University of Southampton

# COMP6237 Data Mining
## Lecture 11: Finding Features II (Topic Modelling)

Zhiwu Huang

Zhiwu.Huang@soton.ac.uk

Lecturer (Assistant Professor) @ VLC of ECS

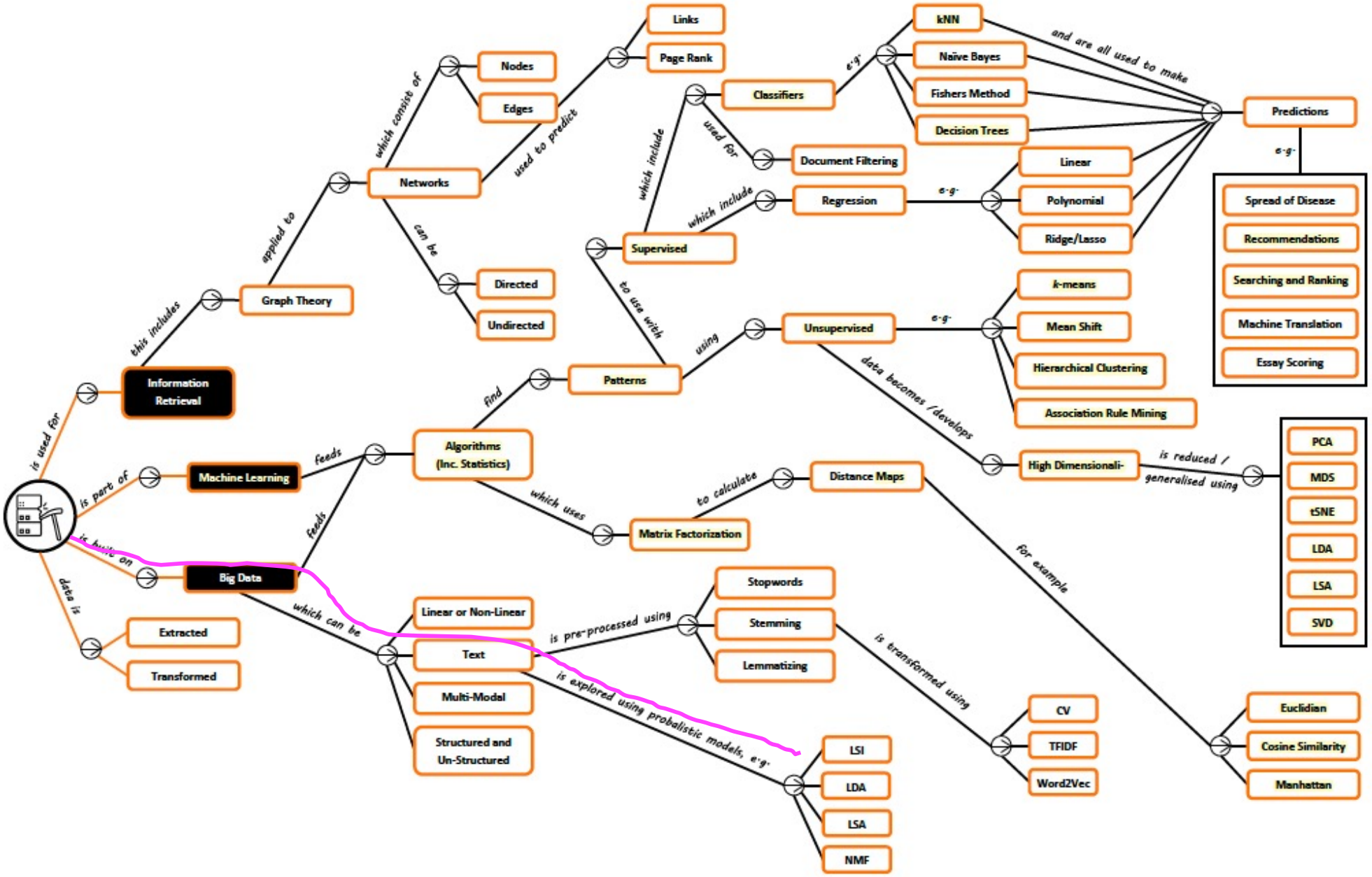
University of Southampton

Lecture slides available here:

http://comp6237.ecs.soton.ac.uk/zh.html

(Thanks to Prof. Jonathon Hare and Dr. Jo Grundy for providing the lecture materials used to develop the slides.)

# Topic Modelling – **Roadmap**



# Topic Modelling – **Textbook**

## CHAPTER 10

# Finding Independent Features

Most of the chapters so far have focused primarily on *supervised* classifiers, except Chapter 3, which was about *unsupervised* techniques called *clustering*. This chapter will look at ways to extract the important underlying features from sets of data that are not labeled with specific outcomes. Like clustering, these methods do not seek to make predictions as much as they try to characterize the data and tell you interesting things about it.

- Programming Collective Intelligence: Building Smart Web 2.0 Applications *T. Segaran*.

# Topic Modelling – **Overview (1/4)**

Uncover hidden thematic structure in a collection of documents
Helps with

- Searching
- Browsing
- Summarising

*A single document will often have more than one topic*

*Topic modeling is an unsupervised document classification method, akin to clustering in numerical data analysis*

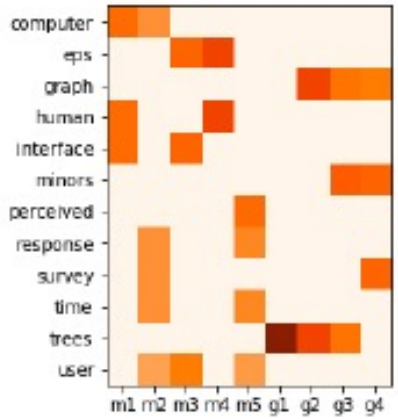
Credit: Jo Grundy

# Topic Modelling – **Overview (2/4)**

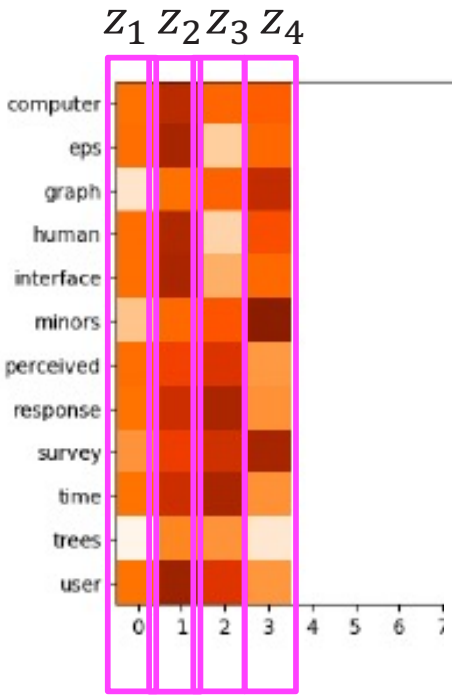
LSA found *concepts* that were linear mixtures of words associated in different documents.

