *Suppose the joint distribution of* $\mathbf{y} = (y_1, y_2, \ldots, y_r)$ *is the Dirichlet distribution with concentration parameter* μ. *Then, the marginal probability density* $q(y)$ *of* y_1 *is given by*

$$q(y) = \frac{\Gamma(r\mu + 1)}{\Gamma(\mu)\Gamma((r-1)\mu + 1)} y^{\mu-1}(1-y)^{(r-1)\mu}, \mu \in (0, 1]$$

where, Γ *is the Gamma function.*

Proof. By definition of the marginal,

$$q(y) = \frac{1}{g(\mu)} y^{\mu-1} \int_{y_2+y_3+\cdots+y_r=1-y} (y_2 y_3 \cdots y_r)^{\mu-1}\, dy_2 dy_3 \ldots dy_r.$$

Put $z_l = y_l/(1-y)$. With this change of variables,

$$q(y) = \frac{1}{g(\mu)} y^{\mu-1}(1-y)^{(r-1)\mu} \left(\int_{z_2+z_3+\cdots+z_r=1} (z_2 z_3 \cdots z_r)^{\mu-1}\, dz_2 dz_3 \ldots dz_r \right)$$

The quantity inside the parentheses is independent of y, so for some c we have

$$q(y) = c y^{\mu-1}(1-y)^{(r-1)\mu}$$

Since $\int_0^1 q(y) dy = 1$, we must have

$$c = \frac{1}{\int_0^1 y^{\mu-1}(1-y)^{(r-1)\mu}} = \frac{\Gamma(r\mu + 1)}{\Gamma(\mu)\Gamma((r-1)\mu + 1)}$$

■

Lemma 5. *Suppose the joint distribution of* $\mathbf{y} = (y_1, y_2, \ldots, y_r)$ *is the Dirichlet distribution with concentration parameter* $\mu \in (0, 1)$*. For* $\zeta \in (0, 1)$*,*

$$\text{Prob}\left(y_1 \geq 1 - \zeta\right) \geq \frac{0.85\mu\zeta^{(r-1)\mu+1}}{(r-1)\mu + 1}$$

Hence for $\mu = 1/r$*, we have* $\text{Prob}\left(y_1 \geq 1 - \zeta\right) \geq 0.4\zeta^2/r$*. If also,* $\zeta < 0.5$*, then,*

$$\text{Prob}\left(Max_{l=1}^{r} y_l \geq 1 - \zeta\right) \geq 0.4\zeta^2$$

Proof. Since $\mu < 1$, we have $y^{\mu-1} > 1$ for $y < 1$ and so $q(y) \geq c(1 - y)^{(r-1)\mu}$. Thus,

$$\int_{1-\zeta}^{1} q(y)dy \geq \frac{c}{(r-1)\mu + 1}\zeta^{(r-1)\mu+1} = \frac{\Gamma(r\mu + 1)}{\Gamma(\mu)\Gamma((r-1)\mu + 1)} \frac{\zeta^{(r-1)\mu+1}}{(r-1)\mu + 1}$$

Note that $\Gamma(\mu) \leq 1/\mu$ for $\mu \in (0, 1)$. Since $r \geq 1$ and $\mu \in (0, 1)$, we have $(r - 1)\mu + 1 \geq 1$. Also, $\Gamma(x)$ is an increasing function for $x \geq 1.5$, so if $(r - 1)\mu + 1 \geq 1.5$, $\Gamma(r\mu + 1) \geq \Gamma((r - 1)\mu + 1)$ and in this case, the first assertion of the lemma follows. If $(r - 1)\mu + 1 \in [1, 1.5]$, then $\Gamma((r - 1)\mu + 1) \leq 1$. Since $r\mu + 1 \in [1, 2.5]$, $\Gamma(r\mu + 1) \geq \min_{z\in[1,2.5]} \Gamma(z) \geq 0.85$. Again, the first assertion follows.
The second assertion of the lemma follows easily. For the third assertion, note that $y_l > 1 - \zeta$, $l = 1, 2, \ldots, r$ are mutually exclusive events for $\zeta < 0.5$, so $\text{Prob}\left(\max_{l=1}^{r} y_l \geq 1 - \zeta\right) = \sum_{l=1}^{r} \text{Prob}\left(y_l > 1 - \zeta\right) = r\,\text{Prob}\left(y_1 \geq 1 - \zeta\right) \geq 0.4\zeta^2$. ■

