T-122.102 Co-occurrence methods in analysis of discrete data

Latent Dirichlet Allocation

David M. Blei, Andrew Y. Ng and Michael I. Jordan

Variational Extensions to EM and Multinomial PCA

Wray Buntine

presented by Jaakko Peltonen, 17.2.2004

Contents

Main topic:

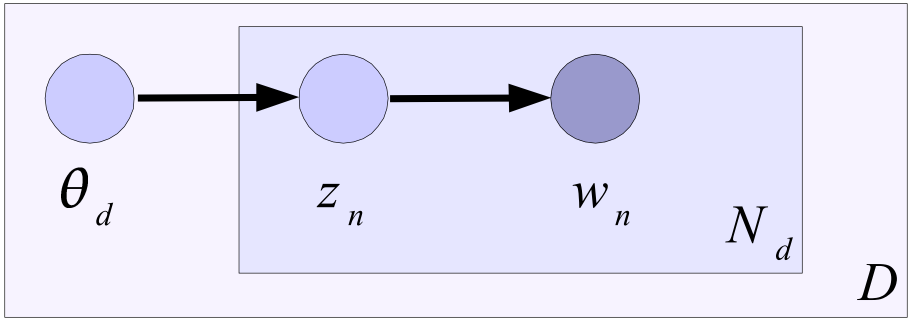
- LDA: generative model for discrete data (e.g. text)
- generalization/improvement to:
  naive Bayes/unigram, unigram mixture, PLSI

Subtopics:

- generative model:
  document = mixture of topics, mixture proportions = latent variable
- variational algorithms for inference + learning
- experiments
- another interpretation: multinomial PCA
- deriving clustering algorithms
- diagnostics

2

LDA: Generative model



- k **latent topics** = prototype word distributions $p(w|z)$

To generate a document (length N):