- Weightings were unconstrained, could be negative
- Difficult to interpret, couldn't give *meaning* to the concept

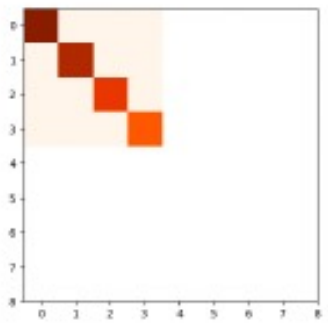
$z_i$: i-th topic



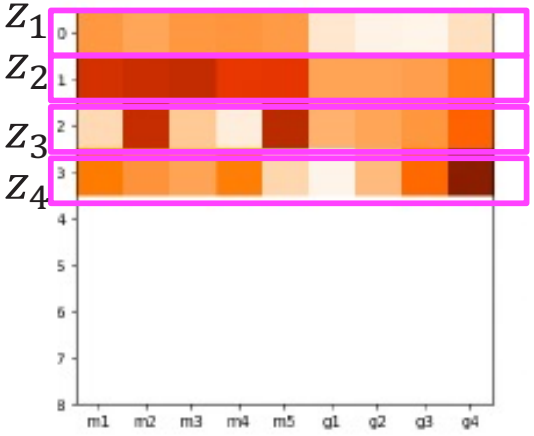

$A$




$U_r$



$\Sigma_r$



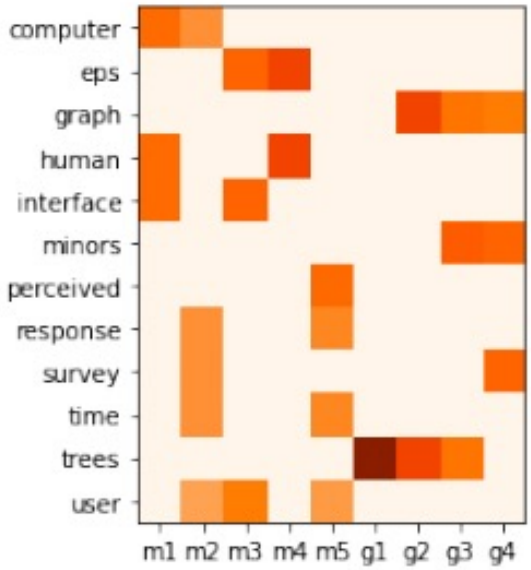

$V_r$

Credit: Jo Grundy

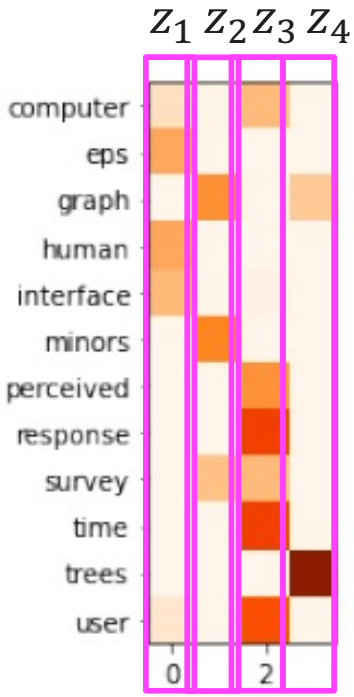
# Topic Modelling – **Overview (3/4)**

## Non-negative Matrix Factorisation (NMF)

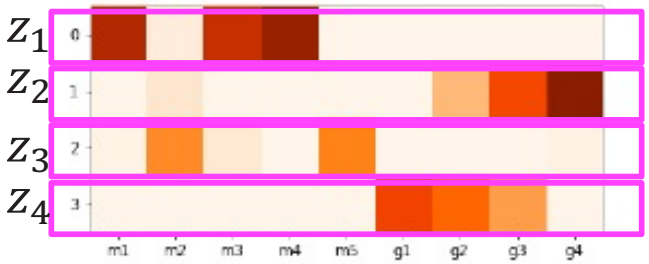
$z_i$: i-th topic




$A$




$W$




$H$

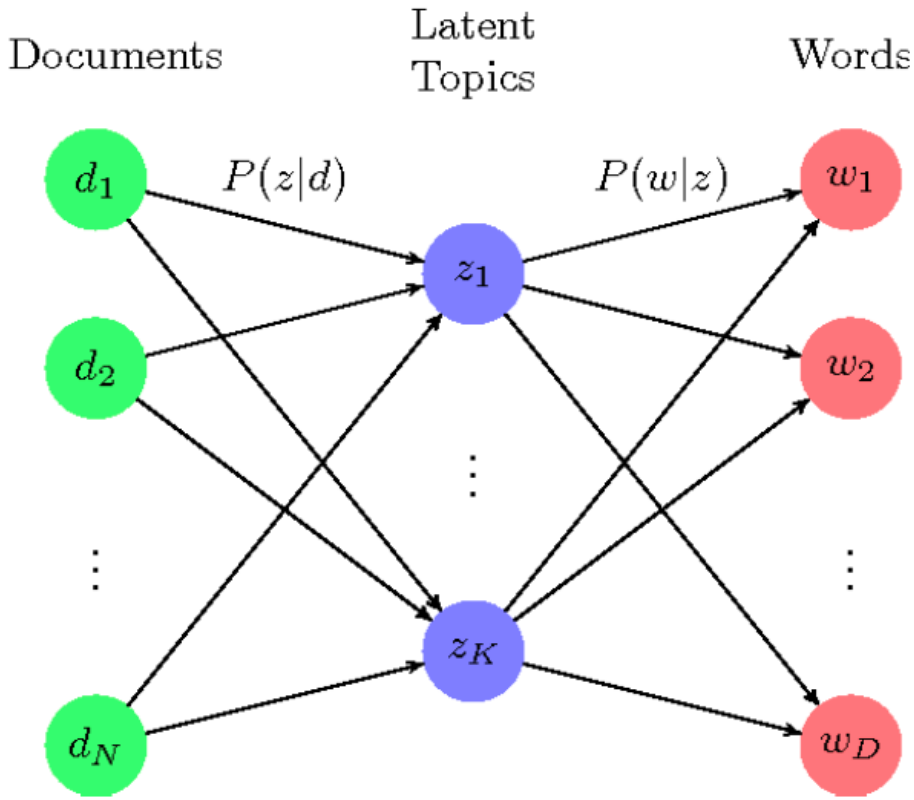
$W$ has the *basis vectors*, showing how the words are clustered

$H$ has the topic memberships for the documents.

Credit: Jo Grundy

# Topic Modelling – **Overview (4/4)**

## Probabilistic LSA (PLSA)

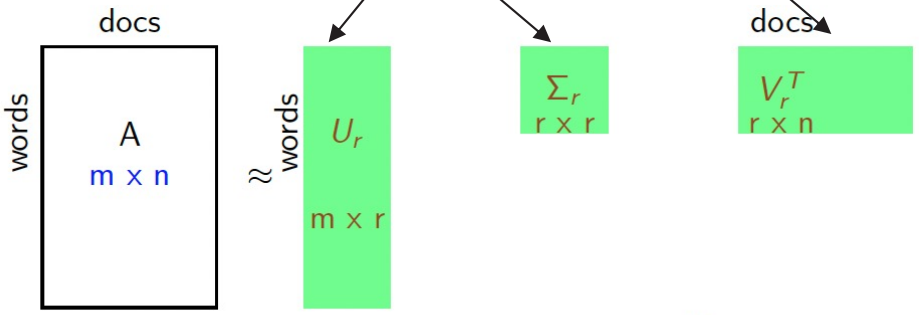



$$P(d,w) = P(d)P(w|d)$$

$$P(w|d) = \sum_{z \in \mathbb{Z}} P(w|z)P(z|d)$$

$$P(d,w) = P(d)\sum_{z \in \mathbb{Z}} P(w|z)P(z|d)$$

$$= \sum_{z \in \mathbb{Z}} P(z)P(w|z)P(z|d)$$




Topic represented by probability distribution over words

$z_i = (w_1, \ldots, w_m)$ $z_1 = (0.3, 0.1, 0.2, 0.3, 0.1)$

Document represented by probability distribution over topics

$d_j = (z_1, \ldots, z_n)$ $d_1 = (0.5, 0.3, 0.2)$

$P(d)$ can be determined directly from our corpus. $P(w|z), P(z|d)$ are modeled as multinomial distributions, and can be trained using the expectation-maximization algorithm (EM)