From the last lemma, it follows that

- **With high probabilty, a constant fraction of the documents have a primary topic of weight at least 0.6 (let $\zeta = 0.4$ in the third assertion).**
- **If the total number of documents, n, is large, there will be many nearly pure documents. If we could find the nearly pure documents for a given topic l, then the average of the A columns corresponding to these documents will be close to the average of those columns in the matrix BC.**

More generally, the concentration parameter may be different for different topics. We then have $\mu_1, \mu_2, \ldots, \mu_r$ so that

$$Prob\ density\ (column\ j\ of\ C = \mathbf{v}) \propto \prod_{l=1}^{r} v_l^{\mu_l - 1}$$

The Dominant Admixture Model

- **Notation: data matrix $A \in \mathbb{R}^{d\times n}$, topic-term matrix $B \in \mathbb{R}^{d\times r}$, and document-topic matrix $B \in \mathbb{R}^{r\times n}$.**
- **Aim: primary topic classification, i.e. to find the primary topic of each document.**
- **Assumptions:**
  - **Primary Topic: Each document has a primary topic.**
  - **Pure Document: Each topic has at least one pure document that is mostly on that topic.**
  - **Catchword: Each topic has at least one catchword.**

The Dominant Admixture Model

Why these assumptions?

- **Intuition:**
  - **Obtain document-term frequency matrix P from data matrix A.** **We will assume c larger than $\Omega(\log(nd))$ and $p_{ij} \geq c/m$ to have $P \approx A$.**
  - **obtain document-topic matrix C from P and B.** **Let $l(i) = \arg\max_{l'=1}^{r} b_{il'}$ and T_l be the set of j with primary topic l. We will prove that if i is a catchword for topic l, that there is a sharp drop in p_{ij} between $j \in T_l$ and $j \notin T_l$. Moreover, whether or not i is a catchword, the primary topic assumption will imply that p_{ij} does not drop by more than a certain factor α among $j \in T_{l(i)}$.**

Formal Assumptions

Parameters: α, β, ρ **and** δ **are real numbers in** $(0, 0.4]$ **satisfying**

$$\beta + \rho \leq (1 - 3\delta)\alpha$$

- **Primary Topic: There is a partition of** $[n]$ **into** $T_1, T_2, \ldots, T_k$ **with:**

$$c_{lj} \begin{cases} \geq \alpha, & \textbf{for } j \in T_l, \\ \leq \beta, & \textbf{for } j \notin T_l. \end{cases}$$

- **Pure Document: For each** l**, there is some** j **with:**

$$c_{lj} \geq 1 - \delta$$

- **Catchwords: For each** l**, there is at least one catchword** i **satisfying:**

$$b_{il'} \leq \rho b_{il}, \qquad \textbf{for } l' \neq l$$

$$b_{il} \geq \mu, \qquad \textbf{where } \mu = \tfrac{c \log(10nd/\delta)}{m\alpha^2\delta^2}, c \textbf{ constant.}$$

Formal Assumptions

Why $a_{ij} \approx p_{ij}$?

Lemma 6. $\text{Prob}\left(|a_{ij} - p_{ij}| \geq \delta\alpha\,\text{Max}\left(p_{ij}, \mu\right)/4\right) \leq \frac{\delta}{10nd}$
So with probability at least $1 - (\delta/10)$,

$$|a_{ij} - p_{ij}| \leq \delta\alpha\,\text{Max}\left(\mu, p_{ij}\right)/4 \quad \forall i, j$$

simultaneously. After paying the failure probability of $\delta/10$, *we henceforth assume that the above holds.*

Proof. Since a_{ij} is the average of m independent Bernoulli trials, each with expectation p_{ij}, by using Hoeffding-Chernoff inequality

$$\text{Prob}\left(|a_{ij} - p_{ij}| \geq \Delta\right) \leq 2\exp\left(-\,\text{cmMin}\left(\frac{\Delta^2}{p_{ij}}, \Delta\right)\right)$$

and plugging in $\Delta = \alpha\delta\,\text{Max}\left(p_{ij}, \mu\right)/4$, we can arrive at the first statement of the lemma. Then the second statement is proved by a union bound over the nd possible (i, j) values. ■

