# New Algorithms for Nonnegative Matrix Factorization and Beyond

**Ankur Moitra**
Institute for Advanced Study
and Princeton University

Algorithmic Aspects of Machine Learning

Note: These are unpolished, incomplete course notes.
Developed for educational use at MIT and for publication through MIT OpenCourseware.

# INFORMATION OVERLOAD!

**Challenge:** develop tools for automatic comprehension of data

This image has been removed due to copyright restrictions.
Please see: http://gretchenrubin.com/wp-content/uploads/2015/07/booksfillthescreen.jpg.

**Topic Modeling: (Dave Blei, etc.)**

- Discover hidden **topics**
- Annotate documents according to these topics
- Organize and summarize the collection

# INFORMATION OVERLOAD!

**Challenge:** develop tools for automatic comprehension of data

**Parceling Out a Nest Egg, Without Emptying It**

By PAUL SULLIVAN

What clients often forget are fixed costs — homes, cars, insurance — that must come down but take time to reduce, she said. Beyond that is her clients' skittish approach to risk; putting all of their money in cash may make them feel safe, she said, but it probably will not support the lifestyle they want for decades.

A generational disconnect is at work here: most people plan to retire at 65, the retirement age established for Social Security in 1935, when the average life expectancy was 61. Today the average is over 80 for men and women with a college degree.

So the $5.12 million gift exemption — created in a compromise between President Obama and Congress in 2010 — presents the well-off with a decision laden with short- and long-term consequences. How much should they give heirs now — and thus avoid giving the government in estate taxes later — while maintaining their lifestyle over a probably longer but still unpredictable remaining life span?

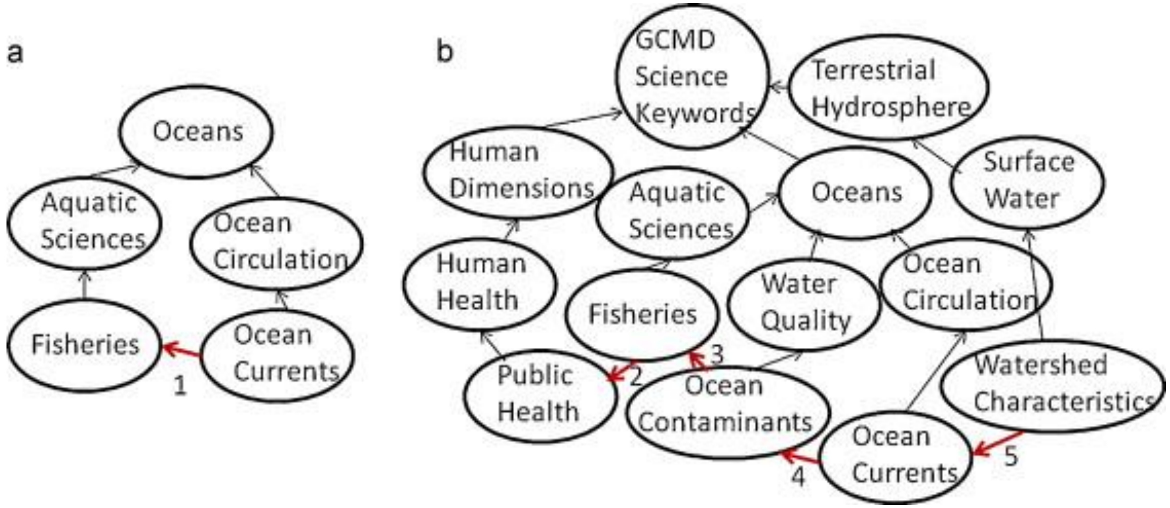


**Topic Modeling: (Dave Blei, etc.)**

- Discover hidden **topics**
- Annotate documents according to these topics
- Organize and summarize the collection

# INFORMATION OVERLOAD!

**Challenge:** develop tools for automatic comprehension of data






**Topic Modeling: (Dave Blei, etc.)**

- Discover hidden **topics**
- Annotate documents according to these topics
- Organize and summarize the collection

# INFORMATION OVERLOAD!

**Challenge:** develop tools for automatic comprehension of data

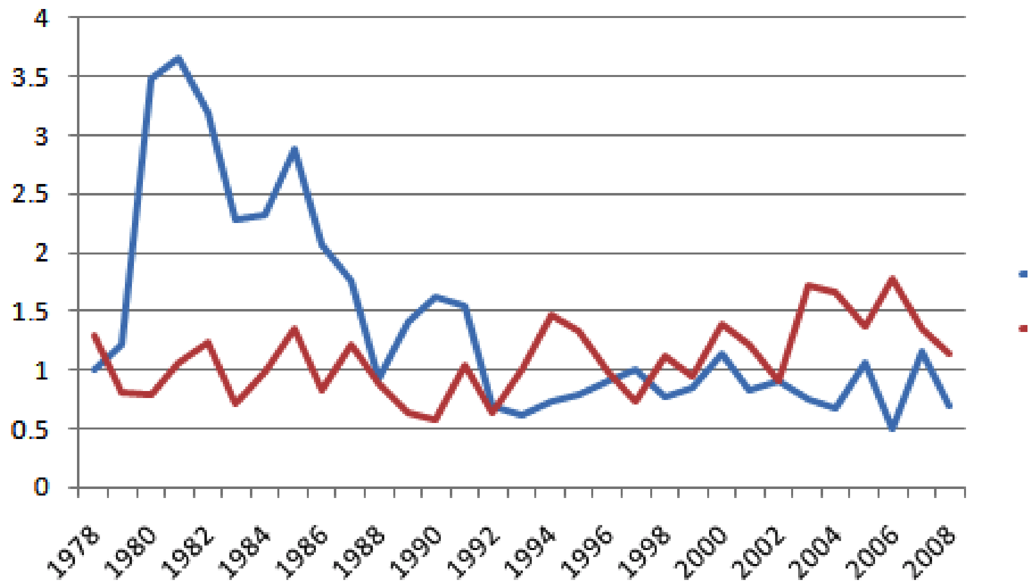



Image courtesy of the Stanford Natural Language Processing Group. Used with permission.

**Topic Modeling: (Dave Blei, etc.)**

- Discover hidden **topics**
- Annotate documents according to these topics
- Organize and summarize the collection

# Parceling Out a Nest Egg, Without Emptying It

By PAUL SULLIVAN

What clients often forget are fixed costs — homes, cars, insurance — that must come down but take time to reduce, she said. Beyond that is her clients' skittish approach to risk; putting all of their money in cash may make them feel safe, she said, but it probably will not support the lifestyle they want for decades.

A generational disconnect is at work here: most people plan to retire at 65, the retirement age established for Social Security in 1935, when the average life expectancy was 61. Today the average is over 80 for men and women with a college degree.

So the $5.12 million gift exemption — created in a compromise between President Obama and Congress in 2010 — presents the well-off with a decision laden with short- and long-term consequences. How much should they give heirs now — and thus avoid giving the government in estate taxes later — while maintaining their lifestyle over a probably longer but still unpredictable remaining life span?

