



NICTA

An Approach to Discrete Component Analysis DCA v0.200 Theory Companion

NICTA Technical Report NICTA-SML-09-002
Statistical Machine Learning Group, Canberra
July 2009

Wray Buntine

NICTA and Australian National University
Locked Bag 8001, Canberra, 2601, ACT Australia
wray.buntine@nicta.com.au,
HIIT, Helsinki, Finland

Abstract

This report is the background theory for Discrete Component Analysis software called DCA. Currently the software is run in stand-alone mode, and scavenges data streaming libraries and Dirichlet utilities from the older *MPCA* system¹. The software itself is written in the C language and compiles on a Linux and a Mac OS X environment. The models presented here are a hierarchical extension of discrete component analysis. This is known under many names (2), such as LDA, multi-aspect models, multinomial PCA, etc.

1. <http://www.componentanalysis.org>

Contents

1	Introduction	1
2	Notation	1
2.1	Documents	1
2.2	Topics	1
3	Model	1
3.1	Basic Models	2
3.2	Alternative Interpretations	2
3.3	Related Work	3
4	Analysis	3
4.1	Likelihoods	4
4.1.1	The fixed structure case	4
4.1.2	The per-document structure case	5
4.1.3	Using Gamma-Poisson models	5
4.2	An approximation	5
5	Algorithms	6
5.1	Gibbs Sampling for Fixed Structure	6
5.2	Gibbs Sampling for per-Document Structure	6
5.3	Gibbs Sampling for Conditional Gamma-Poisson	6
5.4	Hill Climbing in Structure Space	7
5.5	Estimating Hyper-parameters for Component Priors	7
5.5.1	Dirichlet multinomial components	7
5.5.2	Gamma Poisson components	7
5.5.3	Conditional Gamma Poisson components	8
5.6	Estimating Hyper-parameters for per-Document Structure Priors	9
6	Identifiability in the Hierarchical Case	9
A	Proof of convergence for Dirichlet hyper-parameter estimation	9

1. Introduction

This report is the background theory for Discrete Component Analysis software called DCA. Currently the software is run in stand-alone mode, and scavengers data streaming libraries and Dirichlet utilities from the older *MPCA* system². The software itself is written in the C language and compiles on a Linux and a Mac OS X environment. No Windows compilation has been attempted.

The models presented here are a hierarchical extension of discrete component analysis. This is known under many names [2], such as LDA, multi-aspect models, multinomial PCA, etc. The hierarchical extension corresponds most closely to the hierarchical Pachinko allocation models of [4].

2. Notation

This section describes the notation used, both in this report, and in the software. Note that variables and such are, for the most part, respected by the software. Thus the matrix $\vec{\theta}$ below has the variable name `theta` in the text, and the sufficient statistics \vec{N} and \vec{D} have the variable names `dataN` and `dataD`. Only the dimensions, for instance I and J have mnemonic names such as *documents* and *features*.

2.1 Documents

In our data reduction approach, one has I documents numbered $i = 0, \dots, I - 1$. Each document has L_i terms, and terms are numbered $l = 0, \dots, L_i - 1$ for a document i , and are indexed (into the dictionary) with j_l . Assume the dictionary has J entries numbered $0, \dots, J - 1$.

Thus the data we have can be represented as a set of integers in the form $\vec{j}_i = \{j_{i,l} : l = 0, \dots, L_i - 1\}$, with the full set as

$$\{ \{ j_{i,l} : l = 0, \dots, L_i - 1 \} : i = 0, \dots, I - 1 \} .$$

2.2 Topics

Now we will associate the terms with K topics/aspects/components, and give each document a vector of propensities for seeing the topics, represented as a K -dimensional probability vector \vec{q}_i .

We will assign to each term indexed by (i, l) a hidden topic (also called aspect or component) denoted $c_{i,l} \in \{0, \dots, K - 1\}$. This gives the leaf topic the term belongs to. This is modelled as a latent variable.

Associated with this, is a hierarchical topic $h_{i,l} \in \{0, \dots, H - 1\}$, for $H > K$. The topics form the leaf nodes of the hierarchy, and for convenience hierarchical topics are numbered $0, \dots, K - 1$ if they are at the leaf. Note the hierarchical topic is the real objective of the modelling, and the leaf topic is a device to introduce correlations between positions in the hierarchy.

The probability of being at a given hierarchical topic h , given the leaf topic c is:

$$p(h | c, \vec{\Upsilon}) = v_{c,h} .$$

We can assume $\vec{\Upsilon}$ (which has entries $v_{c,h}$) is a constant, or we can include it in the fitting as well.

The topic model is then defined by the $K \times H$ dimensional matrix $\vec{\Upsilon}$ with K row vectors \vec{v}_c for $c = 1, \dots, K - 1$. This should be very regular in form and sparse when the hierarchical structure is strictly tree shaped.

One way to regularise the topic model is give each node a fixed probability of being chosen, regardless of which leaf was the source. By this model, $v_{c,h}$ is independent of c . One can reparameterise the model into the form

$$p(h | \text{ancestors of } h \text{ not chosen}, \vec{\phi}) = \phi_h .$$

Then

$$v_{c,h} = \phi_h \prod_{h' \in \text{ancs}(h)} (1 - \phi_{h'}) . \quad (1)$$

3. Model

We first present the basic model. The model itself can be viewed as a variation of standard discrete component analysis (a discrete version of PCA known under many names [2]). In this case, it can be interpreted as a correlated version of component analysis where independence no longer holds.