1. sample weights for a mixture of topics

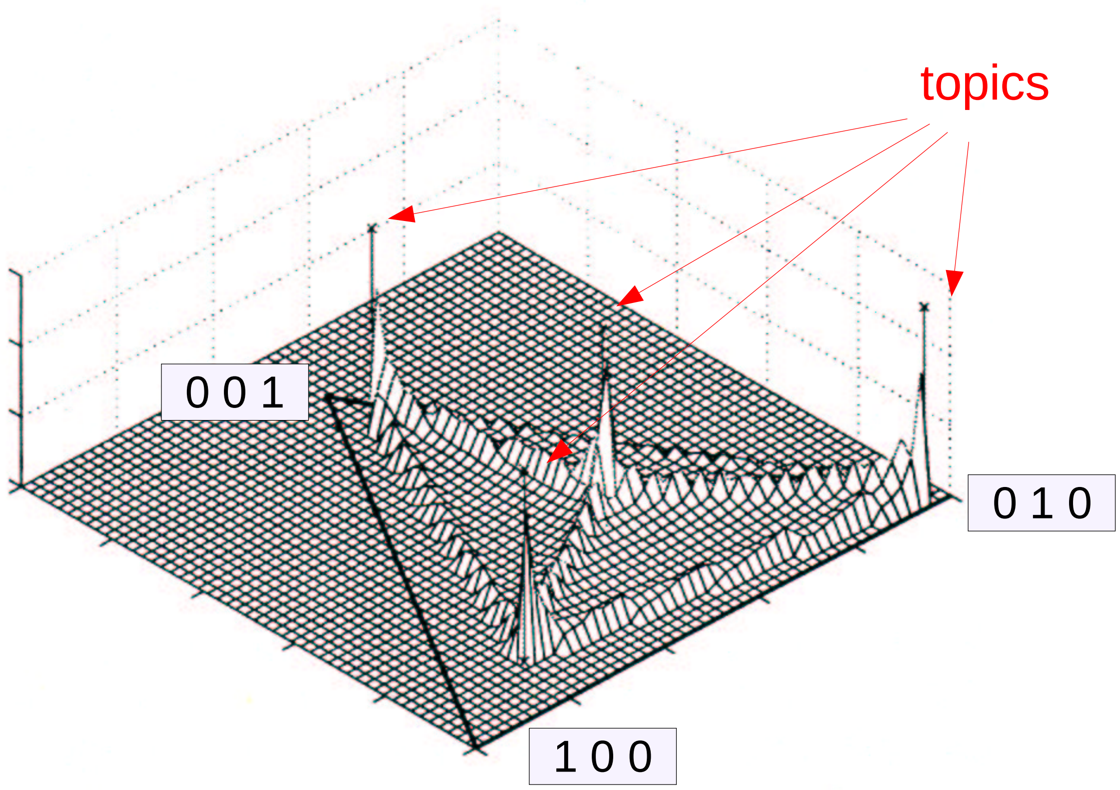
once for each document

2. two interpretations for the same thing:
   a) sample all the words from the corresponding mixture of word distributions
   b) to generate a word, choose a topic, then a word from its word distribution

once for each word

3

LDA: Generative model, continued

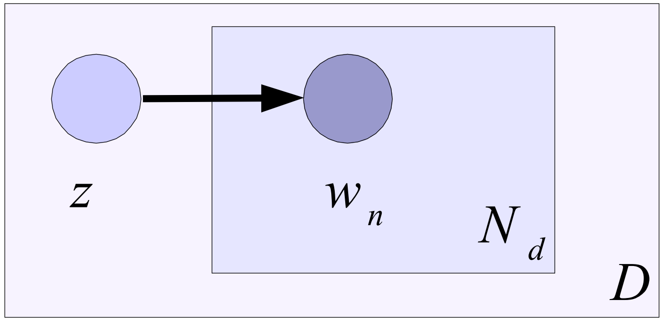


- probability of a document (= word vector $\mathbf{w}$):

$$p(\mathbf{w}) = \int_{\theta} \left(\prod_{n=1}^{N} \sum_{z_n=1}^{k} p(w_n | z_n) \, p(z_n | \theta) \right) p(\theta ; \boldsymbol{\alpha}) \, d\theta$$

- does not generate document lengths

LDA: Related Models



- **mixture of unigrams**: each document generated by 1 topic

$$p(\mathbf{w}) = \sum_{z=1}^{k} \left(\prod_{n=1}^{N} p(w_n \mid z) \right) p(z)$$

- only 1 parameter less than LDA ($k-1$ vs. k)

LDA: Related Models 2

5

- **pLSI**: document index and word are independent given the topic

$$p(d, w) = \sum_{z=1}^{k} p(w \mid z)\, p(z \mid d)\, p(d)$$

- in pLSI, d is just a document index, and $p(z \mid d)$ contains the complexity
  - → $p(z \mid d)$ learned for training documents only, separate parameters for each document
  - → not fully generative, complexity grows with data size
  - → may have **overfitting** problems
- in LDA, $\theta \sim Dirichlet$, and $p(z \mid \theta)$ is simply the z:th element of θ
  - → a generic model for documents, not just for training data

LDA: Inference

6

- Likelihood infeasible to compute exactly (hypergeometric function):

$$p(\mathbf{w};\alpha,\beta)=\frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)}\int_\theta \left(\prod_{i=1}^{k}\theta_i^{\alpha_i-1}\right)\left(\prod_{n=1}^{N}\sum_{i=1}^{k}\prod_{j=1}^{|V|}(\theta_i\beta_{ij})^{w_n^j}\right)d\theta$$