**Personal Finance:** (money, 0.15), (retire, 0.10), (risk, 0.03) …

**Politics:** (President Obama, 0.10), (congress, 0.08), (government, 0.07), …

# Parceling Out a Nest Egg, Without Emptying It

By PAUL SULLIVAN

What clients often forget are fixed costs — homes, cars, insurance — that must come down but take time to reduce, she said. Beyond that is her clients' skittish approach to risk; putting all of their money in cash may make them feel safe, she said, but it probably will not support the lifestyle they want for decades.

A generational disconnect is at work here: most people plan to retire at 65, the retirement age established for Social Security in 1935, when the average life expectancy was 61. Today the average is over 80 for men and women with a college degree.

So the $5.12 million gift exemption — created in a compromise between President Obama and Congress in 2010 — presents the well-off with a decision laden with short- and long-term consequences. How much should they give heirs now — and thus avoid giving the government in estate taxes later — while maintaining their lifestyle over a probably longer but still unpredictable remaining life span?



- Each **document** is a distribution on **topics**
- Each **topic** is a distribution on words

# OUTLINE

Are there efficient algorithms to find the topics?

**Challenge:** We cannot **rigorously** analyze algorithms used in practice! (When do they work? run quickly?)

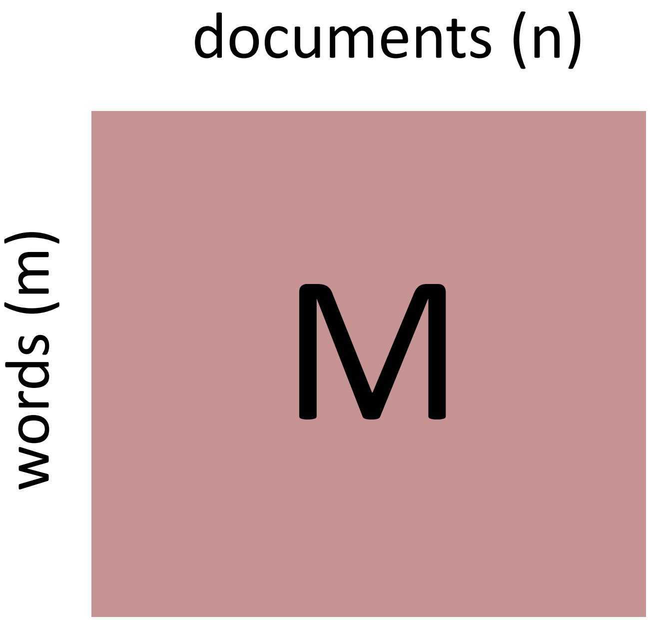
**Part I: An Optimization Perspective**

- Nonnegative Matrix Factorization
- Separability and Anchor Words
- Algorithms for Separable Instances

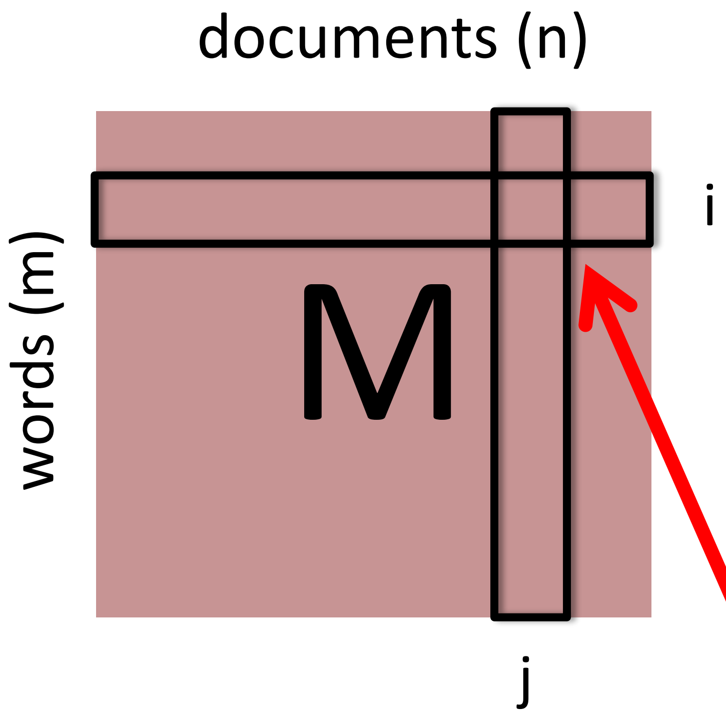
**Part II: A Bayesian Perspective**

- Topic Models (e.g. LDA, CTM, PAM, …)
- Algorithms for Inferring the Topics
- Experimental Results