2. <http://www.componentanalysis.org>

3.1 Basic Models

A model with *fixed structure* is first given. The full probability for a document is given by a product of generative probabilities for each document for the vectors of hidden variables $\vec{q}_i, \vec{c}_i, \vec{h}_i$ and the data vector \vec{j}_i . For each document i , the distribution is:

$$\begin{aligned} \vec{q}_i &\sim \text{Dirichlet}_K(\vec{\alpha}) , \\ c_{i,l} &\sim \text{Discrete}_K(\vec{q}_i) && \text{for } l = 0, \dots, L_i - 1 , \\ h_{i,l} &\sim \text{Discrete}_H(\vec{v}_{c_{i,l}}) && \text{for } l = 0, \dots, L_i - 1 , \\ j_{i,l} &\sim \text{Discrete}_J(\vec{\theta}_{h_{i,l}}) && \text{for } l = 0, \dots, L_i - 1 . \end{aligned}$$

Here, $\vec{v}_{c_{i,l}}$ is a vector, which itself is taken as the $c_{i,l}$ -th column vector of the matrix $\vec{\Upsilon}$. In general, parameter vectors are in lower case Greek, and parameter matrices are upper case Greek. Note the component indicators \vec{c}_i can be aggregated in total counts per class, to become a K -dimensional counts vector. This aggregate corresponds to the score matrix in conventional PCA. The parameter matrix $\vec{\Theta}$ corresponds to the loading matrix in conventional PCA.

Also, a prior is required for some of the $\vec{\Theta}$ parameters:

$$\vec{\theta}_h \sim \text{Dirichlet}_J(\vec{\gamma}) \quad \text{for } h = 0, \dots, H + K - 1 .$$

Two versions of the matrix $\vec{\Upsilon}$ are used, one derived from a simpler node probability.

A second more complex model has *document specific structure*. In this case the $\vec{\Upsilon}$ matrix is not fixed across documents but generated with a vector of Dirichlet distributions. So

$$\vec{v}_c \sim \text{Dirichlet}_K(\vec{\psi}_c) \quad \text{for } c = 0, \dots, K - 1 .$$

Here the model parameters and hyper-parameters are

$\vec{\alpha}$: A K dimensional vector of Dirichlet parameters generating topic probabilities at the leaves, assumed fixed in first versions.

$\vec{\Upsilon}$: A $K \times H$ dimensional matrix defines relationships between leaves and hierarchical topic nodes, assumed fixed in first versions. Column vectors \vec{v}_c give the distribution of hierarchical topics for leaf c .

$\vec{\psi}$: A H dimensional vector defines the prior for $\vec{\Upsilon}$ when it is fixed per document.

$\vec{\Psi}$: A $K \times H$ dimensional matrix defines the prior for $\vec{\Upsilon}$ when it changes per document. Column vectors $\vec{\psi}_c$ give the Dirichler parameters for \vec{v}_c .

$\vec{\Theta}$: A $H \times J$ dimensional matrix defines term probabilities for each hierarchical node, with column vectors $\vec{\theta}_h$ giving them for hierarchical topic h . This is the loading matrix in conventional PCA.

$\vec{\gamma}$: A J dimensional vector defines the prior for row vectors $\vec{\theta}_h$.

3.2 Alternative Interpretations

Different interpretations can be made of the above model.

One can marginalise out the $h_{i,l}$ variables, the distributions on terms becomes dependent on \vec{c}_i instead of \vec{h}_i .

$$\begin{aligned} \vec{q}_i &\sim \text{Dirichlet}_K(\vec{\alpha}) , \\ c_{i,l} &\sim \text{Discrete}_K(\vec{q}_i) && \text{for } l = 0, \dots, L_i - 1 , \\ j_{i,l} &\sim \text{Discrete}_J((\vec{\Upsilon}\vec{\Theta})_{c_{i,l}}) && \text{for } l = 0, \dots, L_i - 1 . \end{aligned}$$

This is the same form as standard discrete component analysis, excepting the prior on the $\Upsilon\Theta$ term is now complex, instead of a simple Dirichlet. Depending on the form of Υ , this causes a strong correlation between some of the components instead of full independence.

Alternatively, one can marginalise out \vec{c}_i , so that

$$\begin{aligned} \vec{q}_i &\sim \text{Dirichlet}_K(\vec{\alpha}) , \\ h_{i,l} &\sim \text{Discrete}_H(\vec{q}_i\Upsilon) && \text{for } l = 0, \dots, L_i - 1 , \\ j_{i,l} &\sim \text{Discrete}_J(\vec{\Theta}_{h_{i,l}}) && \text{for } l = 0, \dots, L_i - 1 . \end{aligned}$$

In this case, it again looks like standard discrete component analysis, however, now, the correlation exists between the component probabilities due to $\vec{q}_i \Upsilon$.