- variational approximation:

$$\begin{aligned}&\log p(\mathbf{w};\alpha,\beta)\\ &\geq E_q[\log p(\mathbf{w}|\mathbf{z};\beta)+\log p(\mathbf{z}|\theta)+\log p(\theta;\alpha)-\log q(\theta,\mathbf{z};\gamma,\phi)]\end{aligned}$$

factorized distribution

$$q(\theta;\gamma)\prod_n q(z_n;\phi_n)$$

- lower bound is computable & differentiable

→ bound can be maximized
to approximate $p(\mathbf{w};\alpha,\beta)$

7

LDA: Inference, continued

- variational EM Algorithm: maximize lower bound on log-likelihood

$$\log p(D) \geq \sum_{m=1}^{M} E_{q_m}[\log p(\theta, \mathbf{z}, \mathbf{w})] - E_{q_m}[\log q_m(\theta, \mathbf{z})]$$

- E step: coordinate ascent (maximize probability bound for 1 document)

$$\phi_{ni} \propto \beta_{iw_n} \exp\left(\Psi(\gamma_i) - \Psi\left(\sum_{j=1}^{k} \gamma_j\right)\right) , \qquad \gamma_i = \alpha_i + \sum_{n=1}^{N} \phi_{ni}$$

- M step:

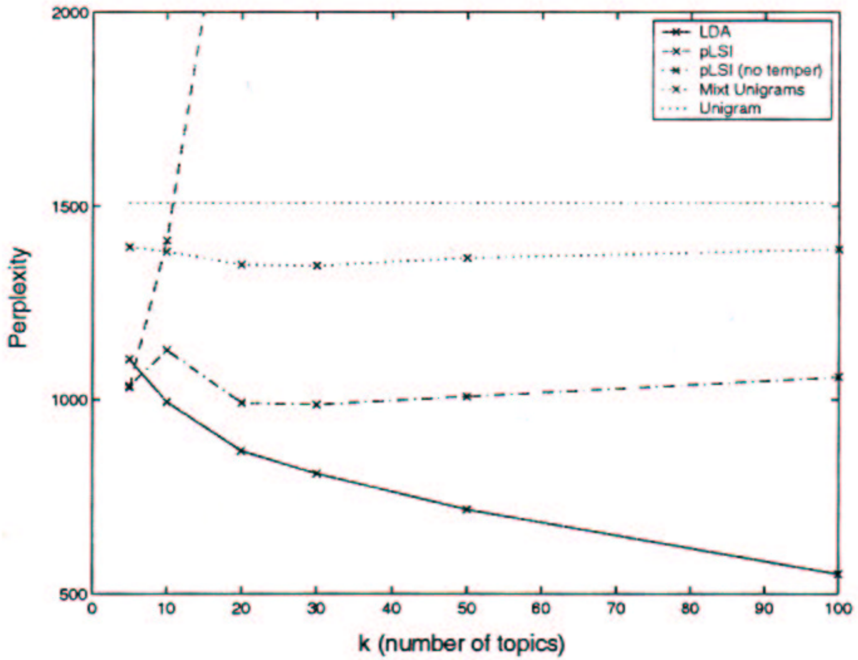
maximize β_{ij} by $\beta_{ij} \propto \sum_{m=1}^{M} \sum_{n=1}^{|\mathbf{w_m}|} \phi_{mni} w_{mn}^{j}$