Formal Assumptions

Algorithm:

1. **Compute Thresholds:** $\mu_i = \alpha(1-\delta)\max_j a_{ij}$.
2. **Do thresholding: Define a matrix $\hat{A}$ by**

$$\hat{a}_{ij} = \begin{cases} 1, & \textbf{if } a_{ij} \geq \mu_i \textbf{ and } \mu_i \geq \mu\alpha\left(1 - \frac{5\delta}{2}\right) \\ 0, & \textbf{otherwise}. \end{cases}$$

3. **Pruning: Let $R_i = \{j | \hat{a}_{ij} = 1\}$. If any R_j strictly ocntains another, set all entries of row i of $\hat{A}$ to zero.**

Why it works?

Theorem 1. *For $i = 1, 2, \ldots, d$, let $R_i = \{j | \hat{a}_{ij} = 1\}$ at the end of the algorithm. Then, each nonempty $R_i = T_{l(i)}$, with $l(i) = \arg\max_{l'=1}^{r} b_{il'}$.*

Lemma 7. *If i is a catchword for topic l, then $R_i = T_l$.*

Formal Assumptions

Why Lemma 2. works?

Proof. The proof consists of three claims.

Claim 5. *For i, a catchword for ropic l,*

$$b_{il} \geq p_{ij} \geq b_{il}\alpha \textit{ for } j \in T_l,$$

$$p_{ij} \leq b_{il}\alpha(1-3\delta) \textit{ for } j \notin T_l.$$

Claim 6. *With probability at least $1-\delta/10$, for every l and every catchword i of l:*

$$a_{lj} \begin{cases} \geq \alpha, & \textit{for } j \in T_l, \\ \leq \beta, & \textit{for } j \notin T_l. \end{cases}$$

Claim 7. *With probability at least $1-\delta$, for every topic l and every catchword i of topic l, the μ_i computed in step 1 of the algorithm satisfies:*

$$\mu_i \in ((1-(5/2)\delta)b_{il}\alpha, b_{il}\alpha(1-(\delta/2))).$$

■

Remark 1. *The first claim states that for $j \in T_l$, p_{ij} is high and for $j \notin T_l$, p_{ij} is low. The second argues that the same for a_{ij} instead of p_{ij}. The third claim shows that the threshold computed in the first step of the algorithm falls between the high and the low.*

Formal Assumptions

Why Theorem 1. works?

Proof. The Lemma 2. proves the catchwords circumstance. Then for **noncatchword situation**, let $a = \max_j a_{ij}$. WLOG. assume that $a \geq \mu(1-(5\delta/2))$. Let $j_0 = \arg\max_j a_{ij}$. Then we claim $p_{i,j_0} \geq a(1-\delta/2)$. Let $l = l(i)$. Then

$$a(1-\delta/2) \leq p_{ij_0} = \sum_{l'=1}^{r} b_{il'}c_{l'j_0} \leq b_{il}$$

Also, if j_1 is a pure document for topic l, we can prove that $p_{i,j_1} \geq b_{il}c_{l,j_1} \geq b_{il}(1-\delta)$. Now, we claim that

$$a_{i,j_1} \geq b_{il}(1-(3\delta/2))$$

Thus, $a \geq b_{il}(1-(3\delta/2))$. Now, for all $j \in T_l$, $p_{ij} \geq b_{il}c_{lj} \geq a(1-\delta/2)\alpha$. Then, with Lemma 1., we know that for all $j \in T_l$, $a_{ij} \geq a(1-\delta)\alpha$. By step 1 of the algorithm, $\mu_i = a(1-\delta)\alpha$, so $\mu_i = a(1-\delta)\alpha$ for all $j \in T_l$. So either $R_i = T_l$ or $T_l \subsetneqq R_i$. In the latter case, the pruning step will set $\hat{(a)}_{ij} = 0$ for all j, since topic l has some catchword i_0 for which $R_{i_0} = T_l$ by Lemma 2. ■

Finding the Term-Topic Matrix

- **Aim: to find the term-topic matrix B.**
- **Assumption (4): Set of Documents. For each l, there is a set W_l of at least δn documents with**

$$c_{lj} \geq 1 - \tfrac{\delta}{4}, \quad \forall j \in W_l$$

This assumption makes sure that a good estimate of B can be found since the corresponding columns of A are independent even conditioned on P.