Both these forms are similar in form to the simplest kind of independent discrete component analysis:

$$\begin{aligned} \vec{q}_i &\sim \text{Dirichlet}_K(\vec{\alpha}) , \\ c_{i,l} &\sim \text{Discrete}_K(\vec{q}_i) && \text{for } l = 0, \dots, L_i - 1 , \\ j_{i,l} &\sim \text{Discrete}_J(\vec{\Theta}_{c_{i,l}}) && \text{for } l = 0, \dots, L_i - 1 . \end{aligned}$$

which can be compressed to

$$\begin{aligned} \vec{q}_i &\sim \text{Dirichlet}_K(\vec{\alpha}) , \\ j_{i,l} &\sim \text{Discrete}_J(\vec{q}_i^T \vec{\Theta}) && \text{for } l = 0, \dots, L_i - 1 . \end{aligned}$$

However, in the first alternative case with $(\Upsilon \vec{\Theta})$, correlations are induced on the loading matrix, $\vec{\Theta}$ term, and in the second alternative case with $\vec{q}_i \Upsilon$ correlations are induced on the score matrix, through the prior \vec{q}_i term. In each case, correlations are represented by Υ . Thus we see there are two equivalent interpretations that add correlation to the standard independent discrete component analysis.

Note, we can also bag up the counts, so that instead of keeping word and topic indexes for each of $l = 0, \dots, L_i - 1$ words, we keep counts $w_{i,j}$ for each word index j , so that

$$w_{i,j} = \sum_{l=0, \dots, L_i-1} 1_{j_{i,l}=j} ,$$

yielding a J -dimensional vector \vec{w}_i . Then discrete component analysis can also be expressed as:

$$\begin{aligned} \vec{q}_i &\sim \text{Dirichlet}_K(\vec{\alpha}) , \\ \vec{w}_i &\sim \text{Multinomial}_J(\vec{q}_i^T \vec{\Theta}, L_i) . \end{aligned}$$

and a corresponding generalisation is given by

$$\begin{aligned} r_{i,k} &\sim \text{Gamma}(\alpha_k, \beta_k) && \text{for } l = 0, \dots, K - 1 , \\ w_{i,j} &\sim \text{Poisson}\left(\sum_k r_{i,k} \theta_{k,j}\right) && \text{for } j = 0, \dots, J - 1 . \end{aligned}$$

The correspondence happens when β_k is constant then the Gammas normalise to the Dirichlet and the individual Poissons normalise to the multinomial. This model can also be generalised to the hierarchical case replacing $\vec{\Theta}$ by $(\Upsilon \vec{\Theta})$, using

$$\begin{aligned} r_{i,k} &\sim \text{Gamma}(\alpha_k, \beta_k) && \text{for } l = 0, \dots, K - 1 , \\ w_{i,j} &\sim \text{Poisson}\left(\sum_{h,k} r_{i,k} v_{k,h} \theta_{h,j}\right) && \text{for } j = 0, \dots, J - 1 . \end{aligned}$$

3.3 Related Work

These models are a variation of hierarchical Pachinko allocation (hPAM), model 2 [4]. The two versions of $\vec{\Upsilon}$ yield different sampling complexities and priors. When $\vec{\Upsilon}$ is derived from $\vec{\phi}$ using Equation (1), the update equations are faster, involving one less loop, and the structures should be more stable. Hierarchies in the approach here can be many levels, be strictly tree structured or fully connected. Many variations are allowed depending on the non-zeroes allowed in $\vec{\Upsilon}$.

These models are different from so-called hierarchical LDA (hLDA) [1] which has a hierarchical model but loses its ability to allow fully-fledged multi-aspect modelling because in the one document, only one most specific component can exist. hLDA allows more flexibility in the structure it builds (using Dirichlet processes), but provides less flexibility in the components. Multi-aspects models would seem to be essential for image processing, and thus hLDA less preferred.

4. Analysis

This section presents the likelihoods, and various marginal versions of it suitable for Gibbs sampling. It also presents a potential approximation that would be useful inside a heuristic search algorithm for hierarchies.

4.1 Likelihoods

4.1.1 THE FIXED STRUCTURE CASE

First, consider the case where $\vec{\Upsilon}$ is fixed between documents.

The likelihood for a document therefore takes the form:

$$p(\vec{q}_i, \vec{c}_i, \vec{h}_i, \vec{j}_i \mid \vec{\alpha}, \vec{\Theta}, \vec{\Upsilon}) = \frac{1}{Z_D(\vec{\alpha})} \prod_{k=0, \dots, K-1} q_{i,k}^{\alpha_k - 1} \prod_{l=0, \dots, L_i} q_{i,c_{i,l}} v_{c_{i,l}, h_{i,l}} \theta_{h_{i,l}, j_{i,l}} .$$

Denote by \vec{C}_i the vector of topic totals for the i -th document, and \vec{W}_h the vector of word totals for the h -th hierarchical topic, so

$$\begin{aligned} C_{i,k} &= \sum_{l=0, \dots, L_i} 1_{c_{i,l}=k} , \\ W_{h,j} &= \sum_{\substack{l=0, \dots, L_i \\ i=0, \dots, I-1}} 1_{h_{i,l}=h} 1_{j_{i,l}=j} . \end{aligned}$$

then one can marginalise out \vec{q}_i and $\vec{\Theta}$ using the priors to obtain

$$\begin{aligned} &p(\{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} \mid \vec{\alpha}, \vec{\gamma}, \vec{\Upsilon}) \\ &= \prod_{h=0, \dots, H-1} \frac{Z_H(\vec{\gamma} + \vec{W}_h)}{Z_H(\vec{\gamma})} \prod_{i=0, \dots, I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} \prod_{l=0, \dots, L_i} v_{c_{i,l}, h_{i,l}} . \end{aligned} \quad (2)$$