Credit: Andrew Levandoski, Jonathan Lobo

# Topic Modelling – **Learning Outcomes**

- **LO1**: Demonstrate an understanding of techniques for finding independent features for topic modeling, such as: (exam)
  - ❖ Comprehending the core concepts of NMF and apply NMF on a dataset
  - ❖ Understanding the key idea and steps of probabilistic models like PLSA and LDA
  - ❖ Discussing the advantages and disadvantages of the learned algorithms
- **LO2**: Implement the learned algorithms for topic modeling (coursework)

***Assessment hints: Multi-choice Questions (single answer: concepts, calculation etc)***

- *Textbook Exercises: textbooks (Programming + Mining)*
- *Other Exercises: https://www-users.cse.umn.edu/~kumar001/dmbook/sol.pdf*
- *ChatGPT or other AI-based techs*

# Topic Modelling – Introduction

There are a number of ways to do 'Topic Modelling'
Using probabilistic models:

- Probabilistic LSA
- Latent Dirichlet Allocation (LDA) (covered in AML)
- Pachinko Allocation (PAM)



Credit: Jo Grundy

# Topic Modelling – NMF

Topic modelling is like clustering as we group documents into similar sets

However we want *soft* clusters
a document should be a weighted mixture of topics

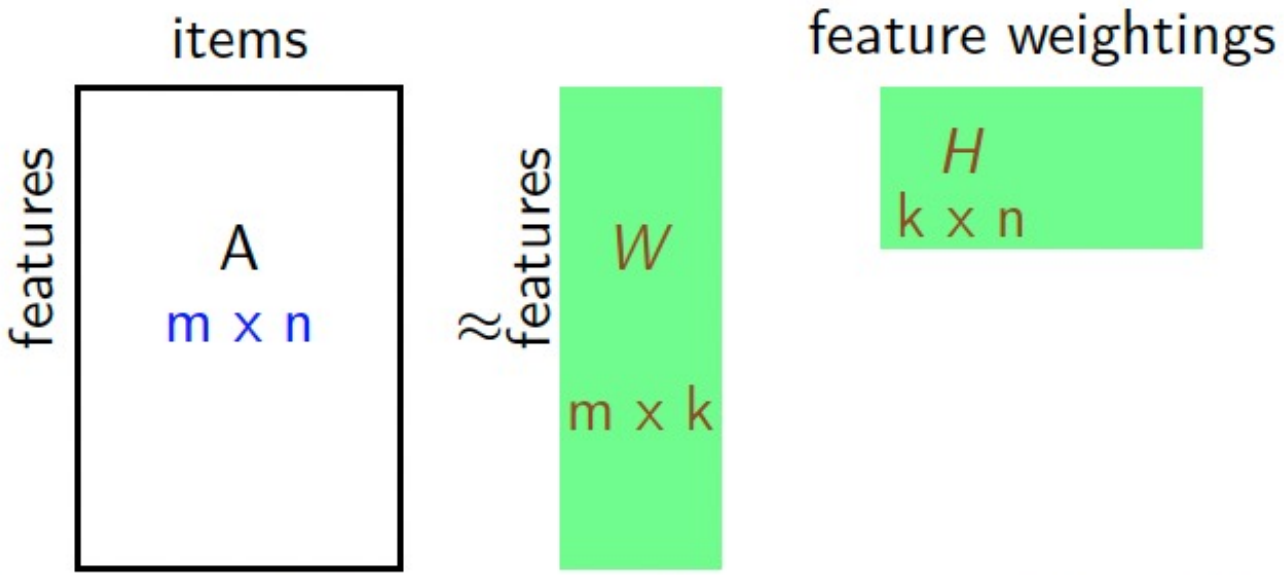
Non-Negative Matrix Factorisation achieves this via a different matrix decomposition

$$A \approx WH$$

With PCA and vector quantisation.

Credit: Jo Grundy

# Topic Modelling – NMF




$W$ contains a $k$ dimensional feature vector for each of the items

The weightings for each document are in the columns of $H$

Each basis vector (row of $W$) can be interpreted as a cluster.
Membership of each cluster is encoded in $H$

Credit: Jo Grundy

# Topic Modelling – NMF

$W$ and $H$ are found by an iterative Expectation Maximisation process

A cost function is minimised:

- Euclidean Norm $||M - WH||^2$
- KL divergence $\sum_{ij}(-M_{ij}\log(WH)_{ij} + (WH)_{ij})$

Lee and Seung 1999, Nature



Credit: Jo Grundy

# Topic Modelling – NMF Algorithm

**Algorithm 1:** NMF Algorithm with Euclidean distance

**Data:** $A$ ($m \times n$ non-negative matrix), $d$ dimensions to use

Initialise $W$ with $m \times k$ random values;

Initialise $H$ with $d \times n$ random values;

**while** *not converged* **do**

$$W_{ij} = W_{ij} \frac{(AH^T)_{ij}}{(WHH^T)_{ij}};$$

$$W_{ij} = \frac{W_{ij}}{\sum_k W_{ik}};$$

$$H_{ij} = H_{ij} \frac{W^T A_{ij}}{(WHH^T)_{ij}};$$

**end**

This has the effect of minimising the norm $||V - WH||_F^2$ subject to $W \geq 0$, $H \geq 0$



Credit: Jo Grundy

# Topic Modelling – NMF Algorithm

**Algorithm 2:** NMF Algorithm with KL-Divergence

**Data:** $A$ ($m \times n$ non-negative matrix), $d$ dimensions to use

Initialise $W$ with $m \times d$ random values;

Initialise $H$ with $d \times n$ random values;

**while** *not converged* **do**

$$W_{ij} = W_{ij} \sum_k \frac{A_{ik}}{(WH)_{ik}} H_{jk};$$

$$W_{ij} = \frac{W_{ij}}{\sum_k W_{ik}};$$

$$H_{ij} = H_{ij} \sum_k W_{ki} \frac{A_{kj}}{(WH)_{kj}};$$

**end**

This has the effect of minimising the generalized KL Divergence $\sum_{ij}(-M_{ij} \log(WH)_{ij} + (WH)_{ij})$ subject to $W \geq 0$, $H \geq 0$



Credit: Jo Grundy

# Topic Modelling – NMF Algorithm

Initialisation is usually random.
Different random initialisations can lead to **instability**
i.e. different results for different runs with the same data and $d$ value.
Improvement was reported using SVD initialisation (Boutsidis and Gallopoulos 2008)
Where:

- $W$ is initialised as $U_d\sqrt{\Sigma_d}$
- $H$ is initialised as $\sqrt{\Sigma_d}V_d^T$

However, a further study reported that random initialisation was better (Utsumi 2010)



Credit: Jo Grundy

# Topic Modelling – NMF Algorithm

Other variants can involve:

- For distance: Use of Bregman divergence (Li *et al* 2012)
- For optimisation: alternating least squares with projected gradient method for sub-problems (Lin 2007)
- For constraints:
  - Enforcing Sparseness (Hoyer 2004)
  - Using background information (Semi-NMF)
- Inputs: Symmetric matrices, e.g. Document - Documents cosine similarity matrix (Ding & He, 2005)