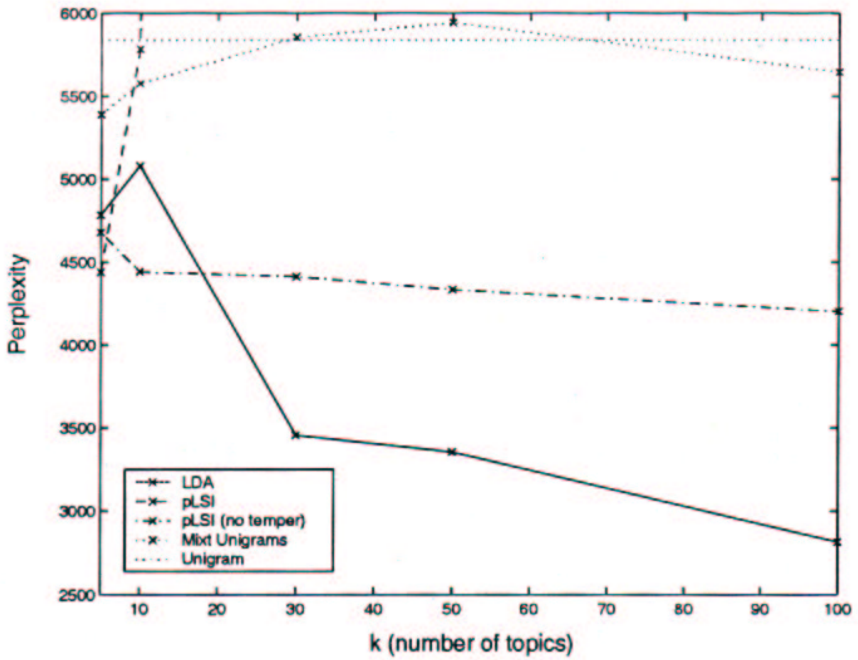
maximize α_i by Newton-Raphson method

8 LDA: Experiments 1

- **language modeling**: text corpora TREC AP (news) and CRAN (abstracts)
- evaluated by **perplexity** (inverse of **per-word likelihood** of text data)

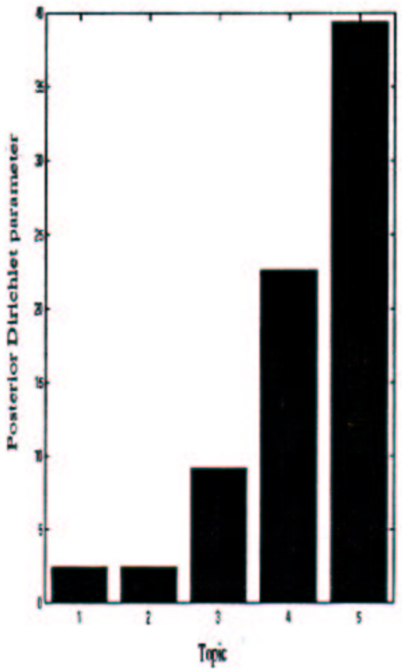
$$perplexity(D_{test}) = \exp\left(-\sum_m \log p(\mathbf{w_m}) / \sum_m |\mathbf{w_m}|\right)$$





LDA: Experiments 1, continued

- example document, topics with largest prior:



Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
SCHOOL	MILLION	SAID	SAID	SAID
SAID	YEAR	AIDS	NEW	NEW
STUDENTS	SAID	HEALTH	PRESIDENT	MUSIC
BOARD	SALES	DISEASE	CHIEF	YEAR
SCHOOLS	BILLION	VIRUS	CHAIRMAN	THEATER
STUDENT	TOTAL	CHILDREN	EXECUTIVE	MUSICAL
TEACHER	SHARE	BLOOD	VICE	BAND
POLICE	EARNINGS	PATIENTS	YEARS	PLAY
PROGRAM	PROFIT	TREATMENT	COMPANY	WON
TEACHERS	QUARTER	STUDY	YORK	TWO
MEMBERS	ORDERS	IMMUNE	SCHOOL	AVAILABLE
YEAROLD	LAST	CANCER	TWO	AWARD
GANG	DEC	PEOPLE	TODAY	OPERA
DEPARTMENT	REVENUE	PERCENT	COLUMBIA	BEST