This formula is adequate when the Υ values are constant. If these are to be fitted as well, then a prior is needed and, assuming the parts are Dirichlet, then standard methods can be used to obtain a closed form with Υ integrated out. Since Υ sums to one on its columns, we can place Dirichlet priors on each column, with prior vectors $\vec{\psi}$ of dimension H , and then:

$$\begin{aligned} &p(\{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} \mid \vec{\alpha}, \vec{\gamma}, \vec{\psi}) \\ &= \prod_{h=0, \dots, H-1} \frac{Z_H(\vec{\gamma} + \vec{W}_h)}{Z_H(\vec{\gamma})} \prod_{i=0, \dots, I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} \prod_{k=0, \dots, K-1} \frac{Z_H(\vec{\psi} + \vec{D}_k)}{Z_H(\vec{\psi})} , \end{aligned} \quad (3)$$

where

$$D_{k,h} = \sum_{\substack{l=0, \dots, L_i \\ i=0, \dots, I-1}} 1_{h_{i,l}=h} 1_{c_{i,l}=k} .$$

Using the fixed model of Formula (1), and using a conjugate Beta prior on each ϕ_h with parameters $\psi_{h,0}, \psi_{h,1}$ (from the vector $\vec{\Psi}$):

$$\begin{aligned} &p(\{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} \mid \vec{\alpha}, \vec{\gamma}, \vec{\Psi}) \\ &= \prod_{h=0, \dots, H-1} \frac{Z_H(\vec{\gamma} + \vec{W}_h)}{Z_H(\vec{\gamma})} \prod_{i=0, \dots, I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} \\ &\quad \prod_{h=0, \dots, H-1} \frac{\text{Beta}(N_h + \psi_{h,0}, M_h + \psi_{h,1})}{\text{Beta}(\psi_{h,0}, \psi_{h,1})} , \end{aligned} \quad (4)$$

where

$$\begin{aligned} N_h &= \sum_{\substack{l=0, \dots, L_i \\ i=0, \dots, I-1}} 1_{h_{i,l}=h} , \\ M_h &= \sum_{\substack{l=0, \dots, L_i \\ i=0, \dots, I-1}} 1_{h_{i,l} \in \text{decs}(h)/h} . \end{aligned}$$

4.1.2 THE PER-DOCUMENT STRUCTURE CASE

Second, consider the case where $\vec{\Upsilon}$ is document specific, sampled from a vector of Dirichlets. In this case, a second part of the likelihood is given by

$$p(\vec{\Upsilon} | \vec{\alpha}, \vec{\Psi}, \vec{\Theta}) = \prod_{k=0, \dots, K-1} \frac{1}{Z_D(\vec{\psi}_k)} \prod_{h=0, \dots, H-1} v_{k,h}^{\psi_{k,h}^{-1}}.$$

Using the same analysis as before, we now need the structure statistics kept per document, so

$$D_{i,k,h} = \sum_{l=0, \dots, L_i} 1_{h_{i,l}=h} 1_{c_{i,l}=k}.$$

and the resultant likelihood after marginalising out $\vec{\Upsilon}$ per document becomes:

$$\begin{aligned} & p(\{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} | \vec{\alpha}, \vec{\gamma}, \vec{\Psi}) \\ &= \prod_{h=0, \dots, H-1} \frac{Z_H(\vec{\gamma} + \vec{W}_h)}{Z_H(\vec{\gamma})} \prod_{i=0, \dots, I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} \prod_{\substack{k=0, \dots, K-1 \\ i=0, \dots, I-1}} \frac{Z_H(\vec{\psi}_k + \vec{D}_{i,k})}{Z_H(\vec{\psi}_k)}, \end{aligned} \quad (5)$$

Using the fixed model of Formula (1), and using a conjugate Beta prior on each ϕ_h with parameters $\psi_{h,0}, \psi_{h,1}$, the analysis is modified analogously to the above case.

4.1.3 USING GAMMA-POISSON MODELS

Both the above versions are easily modified to handle the Gamma-Poisson case of hierarchical components introduced in Section 3.2. The term for the component prior involving $\vec{\alpha}$, which is a ratio of normalising constants for each document, is replaced by the corresponding Gamma normalising constants

$$\prod_{\substack{k=0, \dots, K-1 \\ i=0, \dots, I-1}} \frac{Z_\Gamma(\alpha_k + C_{i,k}, \beta_k + 1)}{Z_\Gamma(\alpha_k, \beta_k)},$$

where $Z_\Gamma(\alpha_k, \beta_k) = \Gamma(\alpha_k) \beta_k^{-\alpha_k}$. The conditional Gamma-Poisson model has an additional term as given in [2].