Credit: Jo Grundy

# Topic Modelling – NMF Examples

Example:
a set of strings:

| | |
|---|---|
| m1 | "Human machine interface for ABC computer applications" |
| m2 | "A survey of user opinion of computer system response time" |
| m3 | "The EPS user interface management system" |
| m4 | "System and human system engineering testing of EPS" |
| m5 | "Relation of user perceived response time to error measurement" |
| g1 | "The generation of random, binary, ordered trees" |
| g2 | "The intersection graph of paths in trees" |
| g3 | "Graph minors IV: Widths of trees and well-quasi-ordering" |
| g4 | "Graph minors: A survey" |

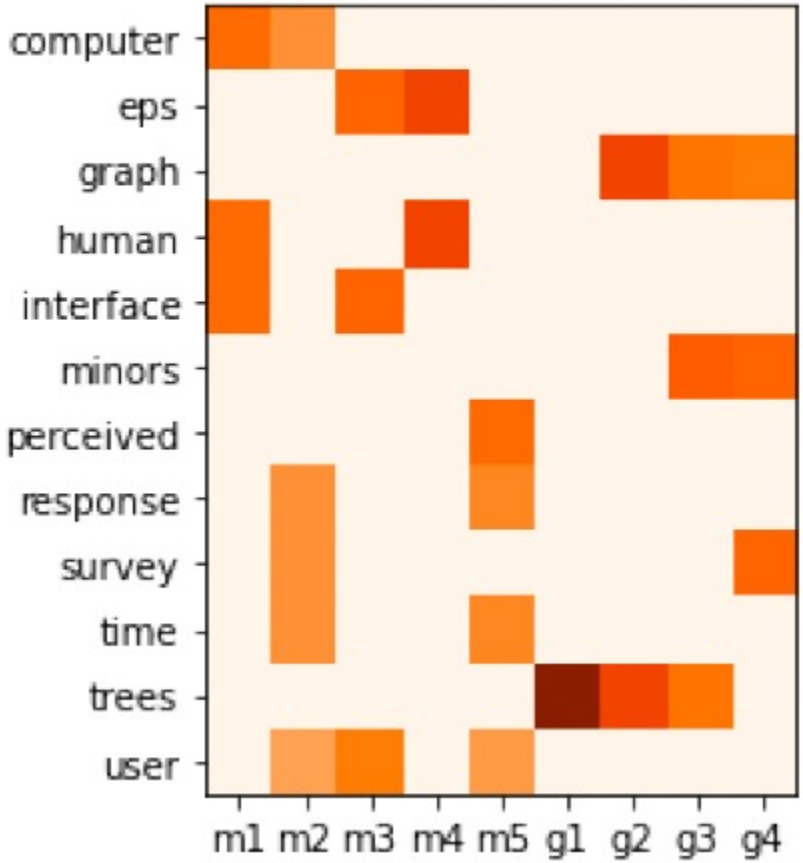
http://lsa.colorado.edu/papers/dp1.LSAintro.pdf

Credit: Jo Grundy

# Topic Modelling – NMF Examples

calculate TF.IDF

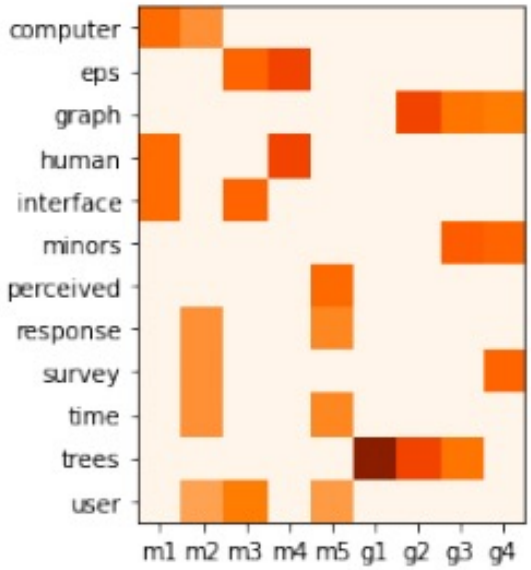
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.58 | 0.46 | 0. | 0. | 0. | 0. | 0. | 0. | 0. |
| 0. | 0. | 0.6 | 0.71 | 0. | 0. | 0. | 0. | 0. |
| 0. | 0. | 0. | 0. | 0. | 0. | 0.71 | 0.55 | 0.52 |
| 0.58 | 0. | 0. | 0.71 | 0. | 0. | 0. | 0. | 0. |
| 0.58 | 0. | 0.6 | 0. | 0. | 0. | 0. | 0. | 0. |
| 0. | 0. | 0. | 0. | 0. | 0. | 0. | 0.63 | 0.6 |
| 0. | 0. | 0. | 0. | 0.58 | 0. | 0. | 0. | 0. |
| 0. | 0.46 | 0. | 0. | 0.49 | 0. | 0. | 0. | 0. |
| 0. | 0.46 | 0. | 0. | 0. | 0. | 0. | 0. | 0.6 |
| 0. | 0.46 | 0. | 0. | 0.49 | 0. | 0. | 0. | 0. |
| 0. | 0. | 0. | 0. | 0. | 1. | 0.71 | 0.55 | 0. |
| 0. | 0.4 | 0.52 | 0. | 0.43 | 0. | 0. | 0. | 0. |



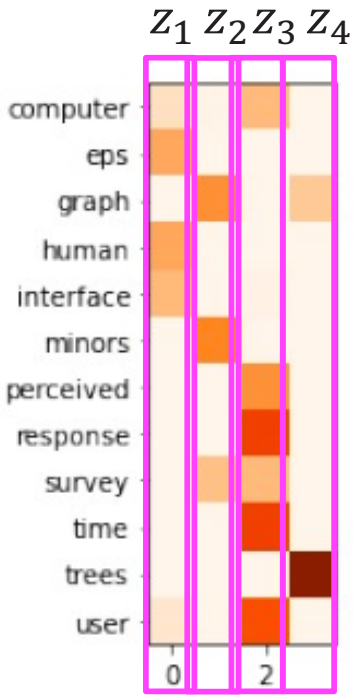

Credit: Jo Grundy

# Topic Modelling – NMF Examples

NMF: $A \approx WH$, $d = 4$

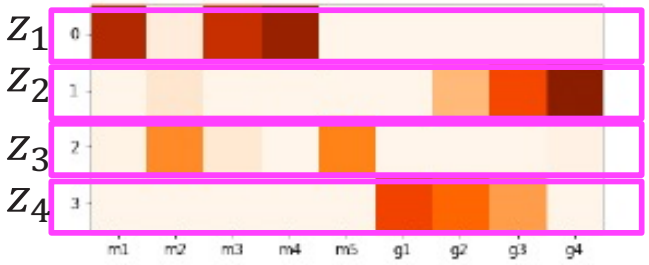



$A$




$W$

$z_i$: i-th topic




$H$

$W$ has the *basis vectors*, showing how the words are clustered
$H$ has the topic memberships for the documents.