# WORD-BY-DOCUMENT MATRIX

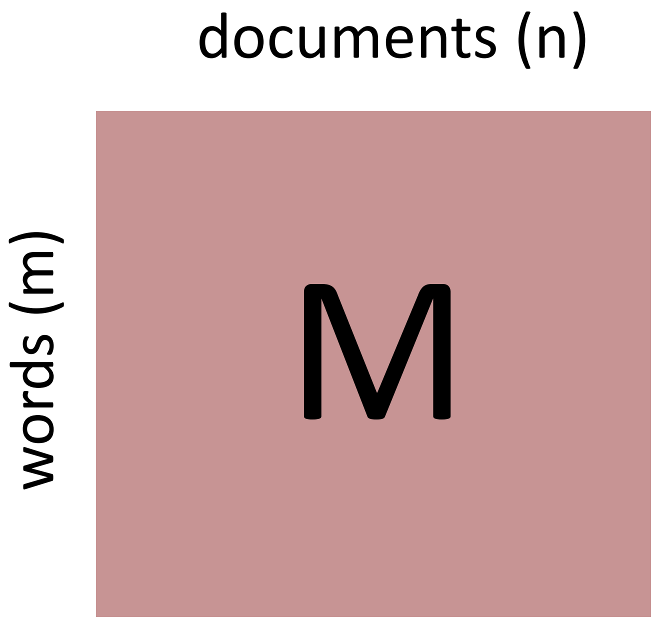


# WORD-BY-DOCUMENT MATRIX

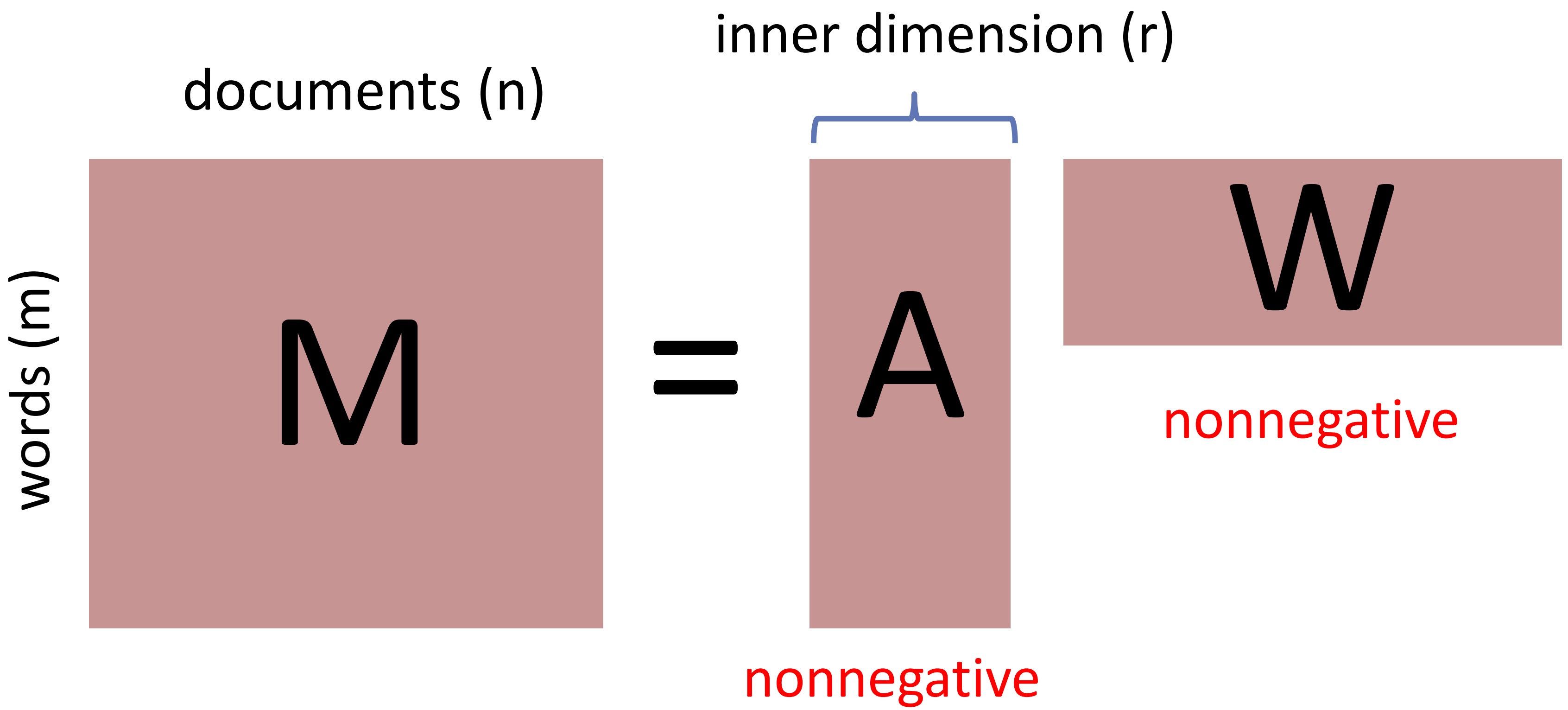



relative frequency of word i in document j

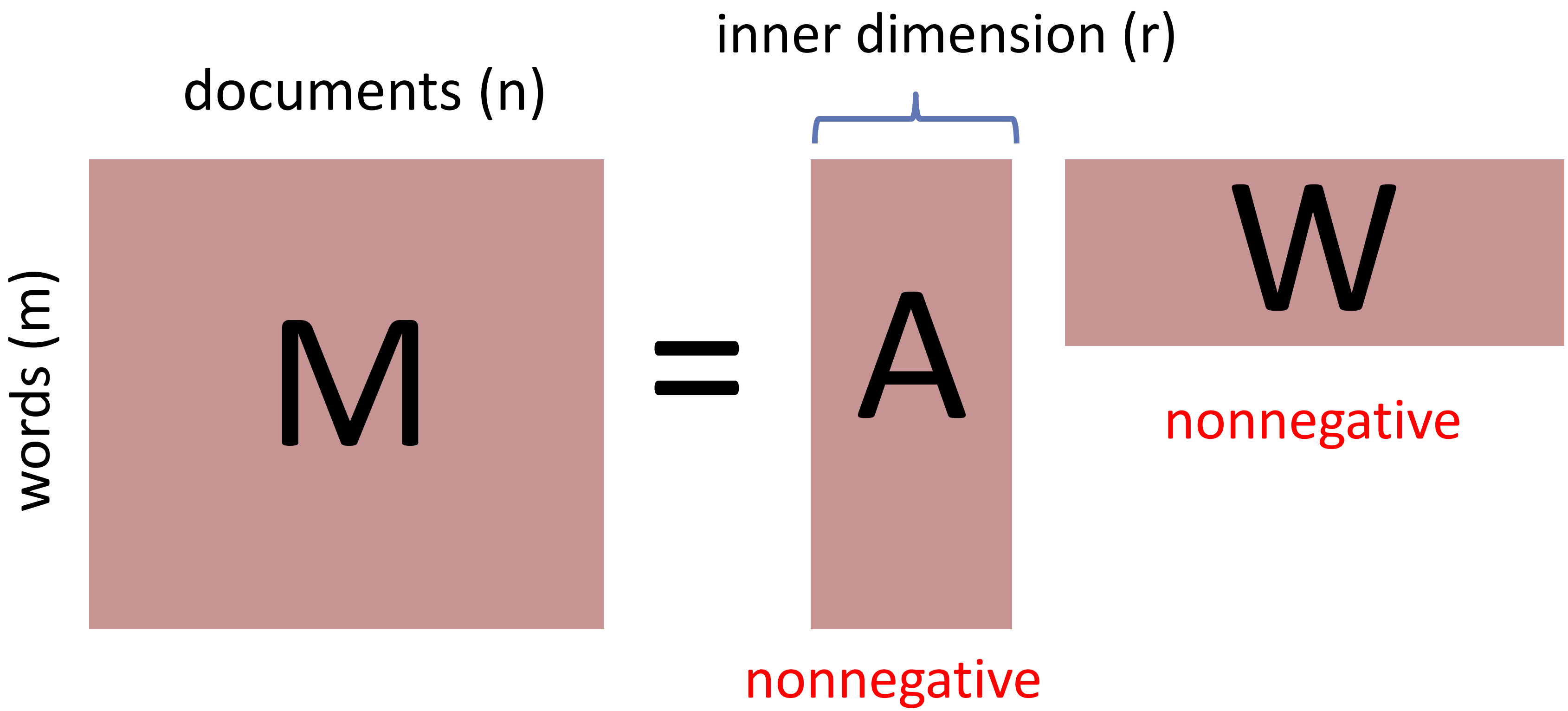
# WORD-BY-DOCUMENT MATRIX



# NONNEGATIVE MATRIX FACTORIZATION

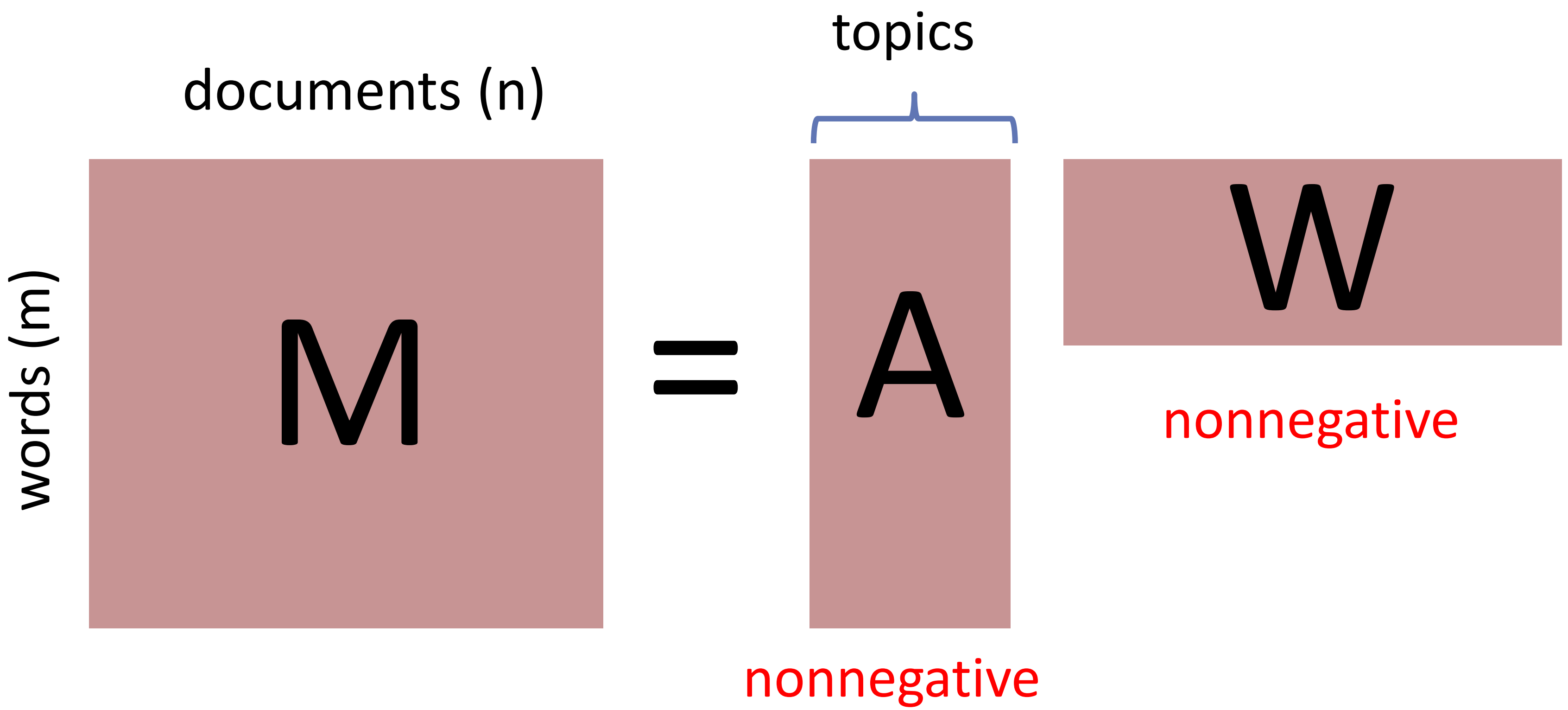