$$\prod_{\substack{k=0, \dots, K-1 \\ i=0, \dots, I-1}} \left((1 - \rho_k) \frac{Z_\Gamma(\alpha_k + C_{i,k}, \beta_k + 1)}{Z_\Gamma(\alpha_k, \beta_k)} + \rho_k 1_{C_{i,k}=0} \right).$$

The main term in the product simplifies to

$$\begin{cases} (1 - \rho_k) \frac{Z_\Gamma(\alpha_k + C_{i,k}, \beta_k + 1)}{Z_\Gamma(\alpha_k, \beta_k)} & \text{when } C_{i,k} > 0 \\ \rho_k + (1 - \rho_k) \frac{\beta_k^{\alpha_k}}{(1 + \beta_k)^{\alpha_k}} & \text{when } C_{i,k} = 0 \end{cases}$$

4.2 An approximation

We will consider an approximation that will allow one to iterate over candidate values of \vec{c}_i . For this, we sum out eligible values of $h_{i,l}$ matching a given $c_{i,l}$. Note we expect most $v_{c,h}$ are zero.

$$\begin{aligned} & p(\{\vec{c}_i, \vec{j}_i : i = 0, \dots, I-1\} | \vec{\alpha}, \vec{\gamma}, \vec{\Upsilon}) \\ &= \prod_{i=0, \dots, I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} \sum_{\substack{h_{i,l} : l=0, \dots, L_i \\ i=0, \dots, I-1}} \left(\prod_{h=0, \dots, H-1} \frac{Z_H(\vec{\gamma} + \vec{W}_h)}{Z_H(\vec{\gamma})} \prod_{\substack{i=0, \dots, I-1 \\ l=0, \dots, L_i}} v_{c_{i,l}, h_{i,l}} \right). \end{aligned}$$

The sum on the right now takes the form of an expectation under the distribution $q(h_{i,l} | c_{i,l}) = v_{c_{i,l}, h_{i,l}}$. A first crude approximation is to replace \vec{W}_h by its mean under this distribution, lets call it \widehat{W}_h , so

$$\begin{aligned} & p(\{\vec{c}_i, \vec{j}_i : i = 0, \dots, I-1\} | \vec{\alpha}, \vec{\gamma}, \vec{\Upsilon}) \\ & \approx \prod_{i=0, \dots, I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} \prod_{h=0, \dots, H-1} \frac{Z_J(\vec{\gamma} + \widehat{W}_h)}{Z_J(\vec{\gamma})}, \end{aligned} \quad (6)$$

where

$$\widehat{W}_{h,j} = \sum_{\substack{l=0,\dots,L_i \\ i=0,\dots,I-1}} v_{c_{i,l},h} 1_{j_{i,l}=j} = \sum_{c=0,\dots,K-1} v_{c,h} W'_{c,j},$$

$$\text{where } W'_{c,j} = \sum_{\substack{l=0,\dots,L_i \\ i=0,\dots,I-1}} 1_{c_{i,l}=c} 1_{j_{i,l}=j}$$

Note this approximation should work quite well for frequent words due to the tendency of large sums to behave like Gaussians and the general central limit theorem.

5. Algorithms

This section presents some of the basic algorithms used.

5.1 Gibbs Sampling for Fixed Structure

A local search or Gibbs sampling algorithm for the exact formula is achieved by considering the Formula (3) with respect to just a single item (i, l) . In this case, we get

$$p\left(c_{i,l}, h_{i,l} \mid \vec{\alpha}, \vec{\gamma}, \vec{\Psi}, \{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} / \{c_{i,l}, h_{i,l}\}\right) \quad (7)$$

$$\propto (\alpha_{c_{i,l}} + C_{i,c_{i,l}}) \frac{\gamma_{h_{i,l},j_{i,l}} + W_{h_{i,l},j_{i,l}}}{\sum_j (\gamma_{h_{i,l},j} + W_{h_{i,l},j})} \frac{\psi_{h_{i,l}} + D_{c_{i,l},h_{i,l}}}{\sum_h (\psi_h + D_{c_{i,l},h})}$$

Using the fixed model of Formula (1) for the prior on $\vec{\Upsilon}$, and its corresponding posterior Formula (4), this becomes

$$p\left(c_{i,l}, h_{i,l} \mid \vec{\alpha}, \vec{\gamma}, \vec{\Psi}, \{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} / \{c_{i,l}, h_{i,l}\}\right) \quad (8)$$

$$\propto (\alpha_{c_{i,l}} + C_{i,c_{i,l}}) \frac{\gamma_{h_{i,l},j_{i,l}} + W_{h_{i,l},j_{i,l}}}{\sum_j (\gamma_{h_{i,l},j} + W_{h_{i,l},j})}$$

$$\frac{\psi_{h,0} + N_h}{\psi_{h,0} + \psi_{h,1} + N_h + M_h} \prod_{h' \in \text{ancs}(h)} \frac{\psi_{h',1} + M_{h'}}{\psi_{h',0} + \psi_{h',1} + N_{h'} + M_{h'}}$$

Note that $c_{i,l}$ only occurs in one term, so can be easily summed out to form the marginal on $h_{i,l}$. Thus, one can sample the $h_{i,l}$ individually using marginalised Equation (8), and then the $c_{i,l}$. So sample using $p(h_{i,l} \mid \dots)$ and then $p(c_{i,l} \mid h_{i,l}, \dots)$ where $p(h_{i,l} \mid \dots) = \sum_{c_{i,l}} p(c_{i,l}, h_{i,l} \mid \dots)$.