Credit: Jo Grundy

# Topic Modelling – NMF Algorithm

For this method, and for LSA, the size of the reduced dimensionality is chosen manually

Can chose based on the error from reconstruction

though like $K$ means, and K nearest neighbours, this will be lower for higher values of $d$

Can run many times and build up a *consensus matrix*

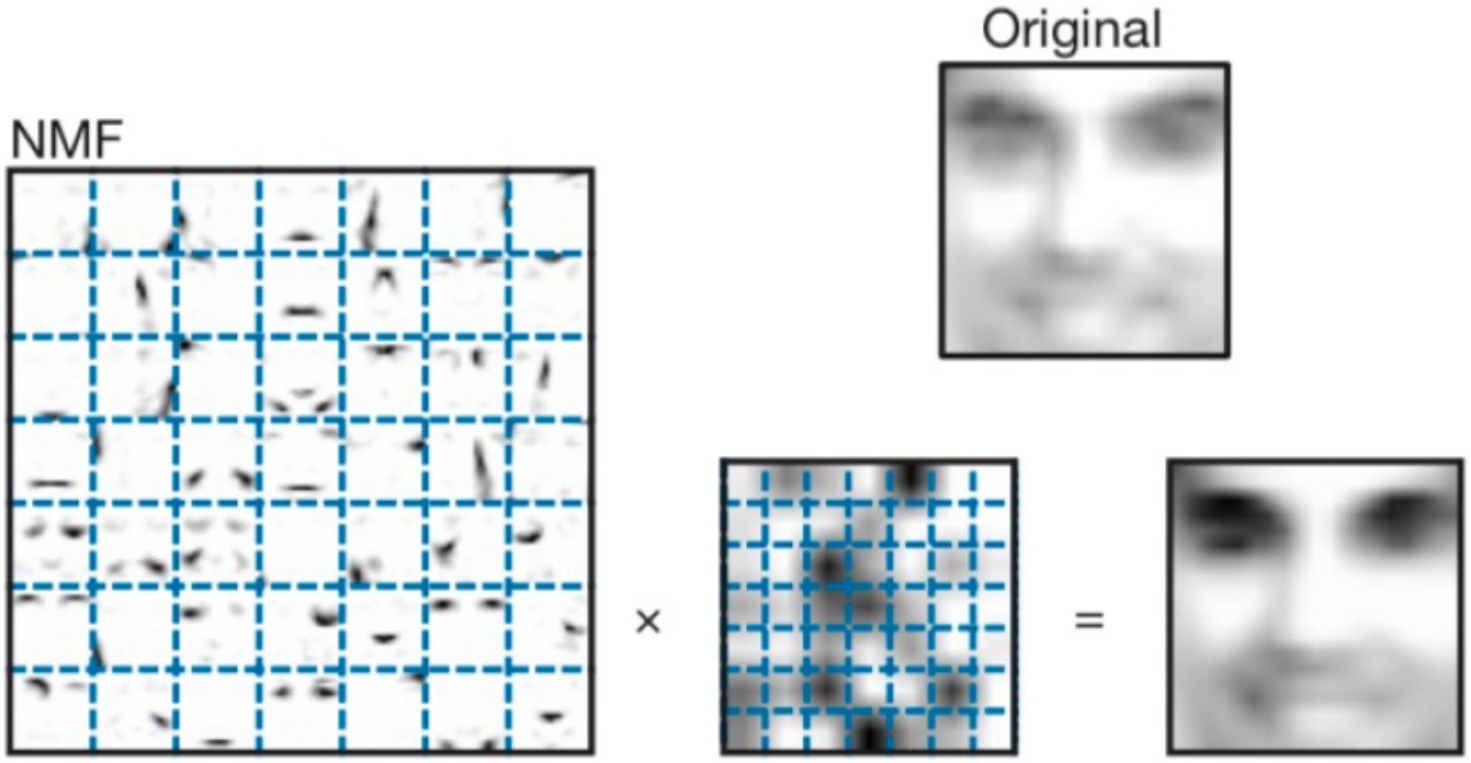
Can also examine the *stability* of multiple random initialised runs for each value of $d$



Credit: Jo Grundy

# Finding Features – NMF Examples

On a database of facial images, NMF constructed a decomposition of those faces in to parts, that H contained the weights to reconstruct every face from.



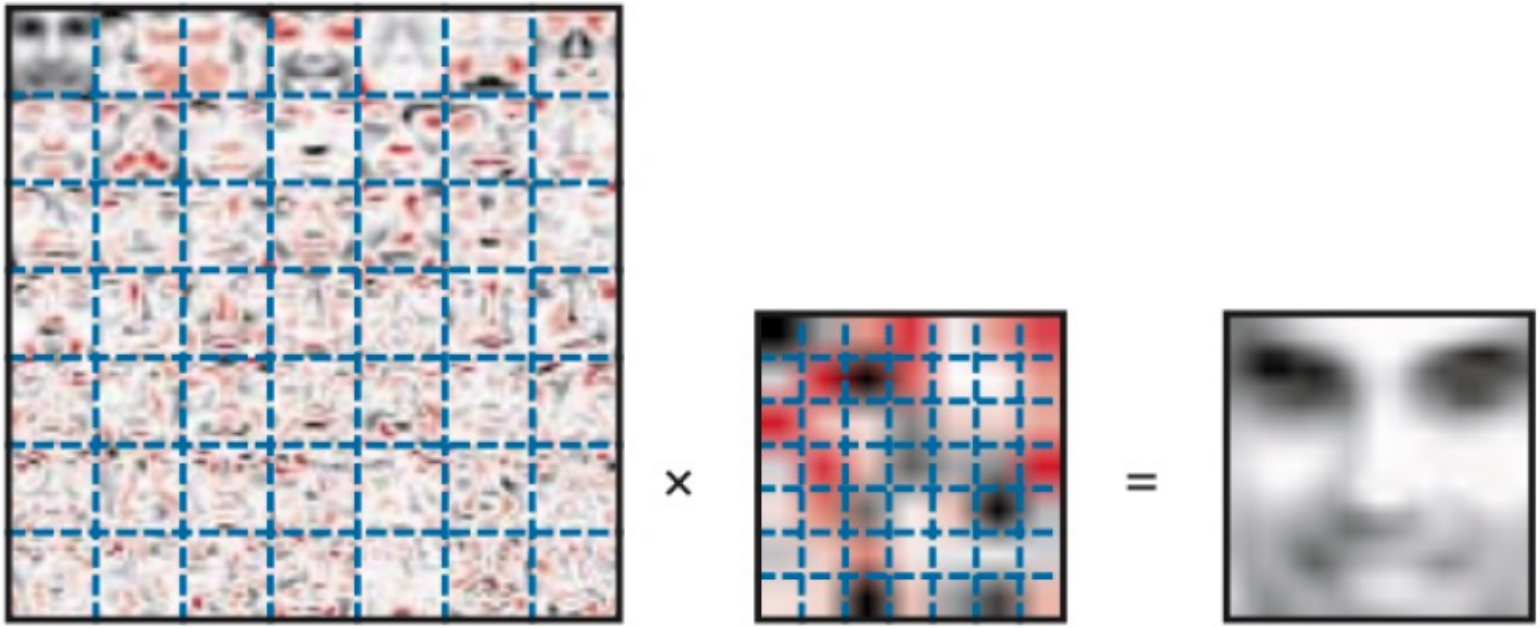

The face can be built up using a selection of the mouths, noses and eyes in the representation

Credit: Jo Grundy

# Finding Features – NMF Examples

Constraining the values to be non negative forces the representation to be sparse, as they must all be additive.