- **Motivation: If we could find the set of pure documents for each topic with possibly a somall fraction of errors, we could average them. We use the primary topic classification in the last section for this. We also know that for the catchword i of topic l, the maximum value of p_{ij} occurs for a pure document and indeed if the assumption above holds, the set of $\delta n/4$ documents with the top $\delta n/4$ values of p_{ij} should be all pure documents. But we should discover catchwords first.**
- **Property: $\delta n/4^{th}$ maximum value among p_{ij}, $j \in T_l$ is substantially higher than the $\delta n/4^{th}$ maximum value among p_{ij}, $j \in T_{l'}$ for any $l' \neq l$.**

Finding the Term-Topic Matrix

Algorithm:

Output: B.

1. **Init: Compute T_l, $l = 1, 2, \ldots, r$ by primary topic classification.**
2. **For $l = 1, 2, \ldots, r$ and for $i = 1, 2, \ldots, d$, let $g(i, l)$ be the $(1 - \delta n/4)^{th}$ fractile of $\{A_{ij} : j \in T_l\}$.**
3. **For each l, chose an $i(l)$ such that:**

$$g(i(l), l) \geq (1 - (\delta/2))\mu; \quad g\left(i(l), l'\right) \leq (1 - 2\delta)\alpha g(i(l), l) \quad \forall l' \neq l. \tag{1}$$

4. **Let R_l be the set of $\delta n/4$ j's among $j = 1, 2, \ldots, n$ with the highest $A_{i(l),j}$.**
5. **Return $\tilde{B}_{.,l} = \frac{1}{|R_l|} \sum_{j \in R_l} A_{.,j}$ as our approximation to $B_{.,l}$.**
6. **End for.**

Finding the Term-Topic Matrix

Why This Algorithm works?

Lemma 8. *$i(l)$ satisfying (1) exists for each l.*

Proof. Let i be a catchword for l. Then since $\forall j \in W_l$, $p_{ij} \geq b_{il}c_{lj} \geq b_{il}(1-(\delta/4))$ and $b_{il} \geq \mu$, we have $a_{ij} \geq (1-(\delta/2))$ and so $g(i,l) \geq (1-(\delta/2))b_{il} \geq (1-(\delta/2))\mu$ by Lemma 1. For $j \notin T_l$, $a_{ij} \leq b_{il}\alpha(1-(5\alpha/2))$ by Claim 2 and so $g\left(i,l'\right) \leq b_{il}\alpha(1-(5\delta/2))$. So $g(i,l)$ statisfies both the requirements of step 2 of the algorithm. ∎

Fixed l, let $i = i(l)$ and $\rho_i = \max_{k=1} b_{ik}$.

Lemma 9. $\rho_i \geq \left(1-\frac{3}{4}\delta\right)\mu.$

Lemma 10. $b_{ik} \leq \alpha b_{il}, \quad \forall k \neq l$

Lemma 11. *For each $j \in R_i$ of step 3 of the algorithm, we have $c_{lj} \geq 1-2\delta$.*

Theorem 2. *Assume $n \geq \frac{cd}{m\delta^3}; m \geq \frac{c}{\delta^2}$. For all $l, 1 \leq l \leq r$, the $\hat{b}_{\cdot,l}$ returned by step 4 of the Algorithm satisfies $\left\|b_{\cdot,l} - \hat{b}_{\cdot,1}\right\|_1 \leq 6\delta$.*

Hidden Markov Models (HMM)

- **Introduction: A HMM consists of a finite set of states with a transition between each pair of states.**
- **Notation: α, an initial probability distribution on the states; a_{ij}, a transition probability associated with the transition from state i to state j; $p(O, i)$, the probability of outputting the symbol O in state i.**
- **Result: A state transition to a new state followed by the output of a symbol. The HMM starts by selecting a start state according to the distribution α and outputting a symbol.**
- **Problems:**
  - **How probable is an output sequence?**
  - **The most likely sequence of states - the Viterbi algorithm**
  - **Determining the underlying hidden Markov model**