LDA: Experiments 2

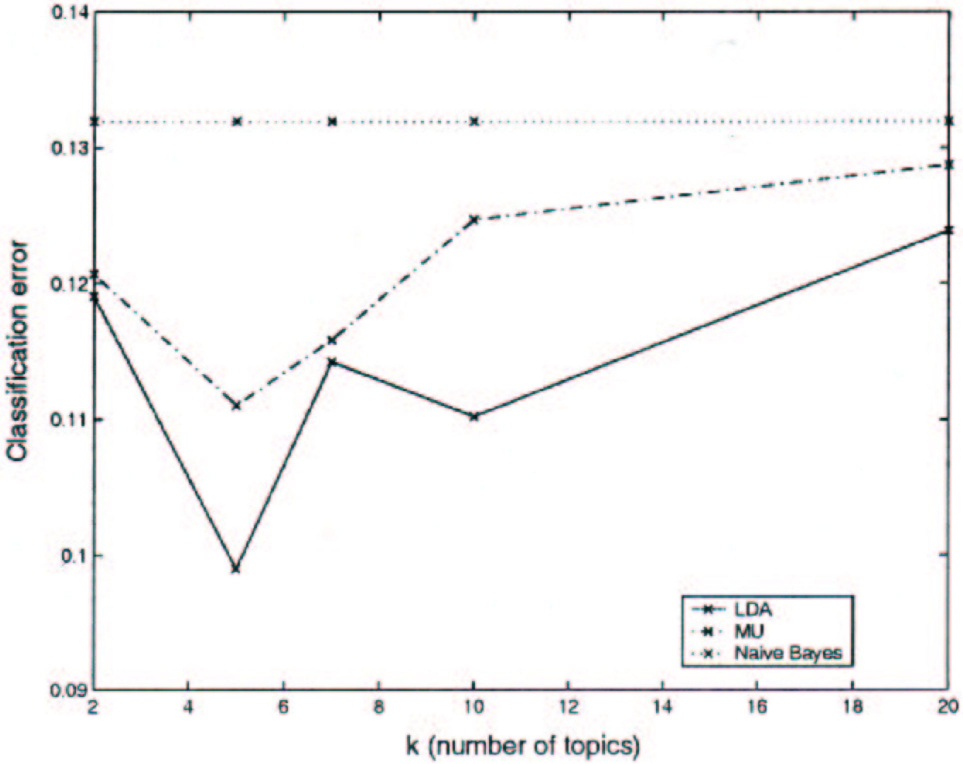
- **text classification**: WebKB dataset

- for each class, learn a separate model $p(\mathbf{w}|c)$

- classify unseen document by Bayes' rule

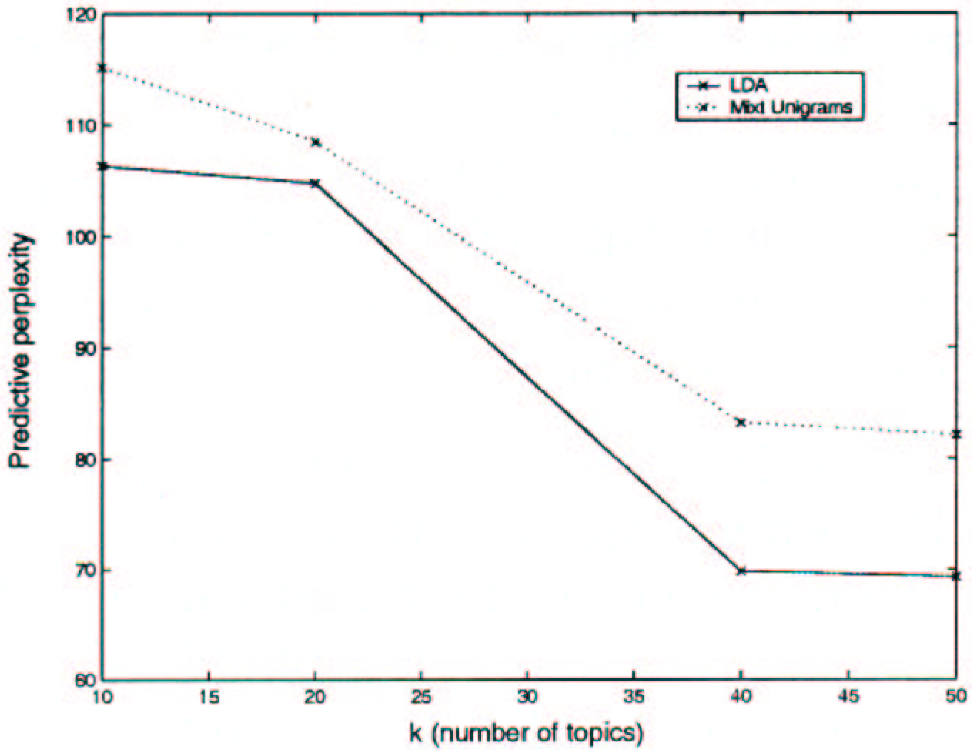
$$arg\ max_c\ p(c|\mathbf{w})$$
$$= arg\ max_c\ p(\mathbf{w}|c)\,p(c)$$