5.2 Gibbs Sampling for per-Document Structure

A local search or Gibbs sampling algorithm for the exact formula is achieved by considering the Formula (5) with respect to just a single item (i, l) . In this case, we get

$$p\left(c_{i,l}, h_{i,l} \mid \vec{\alpha}, \vec{\gamma}, \vec{\Psi}, \{\vec{c}_i, \vec{h}_i, \vec{j}_i : i = 0, \dots, I-1\} / \{c_{i,l}, h_{i,l}\}\right) \quad (9)$$

$$\propto (\alpha_{c_{i,l}} + C_{i,c_{i,l}}) \frac{\gamma_{h_{i,l},j_{i,l}} + W_{h_{i,l},j_{i,l}}}{\sum_j (\gamma_{h_{i,l},j} + W_{h_{i,l},j})} \frac{\psi_{c_{i,l},h_{i,l}} + D_{i,c_{i,l},h_{i,l}}}{\sum_h (\psi_{c_{i,l},h} + D_{i,c_{i,l},h})}$$

The only difference here is that statistics $D_{c,h}$ are now indexed by document, so are $D_{i,c,h}$.

Using the fixed model of Formula (1) for the prior on $\vec{\Upsilon}$ and its posterior, this corresponds to the previous case, where statistics N_h and M_h become indexed by the document, so they become $N_{i,h}$ and $M_{i,h}$.

5.3 Gibbs Sampling for Conditional Gamma-Poisson

In the conditional Gamma-Poisson model, replace the term of $k = c_{i,l}$, which is, $(\alpha_{c_{i,l}} + C_{i,c_{i,l}})$ in the previous sampling Formula (7), (8) and (9), by

$$\frac{\alpha_k + C_{i,k}}{1 + \beta_k} \left(1 + \frac{\rho_k (1 + \beta_k)^{\alpha_k}}{(1 - \rho_k) \beta_k^{\alpha_k}}\right)^{-1_{C_{i,k}=0}}.$$

The regular Gamma-Poisson case holds when $\rho_k = 0$.

5.4 Hill Climbing in Structure Space

Using the approximation of Formula (6), one can build a structure bottom up. First, components are build using standard DCA where no structure $\vec{\Upsilon}$ is used. In this case, $H = K$ and $\vec{\Upsilon}$ is the identity matrix. Then one proposes to combine pairs of components by creating a hierarchical node that contains them, or add components to another hierarchical node. Thus is done using Formula (6) and a fixed model of $\vec{\Upsilon}$ derived from $\vec{\phi}$ using Equation (1), where the structure itself is being modified by joining leaf nodes, etc.

5.5 Estimating Hyper-parameters for Component Priors

When estimating the hyper-parameter $\vec{\alpha}$ for the components, the key formula is the ratio of normalising constants. We consider the maximum likelihood formula here (and in practice, some small conjugate priors can be added).

5.5.1 DIRICHLET MULTINOMIAL COMPONENTS

For the Dirichlet case, this evaluates to:

$$\prod_{i=0,\dots,I-1} \frac{Z_K(\vec{\alpha} + \vec{C}_i)}{Z_K(\vec{\alpha})} = \prod_{i=0,\dots,I-1} \frac{\Gamma(\sum_k \alpha_k)}{\Gamma(\sum_k (\alpha_k + C_{i,k}))} \prod_k \frac{\Gamma(\alpha_k + C_{i,k})}{\Gamma(\alpha_k)}.$$

Taking the first derivative and rearranging, this has a maximum for the $\vec{\alpha}$ when for each k ,

$$\psi_0(\alpha_k) - \psi_0\left(\sum_k \alpha_k\right) = \frac{1}{I} \sum_{i=0,\dots,I-1} \left(\psi_0(\alpha_k + C_{i,k}) - \psi_0\left(\sum_k (\alpha_k + C_{i,k})\right) \right). \quad (10)$$

A nested fixed point update for the Dirichlet case is as follows:

1. Compute for each k

$$E_k = \frac{1}{I} \sum_{i=0,\dots,I-1} \left(\psi_0(\alpha_k + C_{i,k}) - \psi_0\left(\sum_k (\alpha_k + C_{i,k})\right) \right).$$

2. Solve the set of equations $\psi_0(\alpha_k) - \psi_0(\sum_k \alpha_k) = E_k$ using the iteration

$$\alpha_k \leftarrow \psi_0^{-1}\left(E_k + \psi_0\left(\sum_k \alpha_k\right)\right),$$

where the inverse of the digamma function is given in Minka's notes on Dirichlets.

The two steps constitute one cycle of a fixed point update. Call this the *Dirichlet hyper-parameter estimation algorithm*. Note the first step is done during the larger process of performing the general Gibbs cycle. The second step is done at the end of the major Gibbs cycle.