PCA

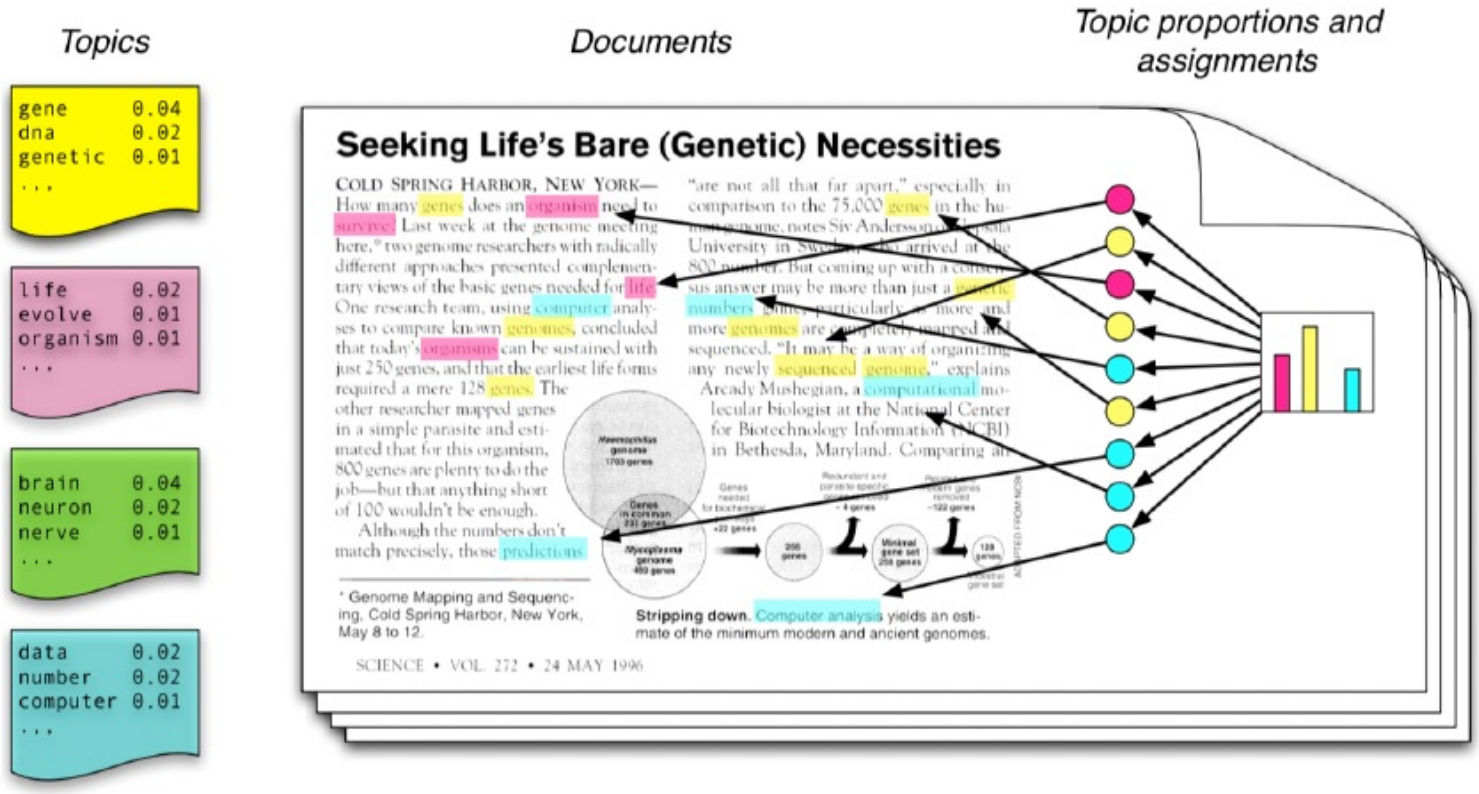


if PCA is used then the representation does not decompose the data

Credit: Jo Grundy

# Topic Modelling – Probabilistic Models

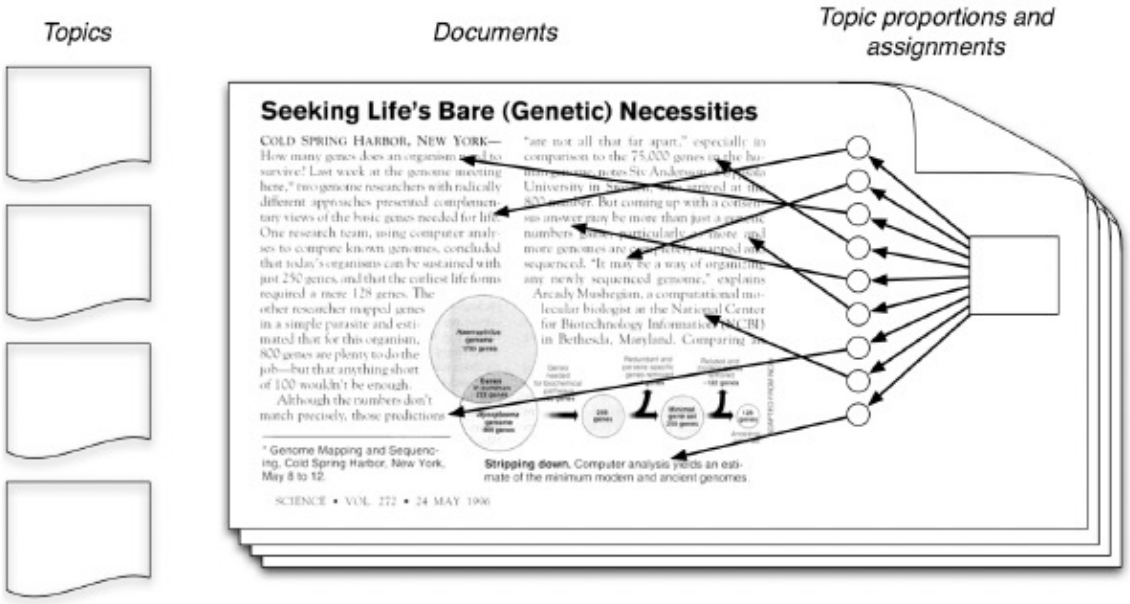
We model a **document** as a mixture of topics
A **topic** is a distribution over words
Each **word** in the document is drawn from a topic




Credit: Jo Grundy

# Topic Modelling – Probabilistic Models

In reality, only the document is visible
Topic distributions and assignments are *hidden*




We need to *infer* the hidden variables: i.e. compute the distribution conditioned on the documents
$p(topics, props, assignments|documents)$

Credit: Jo Grundy

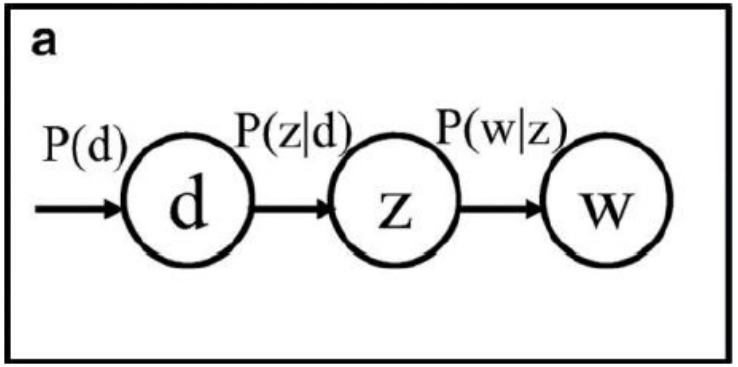
# Topic Modelling – PLSA

Probabilistic Latent Semantic Analysis

- given a corpus
- observations are pairs of words and documents $(w, d)$
- each observation is associated with latent class variable $c$

Assumes probability of a co-occurrence of a word and document $P(w, d)$ is a mixture of conditionally independent multinomial distributions