# NONNEGATIVE MATRIX FACTORIZATION




WLOG we can assume columns of **A, W** sum to one

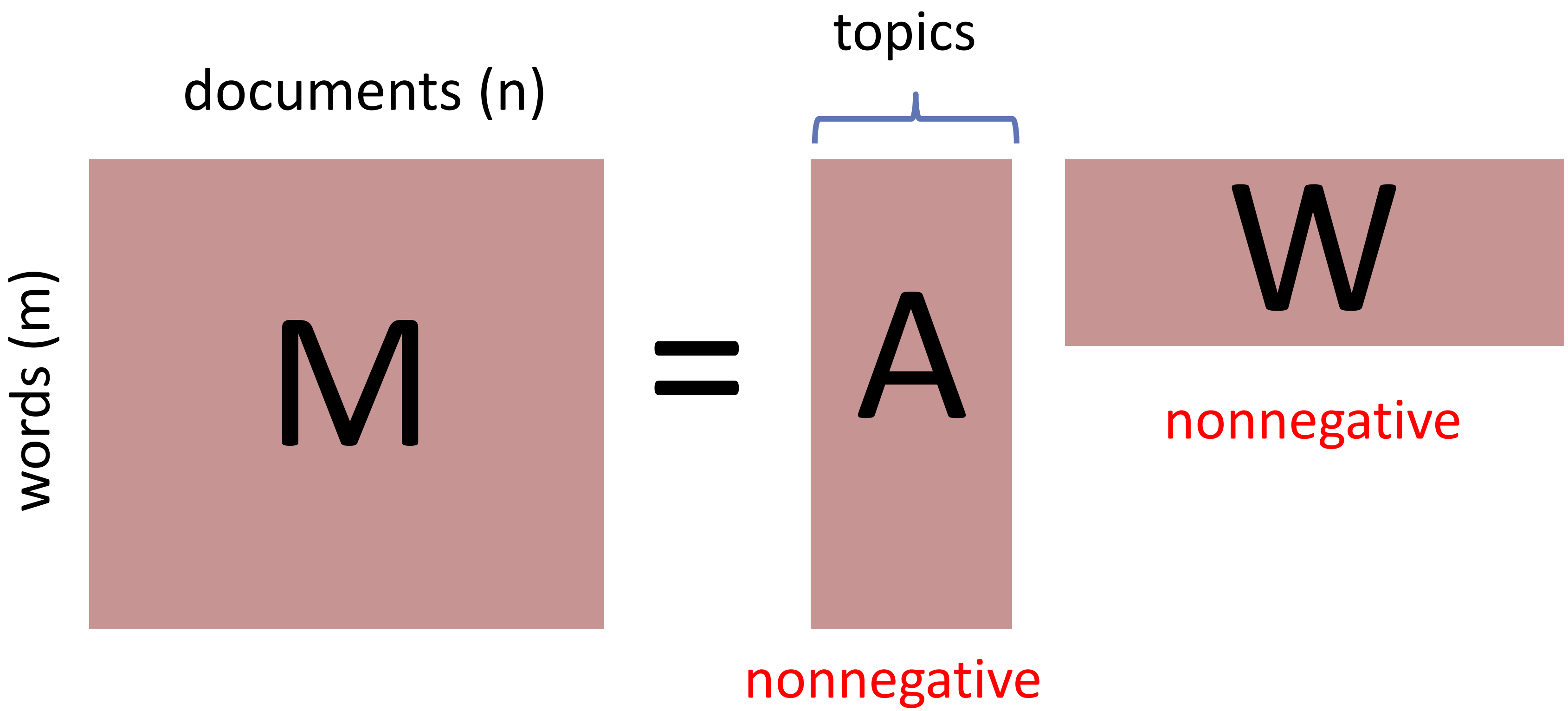
# NONNEGATIVE MATRIX FACTORIZATION




WLOG we can assume columns of **A, W** sum to one

# NONNEGATIVE MATRIX FACTORIZATION

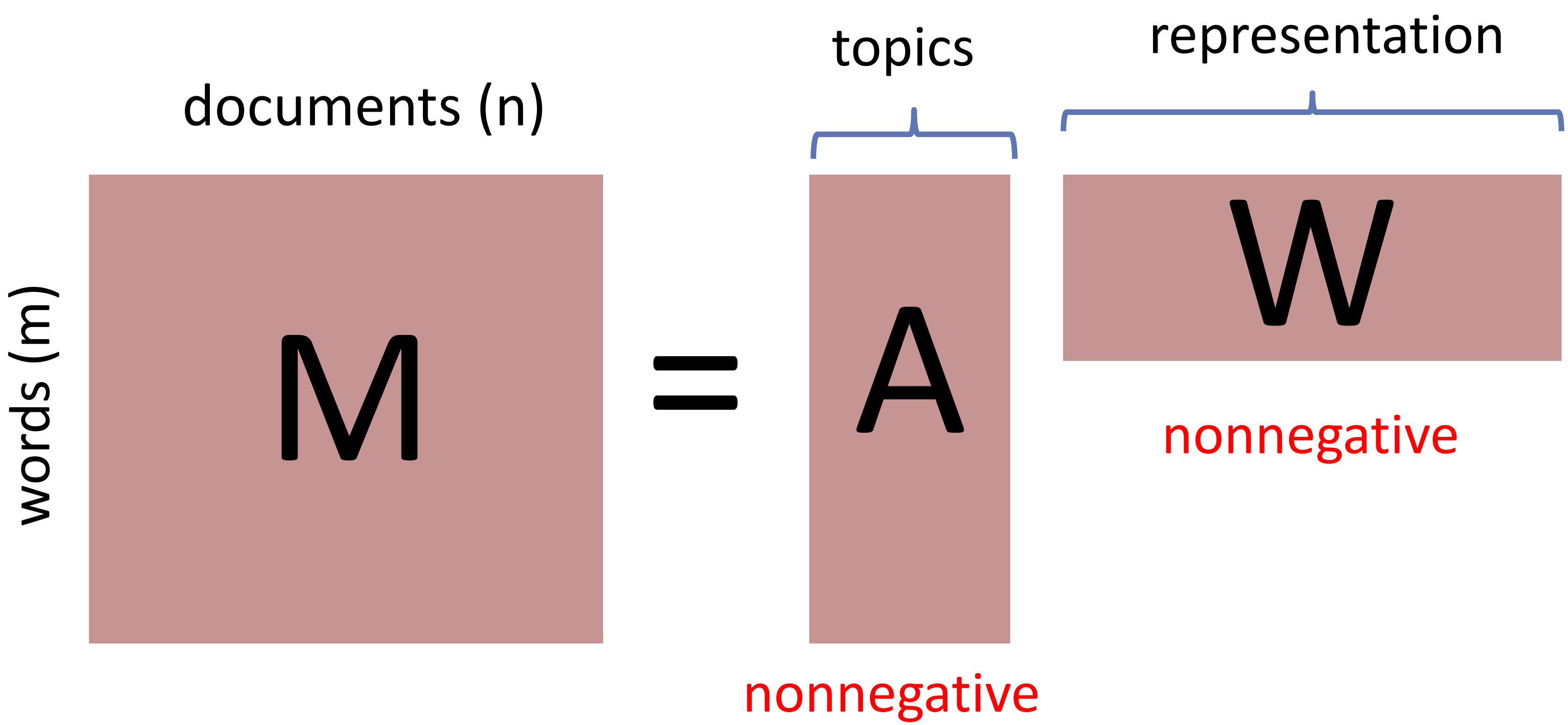