5.5.2 GAMMA POISSON COMPONENTS

For the Gamma-Poisson case, we consider each component k in turn since they are independent. This relevant likelihood term is:

$$\prod_{i=0,\dots,I-1} \frac{\Gamma(\alpha_k + C_{i,k}) \beta_k^{\alpha_k}}{\Gamma(\alpha_k) (\beta_k + 1)^{(\alpha_k + C_{i,k})}}$$

The derivative w.r.t. β_k yields the optimal value

$$\beta_k \leftarrow \frac{\alpha_k}{\overline{C_{\cdot,k}}},$$

where we denote $\overline{C_{\cdot,k}} = \frac{1}{I} \sum_{i=0,\dots,I-1} C_{i,k}$. For α_k , we need to maximise the likelihood which has the derivative

$$\sum_{i=0,\dots,I-1} (\psi_0(\alpha_k + C_{i,k}) - \psi_0(\alpha_k)) + I \log \frac{\beta_k}{1 + \beta_k} = 0$$

Substituting the optimal value for β_k yields the equation

$$\frac{1}{I} \sum_{i=0, \dots, I-1} \psi_0(\alpha_k + C_{i,k}) - \psi_0(\alpha_k) + \log \frac{\alpha_k}{\alpha_k + \overline{C}_{\cdot,k}} = 0$$

This can be solved using the same methods as above.

1. Compute for each k

$$E_k = \frac{1}{I} \sum_{i=0, \dots, I-1} \psi_0(\alpha_k + C_{i,k}) - \log(\alpha_k + \overline{C}_{\cdot,k}) .$$

2. Solve the set of equations $\psi_0(\alpha_k) - \log \alpha_k = E_k$ using the iteration

$$\alpha_k \leftarrow \psi_0^{-1}(E_k + \log \alpha_k) ,$$

The Hessian of this fixed point is less than 0.9 for $\alpha_k < 5$, which is a suitable upper bound for the shape parameter of a Gamma in this case. Call this the *Gamma hyper-parameter estimation algorithm*.

5.5.3 CONDITIONAL GAMMA POISSON COMPONENTS

For the conditional Gamma-Poisson case, we consider each component k in turn since they are independent. This relevant likelihood term is:

$$\prod_{\substack{i=0, \dots, I-1 \\ C_{i,k} > 0}} (1 - \rho_k) \frac{\Gamma(\alpha_k + C_{i,k}) \beta_k^{\alpha_k}}{\Gamma(\alpha_k) (\beta_k + 1)^{(\alpha_k + C_{i,k})}} \prod_{\substack{i=0, \dots, I-1 \\ C_{i,k} = 0}} \left(\rho_k + (1 - \rho_k) \frac{\beta_k^{\alpha_k}}{(1 + \beta_k)^{\alpha_k}} \right)$$

This is best viewed as an EM style problem, with a latent variable $d_{i,k}$ that indicates if the feature is off. The EM intermediate probability then, corresponding to $p(d_{i,k} | C_{i,k})$ is only relevant when $C_{i,k} = 0$, since if $C_{i,k} > 0$, $d_{i,k} = 1$. So we introduce the intermediate probability

$$q_{i,k} = p(d_{i,k} = 0 | C_{i,k} = 0) = q_k = \frac{\rho_k}{\rho_k + (1 - \rho_k) \frac{\beta_k^{\alpha_k}}{(1 + \beta_k)^{\alpha_k}}} .$$

This is independent of the data index i and is used to turn the sum above in the likelihood into a product. Denote $N_{\cdot,k}^+ = \sum_i 1_{C_{i,k} > 0}$ and $N_{\cdot,k}^0 = \sum_i 1_{C_{i,k} = 0}$, so that $N_{\cdot,k}^+ + N_{\cdot,k}^0 = I$, denote $\overline{C}_{\cdot,k} = \frac{1}{I} \sum_{i=0, \dots, I-1} C_{i,k}$. It follows using standard EM theory that two of the re-estimation formula are:

$$\begin{aligned} \rho_k &\leftarrow \frac{N_{\cdot,k}^0 q_{i,k}}{I} \\ \beta_k &\leftarrow \left(1 - \frac{N_{\cdot,k}^0 q_k}{I} \right) \frac{\alpha_k}{\overline{C}_{\cdot,k}} . \end{aligned}$$

For α_k , for EM we need to maximise the likelihood. Its log, dropping terms without α_k , is given by

$$\sum_{i=0, \dots, I-1} (\log \Gamma(\alpha_k + C_{i,k}) - C_{i,k} \log(1 + \beta_k) - \log \Gamma(\alpha_k)) + \alpha_k (I - N_{\cdot,k}^0 q_k) \log \frac{\beta_k}{1 + \beta_k}$$

We wish to maximise this subject to the constraint $\beta_k = (1 - \rho_k) \alpha_k / C_{i,k}$. Lagrange multipliers do not seem to work, so instead substitute β_k for its maximum and differentiate w.r.t. α_k . The first derivative is

$$\sum_{i=0, \dots, I-1} (\psi_0(\alpha_k + C_{i,k}) - \psi_0(\alpha_k)) + I(1 - \rho_k) \log \frac{(1 - \rho_k) \alpha_k}{\overline{C}_{\cdot,k} + (1 - \rho_k) \alpha_k} ,$$

A second derivative can be computed from this as well. Note this function is not concave, however, empirically it appears to be unimodal. Nevertheless, one can make steps in the direction of the derivative. Convergent fixed point updates, as in the previous cases, do not seem to exist.