- here unigram models $\longrightarrow$ naive Bayes



LDA: Experiments 3

- **collaborative filtering**: EachMovie dataset
- users indicate preferred movies (user preferences comparable to document words)
- task: for test users, predict 1 missing preference (movie) based on their other preferences
- quality measure: likelihood given to the true missing movies



A different interpretation: Multinomial PCA

- PCA as a 2-step generative model (Gaussian, Gaussian):

$$m \sim Gaussian(0, \mathbf{I}_\mathbf{K})$$

$$x \sim Gaussian(\Omega\, m + \mu, \sigma\, \mathbf{I}_\mathbf{J}) = \Omega\, m + \mu + Gaussian(0, \sigma\, \mathbf{I}_\mathbf{J})$$

- discrete analogue (Dirichlet/Entropic, Multinomial):

$$\mathbf{m} \sim Dirichlet(\alpha) \quad \text{or} \quad \mathbf{m} \sim Entropic(\lambda)$$

$$\mathbf{x} \sim Multinomial(\Omega\, \mathbf{m}, L)$$

- in both cases, 1st step is a conjugate prior to 2nd (**exponential family**)
- latter model may restrict data to a subspace

- for PCA, 1st step can be included in covariance matrix of 2nd
  $\longrightarrow$ easily solved via EM or as eigenvector problem
- for multinomial case, no such transformation is known

Deriving clustering algorithms: preliminaries

- **exponential family**: parameters and their duals

$$q(\mathbf{x}|\theta) = \exp(\mathbf{t}(\mathbf{x})^T \theta) \;/\; (Y_t(\mathbf{x}) Z_t(\theta))$$

$$\mu_t = E_q\{\mathbf{t}(\mathbf{x})\} = \partial \log Z_t / \partial \theta \;,\; \Sigma_t = Cov_q\{\mathbf{t}(\mathbf{x})\} = \partial \mu_t / \partial \theta$$

- MAP estimate based on finite sample: $\hat{\mu}_t = (\nu_t + \sum_i \mathbf{t}(\mathbf{x_i})) / (S_t + I)$

prior, data (sufficient stats.)

prior, data (sample size)

14

Deriving clustering algorithms, continued

- It isn't known if the MAP for $p(\phi \mid \mathbf{x}_{\{\}})$ can be exactly computed
- Instead, maximize

$$
\begin{aligned}
L(\phi;\theta) &= \log p(\mathbf{x}_{\{\}}, \phi) - KL(q(\mathbf{h}_{\{\}} \mid \theta) \,\|\, p(\mathbf{h}_{\{\}} \mid \mathbf{x}_{\{\}}, \phi)) \\
&= E_{q(\mathbf{h}_{\{\}} \mid \theta)}\{\log p(\mathbf{x}_{\{\}}, \mathbf{h}_{\{\}}, \phi)\} + H(q(h_{\{\}} \mid \theta))
\end{aligned}
$$

- Kullback-Leibler (mean-field) approximation of p by q (exp. family)

$$\theta \leftarrow \frac{\partial}{\partial} mu_t E_q\{\log p(\mathbf{x} \mid \phi) + \log Y_t(\mathbf{x})\}$$