E.g. “personal finance”, (0.15, money), (0.10, retire), (0.03, risk), …




WLOG we can assume columns of **A, W** sum to one

# NONNEGATIVE MATRIX FACTORIZATION

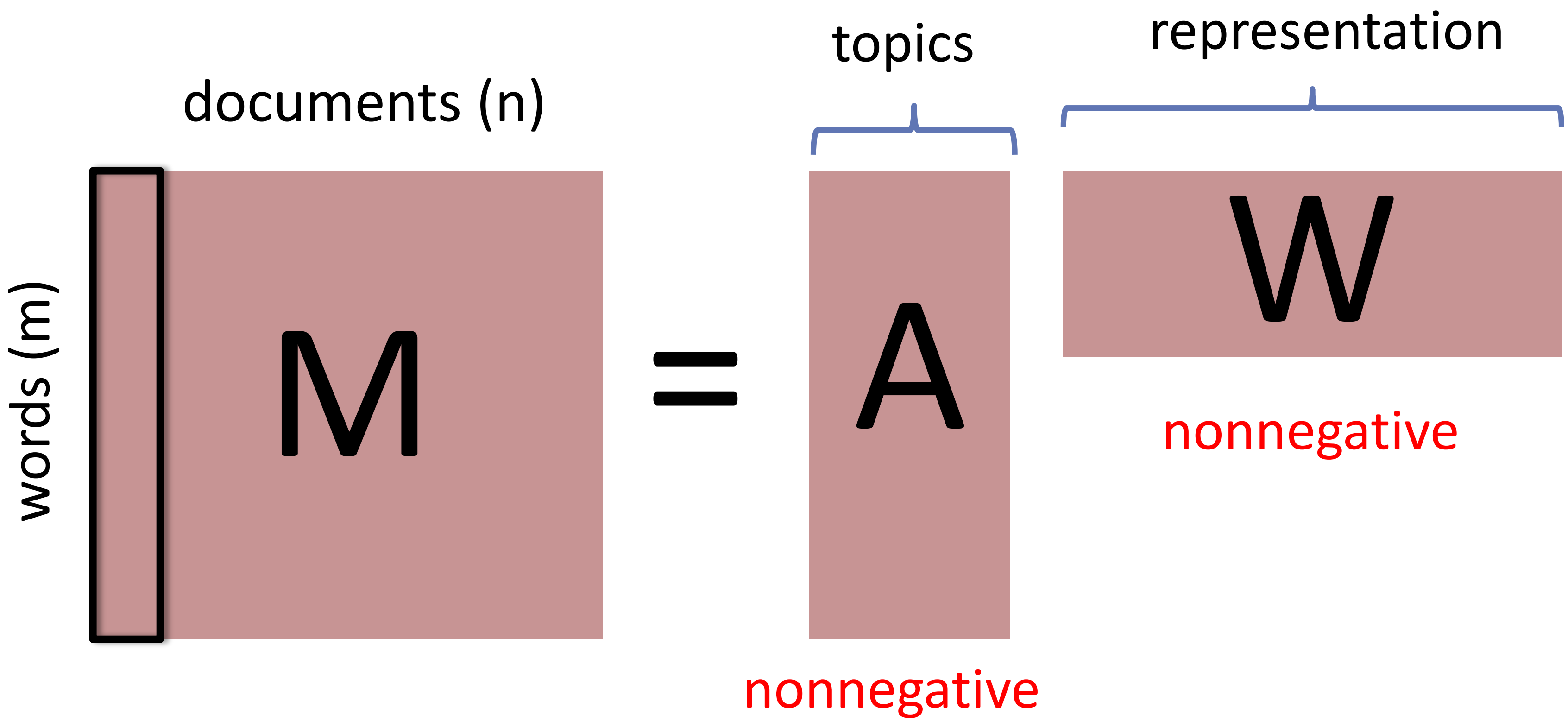
E.g. "personal finance", (0.15, money), (0.10, retire), (0.03, risk), …