Note this is an EM algorithm, and the step for α_k does not optimise. Thus unlike the Gamma and Dirichlet cases, it does not solve for the optimal hyper-parameters each time. In some cases, when r_k is close to one, an algorithm like these two could be developed. For the general case of r_k , however, convergence cannot be shown.

5.6 Estimating Hyper-parameters for per-Document Structure Priors

When the structure parameter matrix $\vec{\Upsilon}$ is shared across documents, it can be readily estimated using standard Dirichlet methods. In the per-document case, however, it is estimated on a per-document basis, and thus is never really known well. Rather, it becomes a nuisance parameter in a broader estimation problem.

The focus now is on estimating the hyper-parameters of matrix $\vec{\Psi}$. For this, we have a product of K standard Dirichlet-multinomial likelihoods that correspond to the case for $\vec{\alpha}$ and the same methods apply.

6. Identifiability in the Hierarchical Case

A key question in the use of these algorithms is the identifiability of models. We have a number of negative results on this which suggest that the hierarchical models will be very difficult to use. These are not included here.

ACKNOWLEDGEMENTS.

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. Wray Buntine acknowledges support from the EU project CLASS (IST project 027978).

References

- [1] D. Blei, T.L. Griffiths, M.I. Jordan, and J.B. Tenenbaum. Hierarchical topic models and the nested Chinese restaurant process. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.
- [2] W.L. Buntine and A. Jakulin. Discrete components analysis. In C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, editors, *Subspace, Latent Structure and Feature Selection Techniques*. Springer-Verlag, 2006.
- [3] B.P. Carlin and S. Chib. Bayesian model choice via MCMC. *Journal of the Royal Statistical Society B*, 57:473–484, 1995.
- [4] D. Mimno, W. Li, and A. McCallum. Mixtures of hierarchical topics with Pachinko allocation. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 633–640. ACM, 2007.

A. Proof of convergence for Dirichlet hyper-parameter estimation

To prove the Dirichlet hyper-parameter estimation of Section 5.5 converges, we consider $\alpha_T = \sum_k \alpha_k$, and prove the corresponding fixed point algorithm converges. This then becomes easier because it is a one-dimensional fixed point.

First, consider the second step. This is also described by Minka’s notes on Dirichlet’s in Section 1. This corresponds to a fixed point update

$$\alpha'_T = \sum_k \psi_0^{-1}(E_k + \psi_0(\alpha_T))$$

Let α'_k be shorthand for the term indexed by k inside the sum. Using the formula for a derivative of the inverse of a function, the derivative of the fixed point formula here is

$$\frac{d\alpha'_T}{d\alpha_T} = \sum_k \frac{\psi_1(\alpha_T)}{\psi_1(\alpha'_k)}$$

Now $x\psi_1(x)$ is monotonic decreasing in x , thus $\frac{\psi_1(\alpha_T)}{\psi_1(\alpha'_k)} < \frac{\alpha'_k}{\alpha_T}$, so in the vicinity of the solution, the right hand side of this last formula is less than $\sum_k \frac{\alpha'_k}{\alpha_T} = 1$. So the fixed point update is convergent near the solution.

Now consider the full fixed point. This is again a fixed point in α_T since α_k is computed from the formula $\psi_0^{-1}(E_k + \psi_0(\alpha_T))$. So construct the derivative from Equation (10).

$$\begin{aligned} & \psi_1(\alpha'_k) \frac{d\alpha'_k}{d\alpha'_T} \frac{d\alpha'_T}{d\alpha_T} - \psi_1(\alpha'_T) \frac{d\alpha'_T}{d\alpha_T} \\ &= \frac{1}{I} \sum_{i=0, \dots, I-1} \left(\psi_1(\alpha_k + C_{i,k}) \frac{d\alpha_k}{d\alpha_T} - \psi_1 \left(\alpha_T + \sum_k C_{i,k} \right) \right) \end{aligned}$$

Rearranging, we get that $\frac{d\alpha'_T}{d\alpha_T}$ is given by

$$\left(\psi_1(\alpha'_k) \frac{d\alpha'_k}{d\alpha'_T} - \psi_1(\alpha'_T) \right)^{-1} \frac{1}{I} \sum_{i=0, \dots, I-1} \left(\psi_1(\alpha_k + C_{i,k}) \frac{d\alpha_k}{d\alpha_T} - \psi_1 \left(\alpha_T + \sum_k C_{i,k} \right) \right)$$

Near the solution, the primes ($'$) can be dropped. Noting that ψ_1 is positive and monotonic decreasing in the ranges concerned, and $\frac{d\alpha_k}{d\alpha_T}$ is non-negative, it follows that

$$\psi_1(\alpha_k) \frac{d\alpha_k}{d\alpha_T} - \psi_1(\alpha_T) \geq \psi_1(\alpha_k + C_{i,k}) \frac{d\alpha_k}{d\alpha_T} - \psi_1 \left(\alpha_T + \sum_k C_{i,k} \right)$$

and the inequality will be strict when $C_{i,k} > 0$. Thus $\frac{d\alpha'_T}{d\alpha_T}$ is strictly less than 1 near a solution as long as the data is non-trivial, and convergence holds.