- Kullback-Leibler approximation by product $q_1(\mathbf{x_1}) q_2(\mathbf{x_2})$

$$q_1(\mathbf{x_1}) \leftarrow \exp(E_{q_2(\mathbf{x_2})}\{\log p(\mathbf{x} \mid \phi)\}) / Z_1$$

$$q_2(\mathbf{x_2}) \leftarrow \exp(E_{q_1(\mathbf{x_1})}\{\log p(\mathbf{x} \mid \phi)\}) / Z_2$$

- If the approximation can reach the true distribution $\longrightarrow$ EM algorithm

Final clustering algorithm

- Model: $\mathbf{m} \sim Dirichlet(\alpha)$ $\mathbf{c} \sim Multinomial(\mathbf{m}, L)$ $\mathbf{w_k} \sim Multinomial(\Omega_{k,\cdot}, c_k)$

  Topic proportions — number of samples from each topic — words from each topic (sum = observed words $\mathbf{r}$)

  $\Omega_{\mathbf{k_1},\cdot} \sim Dirichlet(2\ \mathbf{f})$

  Topic word distributions

- Approximation for hidden data: product distribution $q(\mathbf{m})q(\mathbf{w})$

  $\mathbf{m} \sim Dirichlet(\beta)$, $\mathbf{w}_{\cdot,\mathbf{j}} \sim Multinomial(\gamma_{j,\cdot}, r_j)$

Update rules:

$$\gamma_{j,k,[i]} \leftarrow \frac{1}{Z_{4,j,[i]}} \Omega_{k,j} \exp(\Psi_0(\beta_{k,[i]}) - \Psi_0(\sum_k \beta_{k,[i]}))$$

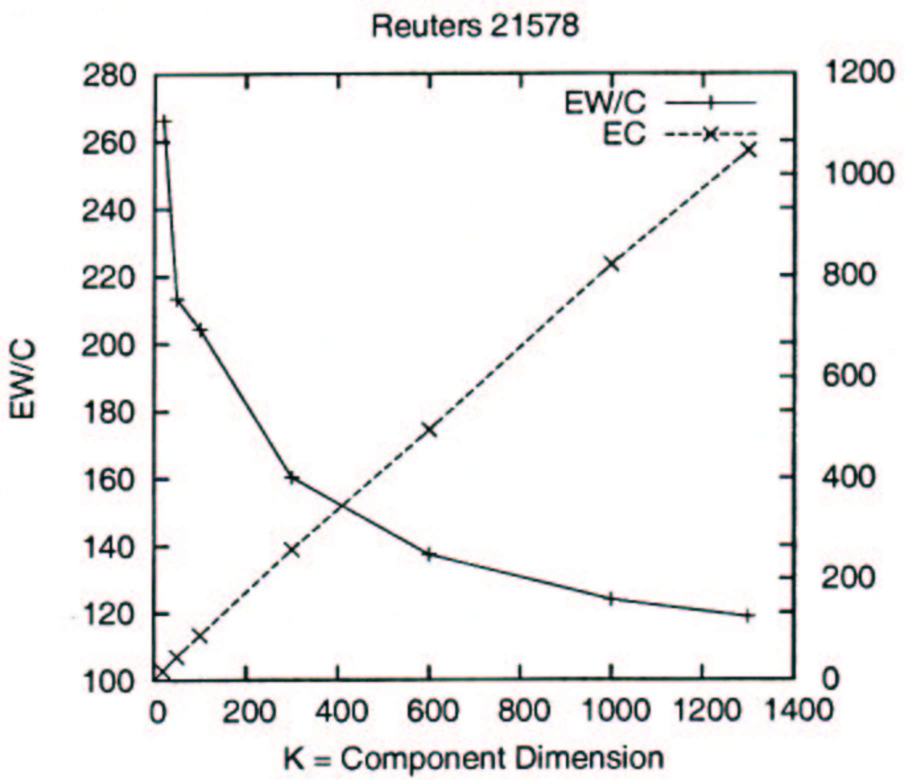
$$\beta_{k,[i]} \leftarrow \alpha_k + \sum_j r_{j,[i]} \gamma_{j,k,[i]}$$