WLOG we can assume columns of **A, W** sum to one

# NONNEGATIVE MATRIX FACTORIZATION

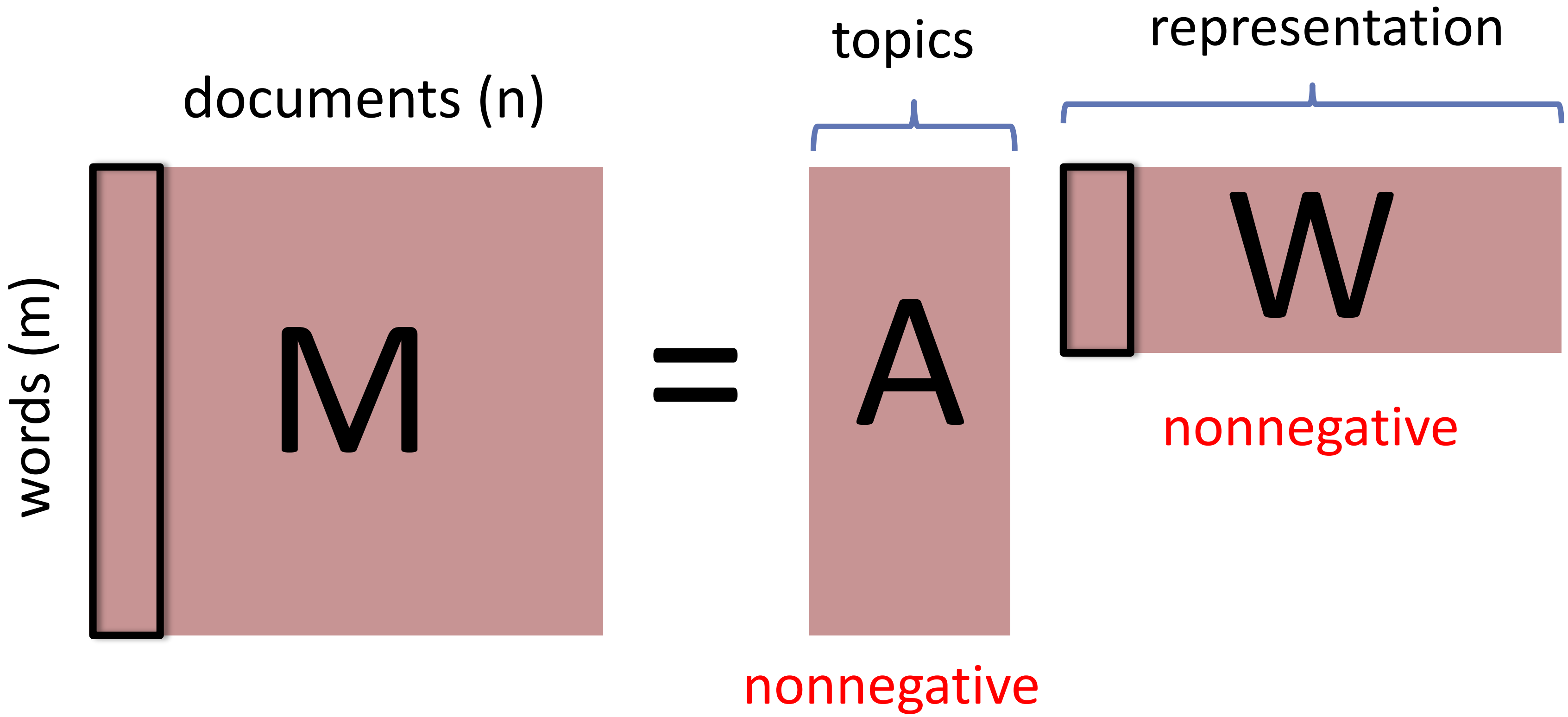
E.g. "personal finance", (0.15, money), (0.10, retire), (0.03, risk), …




WLOG we can assume columns of **A, W** sum to one

# NONNEGATIVE MATRIX FACTORIZATION

E.g. “personal finance”, (0.15, money), (0.10, retire), (0.03, risk), …




WLOG we can assume columns of **A, W** sum to one

# AN ABRIDGED HISTORY

**Machine Learning and Statistics:**

- Introduced by **[Lee, Seung, '99]**
- Goal: extract **latent** relationships in the data
- Applications to text classification, information retrieval, collaborative filtering, etc **[Hofmann '99], [Kumar et al '98], [Xu et al '03], [Kleinberg, Sandler '04],…**

**Theoretical Computer Science:**

- Introduced by **[Yannakakis '90]** in context of **extended formulations**; also related to the **log-rank conjecture**

**Physical Modeling:**

- Introduced by **[Lawton, Sylvestre '71]**
- Applications in chemometrics, environmetrics, economics

# ALGORITHMS FOR NMF?

**Local Search:** given **A**, compute **W**, compute **A**....

- known to fail on worst-case inputs (stuck in local optima)
- highly sensitive to cost-function, update procedure, regularization

Can we give an efficient algorithm that works on all inputs?

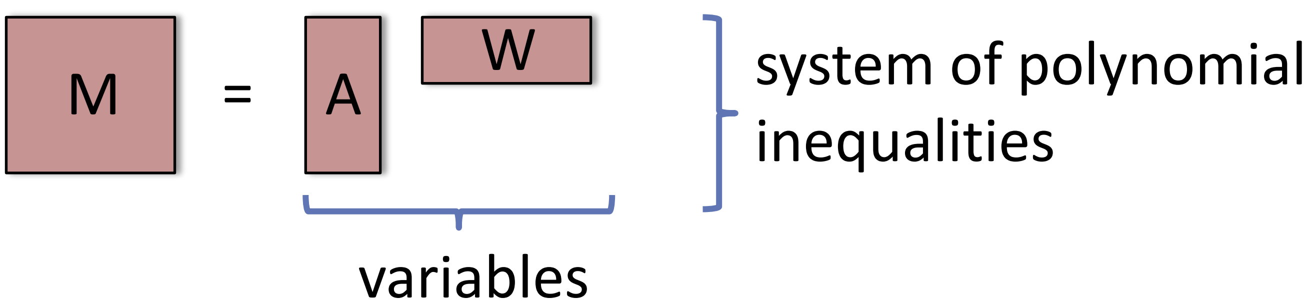
# WORST-CASE COMPLEXITY OF NMF

**Theorem [Vavasis '09]:** It is **NP**-hard to compute NMF

**Theorem [Cohen, Rothblum '93]:** Can solve NMF in time $(nm)^{O(nr+mr)}$

What is the complexity of NMF as a function of **r**?

**Theorem [Arora, Ge, Kannan, Moitra, STOC'12]:** Can solve NMF in time $(nm)^{O(r^2)}$ yet any algorithm that runs in time $(nm)^{o(r)}$ would yield a $2^{o(n)}$ algorithm for 3-SAT.




Can we reduce the number of variables from nr+mr to $O(r^2)$?