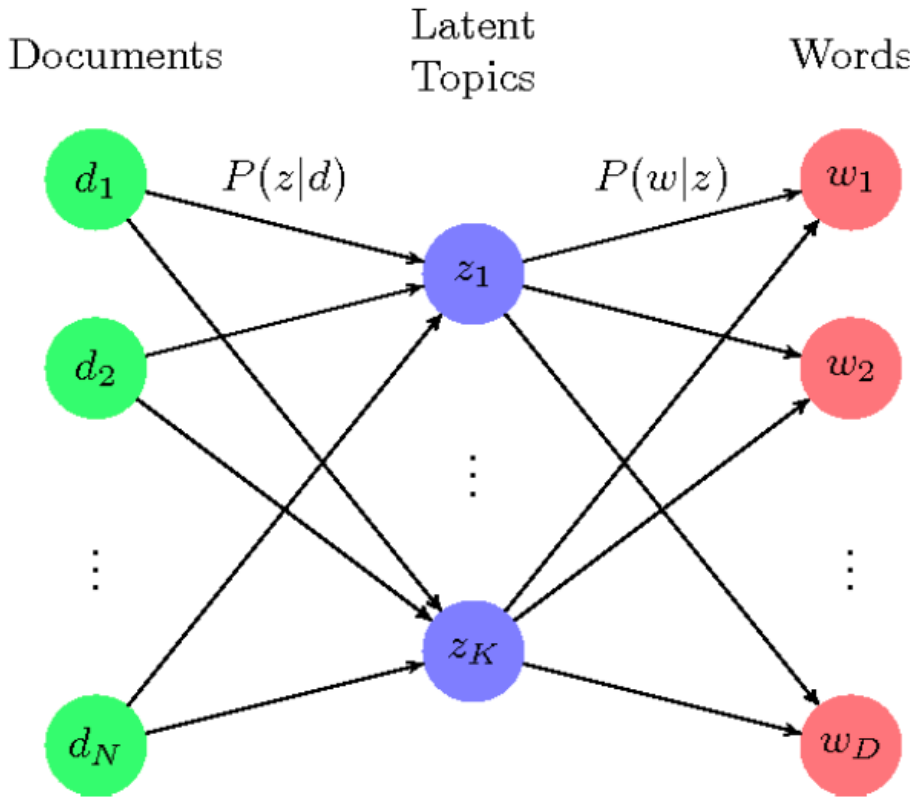
- Basic Generative Model
  - Select document $d$ with probability $P(d)$
  - Select a latent class $z$ with probability $P(z|d)$
  - Generate a word $w$ with probability $P(w|z)$
- Joint Probability Model




$$P(d,w) = P(d)P(w|d) \qquad P(w|d) = \sum_{z \in \mathbb{Z}} P(w|z)P(z|d)$$

Credit: Andrew Levandoski, Jonathan Lobo

# Topic Modelling – PLSA




Topic represented by probability distribution over words

$z_i = (w_1, \ldots, w_m)$ $\quad z_1 = (0.3, 0.1, 0.2, 0.3, 0.1)$

Document represented by probability distribution over topics

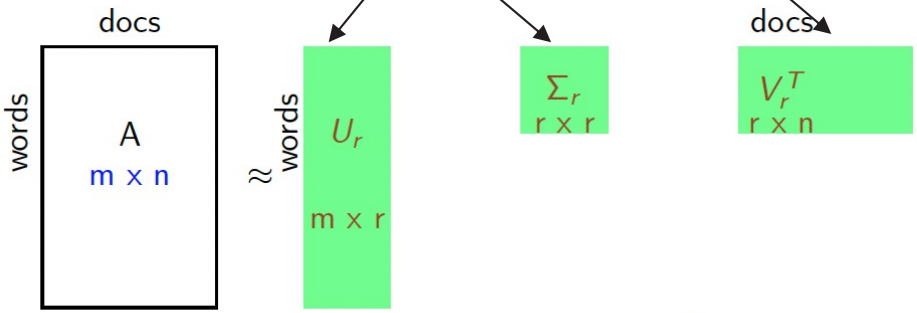
$d_j = (z_1, \ldots, z_n)$ $\quad d_1 = (0.5, 0.3, 0.2)$

$$P(d,w) = P(d)P(w|d)$$

$$P(w|d) = \sum_{z \in \mathbb{Z}} P(w|z)P(z|d)$$

$$P(d,w) = P(d) \sum_{z \in \mathbb{Z}} P(w|z)P(z|d)$$

$$= \sum_{z \in \mathbb{Z}} P(z)P(w|z)P(z|d)$$




$P(d)$ can be determined directly from our corpus. $P(w|z), P(z|d)$ are modeled as multinomial distributions, and can be trained using the expectation-maximization algorithm (EM)

Credit: Andrew Levandoski, Jonathan Lobo

# Topic Modelling – PLSA

- Incomplete in that it provides no probabilistic model at the document level i.e. no proper priors are defined.
- Each document is represented as a list of numbers (the mixing proportions for topics), and there is no generative probabilistic model for these numbers, thus:
  1. The number of parameters in the model grows linearly with the size of the corpus, leading to overfitting
  2. It is unclear how to assign probability to a document outside of the training set
- Latent Dirichlet allocation (LDA) captures the exchangeability of both words *and* documents using a Dirichlet distribution, allowing a coherent generative process for test data



Credit: Andrew Levandoski, Jonathan Lobo

# Topic Modelling – LDA

Latent Dirichlet Allocation (LDA) - also in AML
Bayesian Extension to PLSA

- Uses a Dirichlet prior on the topic distribution per document
- Fully Generative
- Baysian Inference to learn parameters
- Better than PLSA for small datasets, otherwise similar

*Proposed by David Blei, Andrew Ng, Michael Jordan, https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf*

In particular, it uses Dirichlet priors for the document-topic and word-topic distributions, lending itself to better generalization