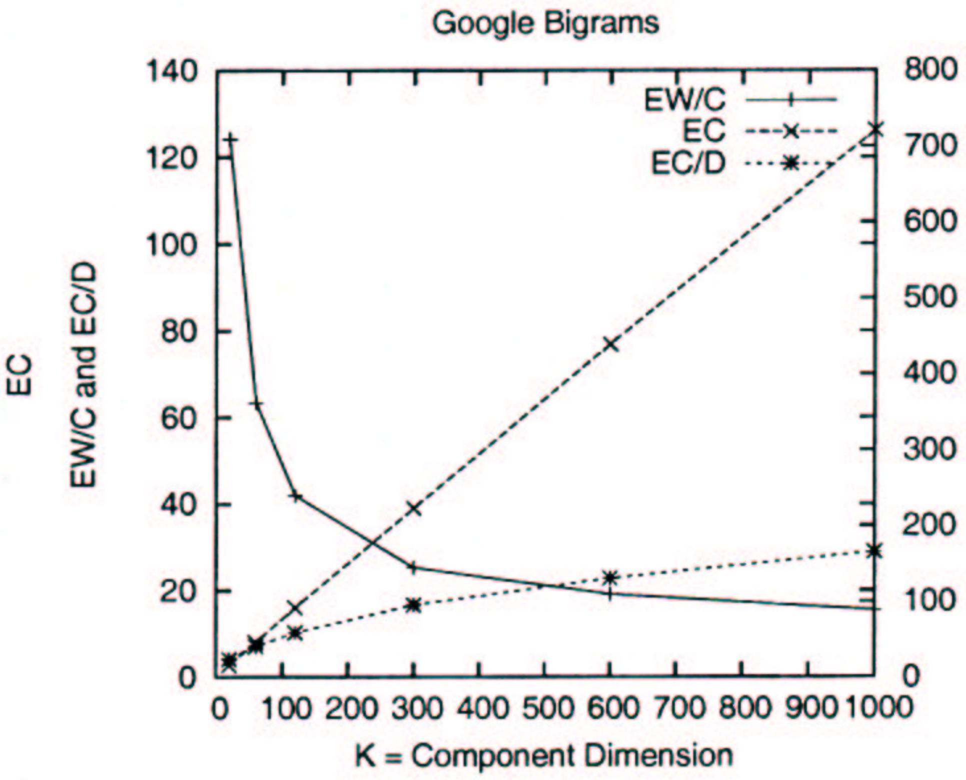
$$\Omega_{k,j} \leftarrow \frac{1}{Z_{4,k}} (2 f_j + \sum_i r_{j,[i]} \gamma_{j,k,[i]})$$

$$\Psi_0(\alpha_k) - \Psi_0(\sum_k \alpha_k) \leftarrow \frac{1}{1+I} [\log \frac{1}{K} + \sum_i (\Psi_0(\beta_{k,[i]}) - \Psi_0(\sum_k \beta_{k,[i]}))]$$

Diagnostics for the algorithms

Reuters-21578 (news, bags-of-words)   Google Bigrams (web pages)





- expected words per component **EW/C**
- expected components per document **EC/D**
- expected components **EC**

entropies of probabilities raised to power 2

Observations

- for Reuters-21578, documents belong to about 2 topics;
  for Google Bigrams, depends on sample size

- on Reuters-21578, prior yielded 4x better performance than ML estimates

- **unfolding** of components in contrast to PCA (adds components)
  → suitable for hierarchical analysis

several components per document (Google bigrams: 30+ per word)
→ suitable for dimensionality reduction