# ALGORITHMS FOR NMF?

**Local Search:** given **A**, compute **W**, compute **A**....

- known to fail on worst-case inputs (stuck in local optima)
- highly sensitive to cost-function, update procedure, regularization

Can we give an efficient algorithm that works on all inputs?

Yes, if and only if **r** is constant

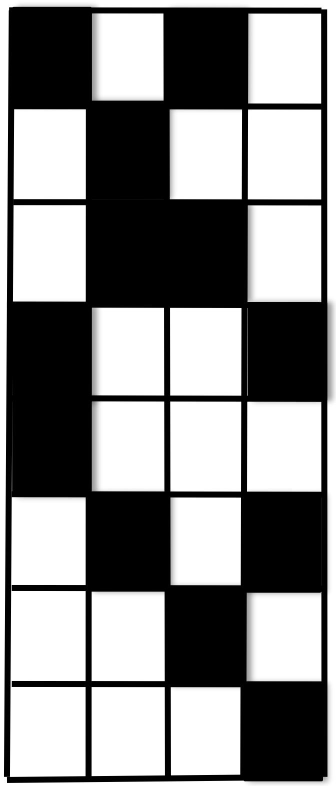
Are the instances we actually want to solve somehow easier?

Focus of this talk: a natural condition so that a **simple** algorithm **provably** works, **quickly**

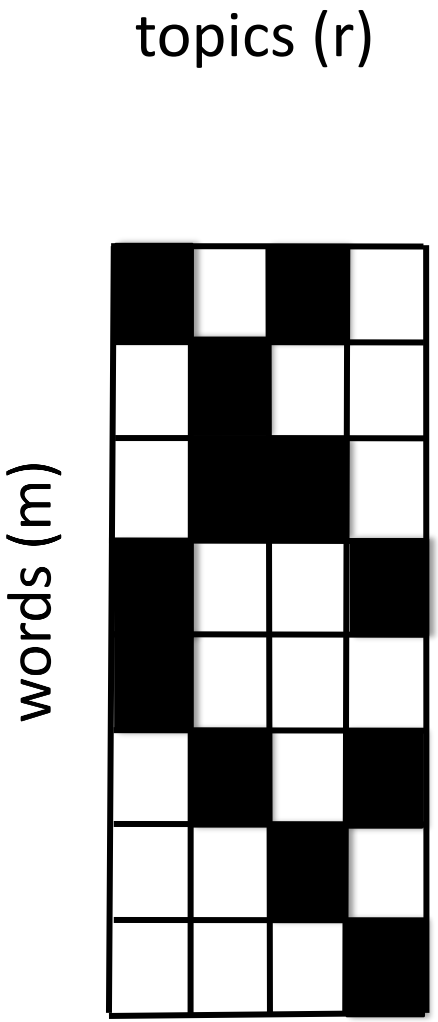
# SEPARABILITY AND ANCHOR WORDS

topics (r)

words (m)

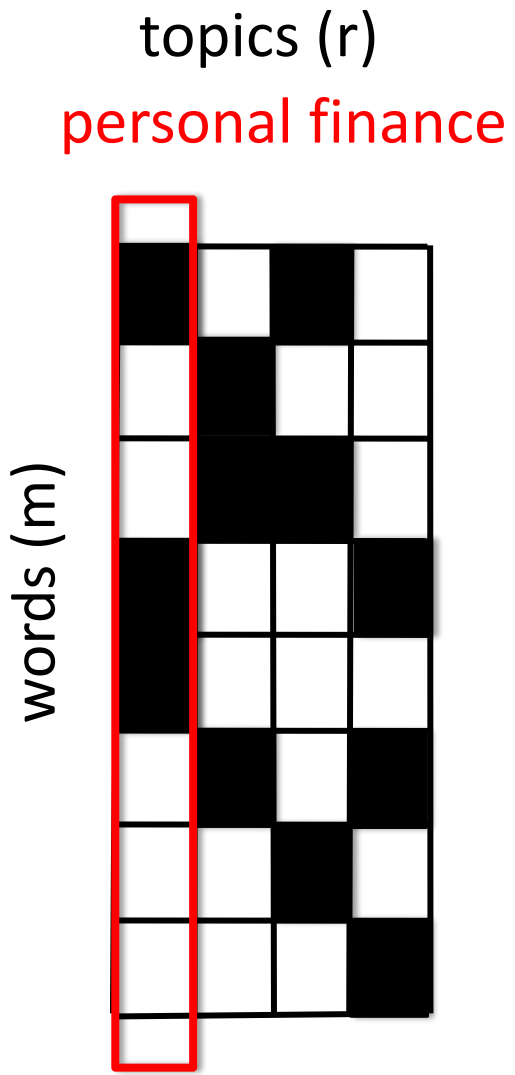


# SEPARABILITY AND ANCHOR WORDS



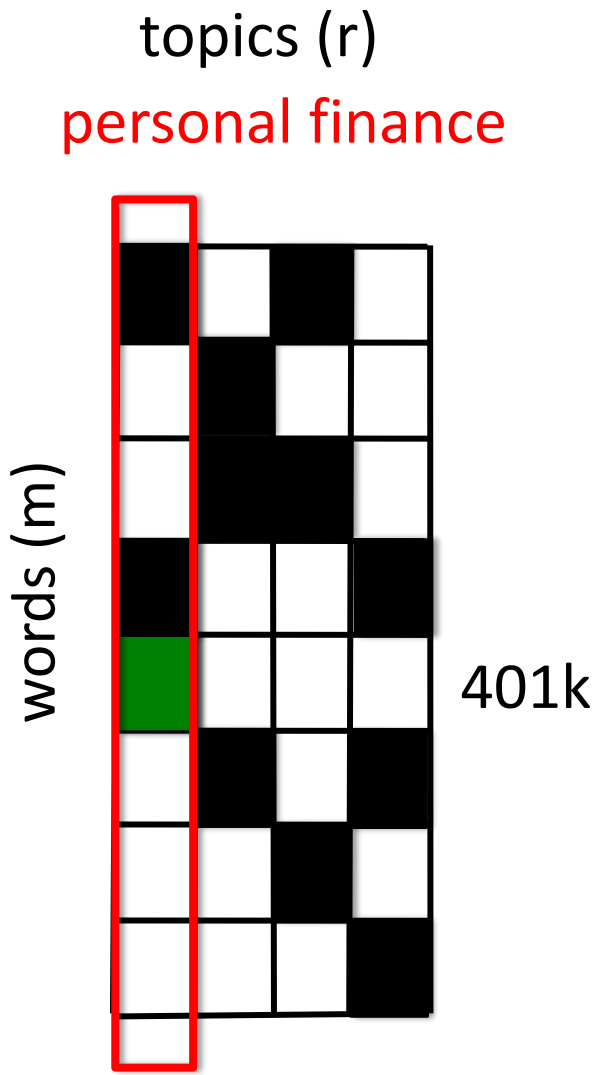

If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS




If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS

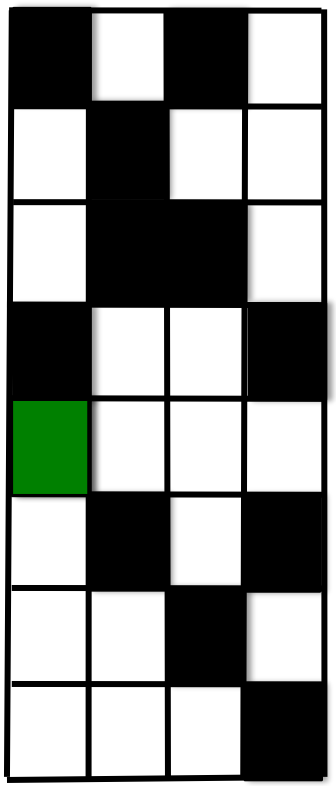



If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS

topics (r)