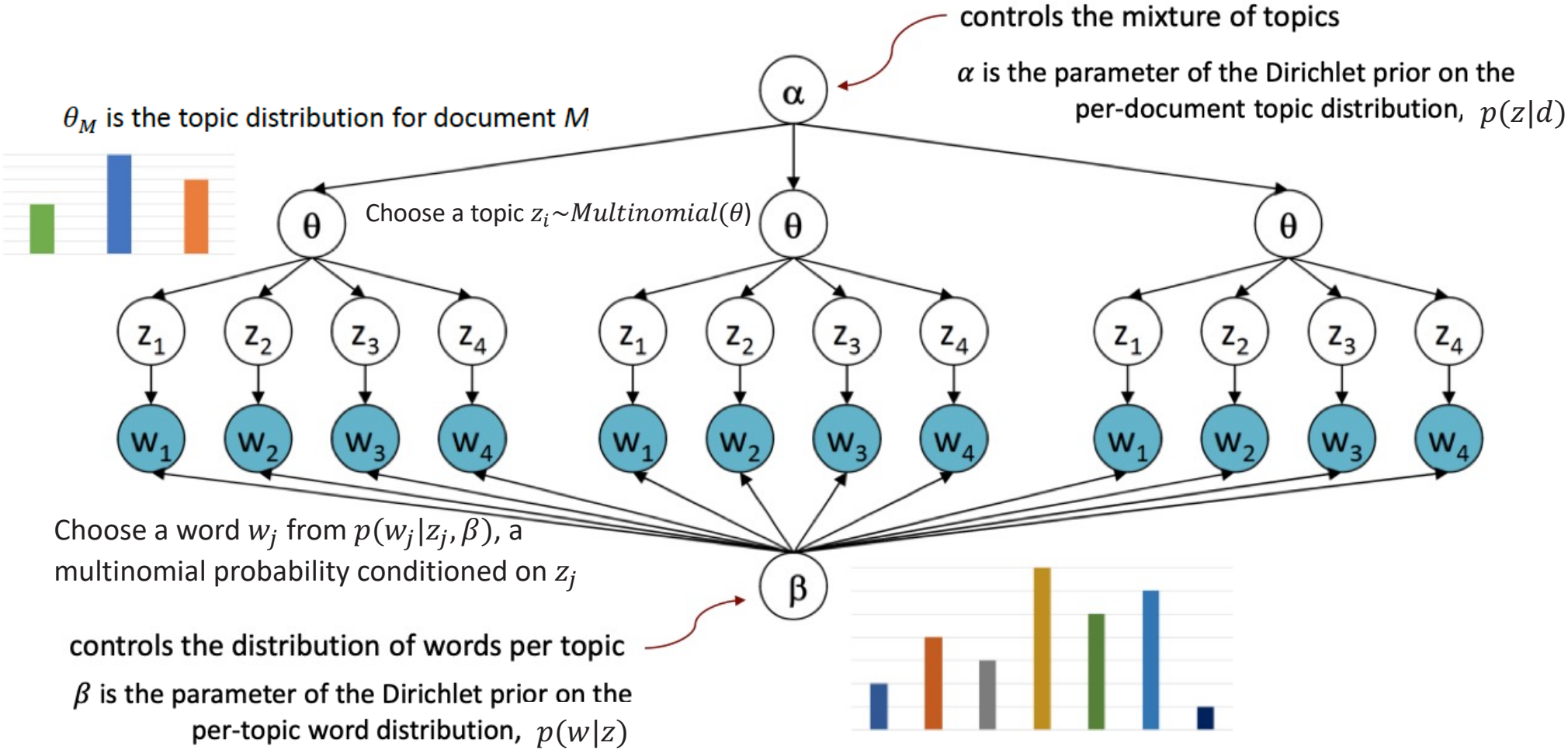
Generalize PLSA by changing the fixed *d* to a Dirichlet prior

Dirichlet Distribution: A 'distribution' of distribution'



Credit: Jo Grundy

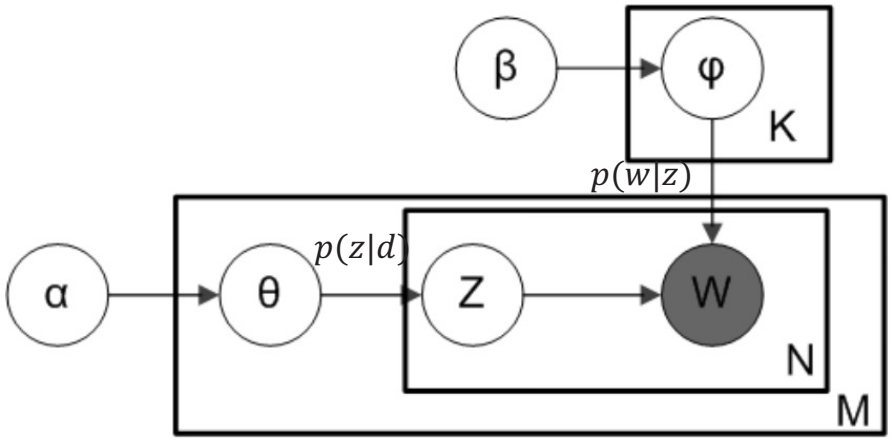
# Topic Modelling – LDA






Credit: Andrew Levandoski, Jonathan Lobo

# Topic Modelling – LDA



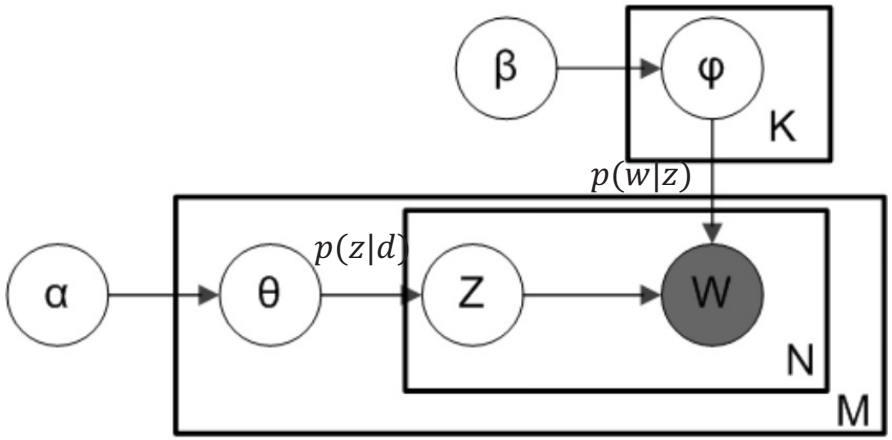

From a dirichlet distribution Dir(α), we draw a random sample representing the *topic distribution*, or topic mixture, of a particular document. This topic distribution is θ. From θ, we select a particular topic Z based on the distribution.

Next, from another dirichlet distribution Dir($\beta$), we select a random sample representing the *word distribution* of the topic Z. This word distribution is ϕ. From ϕ, we choose the word w.

Formally, the process for generating each word from a document is as follows (beware this algorithm uses *c* instead of *z* to represent the topic):

https://ppasupat.github.io/a9online/wtf-is/lsa-plsa-lda.html

# Topic Modelling – LDA




1. Choose $\theta_i \sim \mathrm{Dir}(\alpha)$ (where $i = 1, \ldots, M$; $\theta_i \in \Delta_K$)
   - $\theta_{i,k}$ = probability that document $i \in \{1, \ldots, M\}$ has topic $k \in \{1, \ldots, K\}$.
2. Choose $\phi_k \sim \mathrm{Dir}(\beta)$ (where $k = 1, \ldots, K$; $\phi_k \in \Delta_V$)
   - $\phi_{k,v}$ = probability of word $v \in \{1, \ldots, V\}$ in topic $k \in \{1, \ldots, K\}$.
3. Choose $c_{i,j} \sim \mathrm{Polynomial}(\theta_i)$ (where $c_{i,j} \in \{1, \ldots, K\}$)
4. Choose $w_{i,j} \sim \mathrm{Polynomial}(\phi_{c_{i,j}})$ (where $w_{i,j} \in \{1, \ldots, V\}$)

https://ppasupat.github.io/a9online/wtf-is/lsa-plsa-lda.html

# Topic Modelling – LDA vs. PLSA

LDA typically works better than PLSA because it can generalize to new documents easily

- In PLSA, the document probability is a fixed point in the dataset. If we haven't seen a document, we don't have that data point.
- In LDA, the dataset serves as training data for the Dirichlet distribution of document-topic distributions. If we haven't seen a document, we can easily sample from the Dirichlet distribution and move forward from there.

# Topic Modelling – LDA

Science articles, 17,000 documents, stop words and rare words removed

100 topic LDA using variational inference


Credit: Jo Grundy https://www.cs.columbia.edu/%7Eblei/talks/Blei_MLSS_2012.pdf

University of Southampton

# Topic Modelling – Summary

Topic modelling is an important part of data mining unstructured data

Key Ideas:

- Items are made up from topics
- A small subset of topics for each item
- One key parameter to tune: number of topics



Credit: Jo Grundy