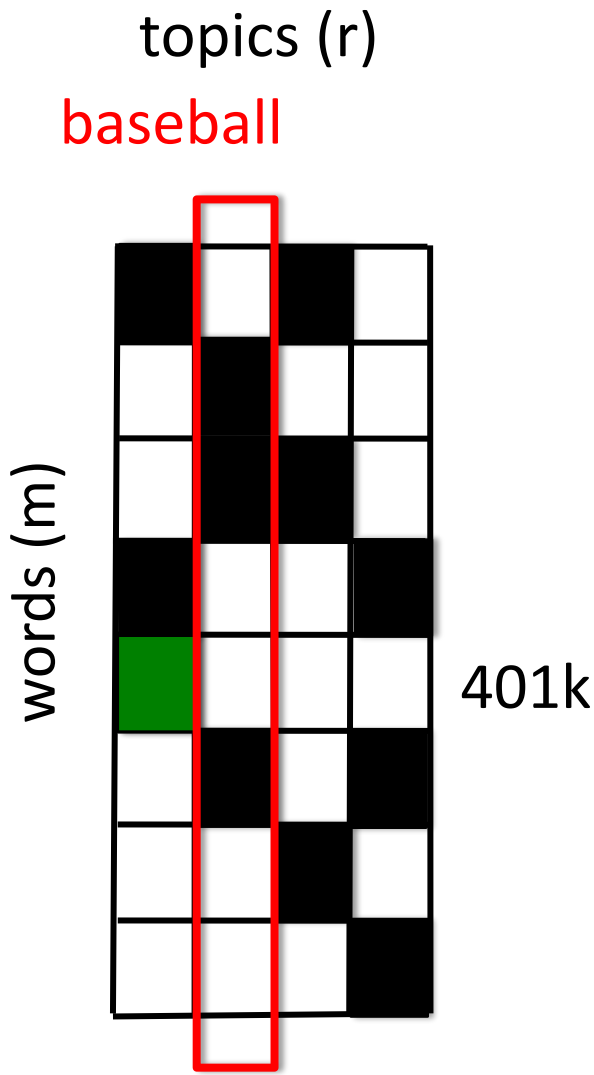
words (m)



401k

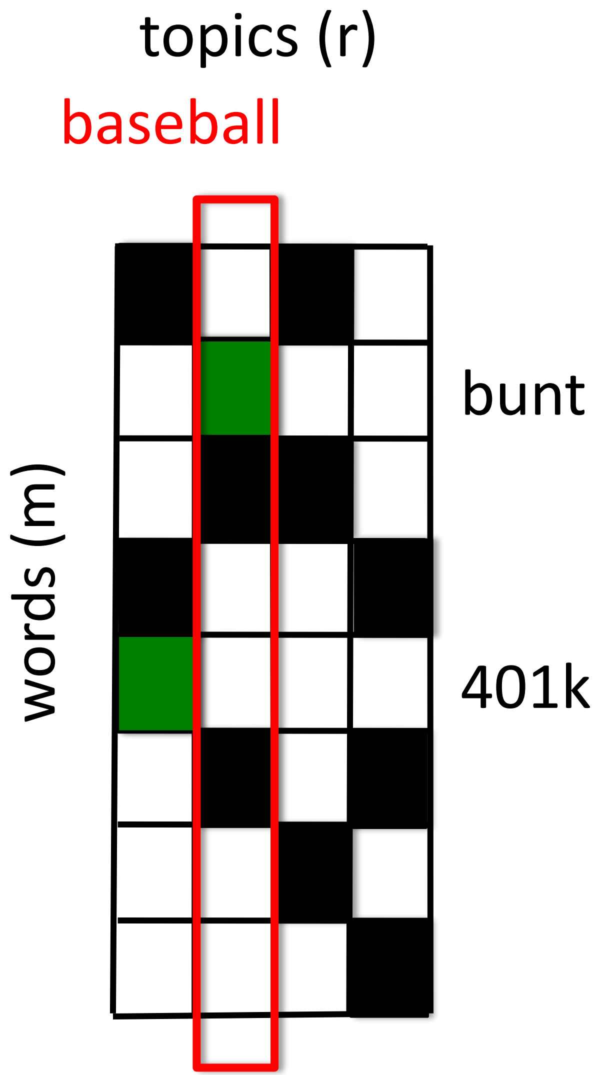
If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS



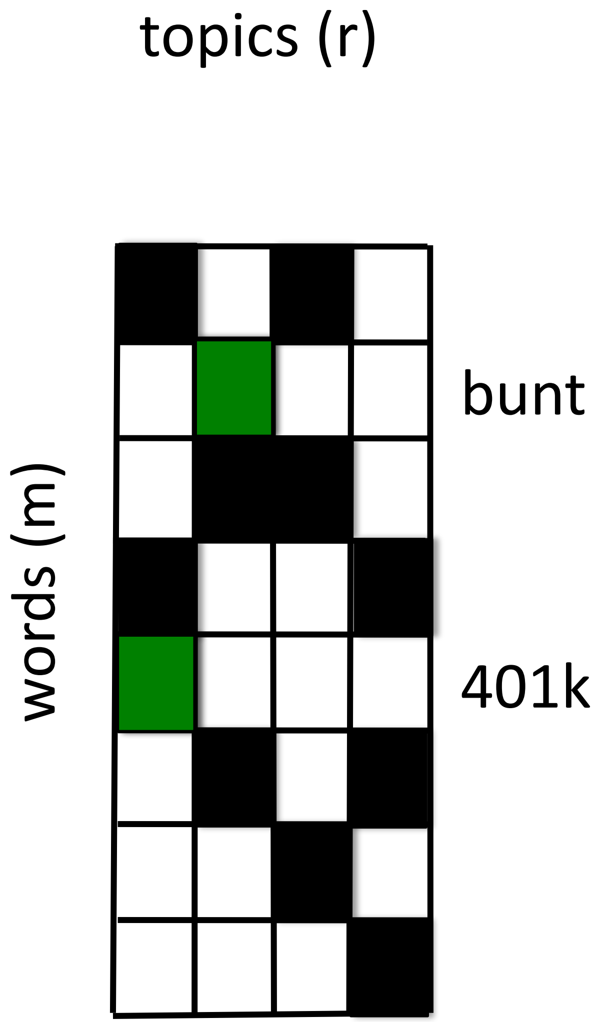

If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS



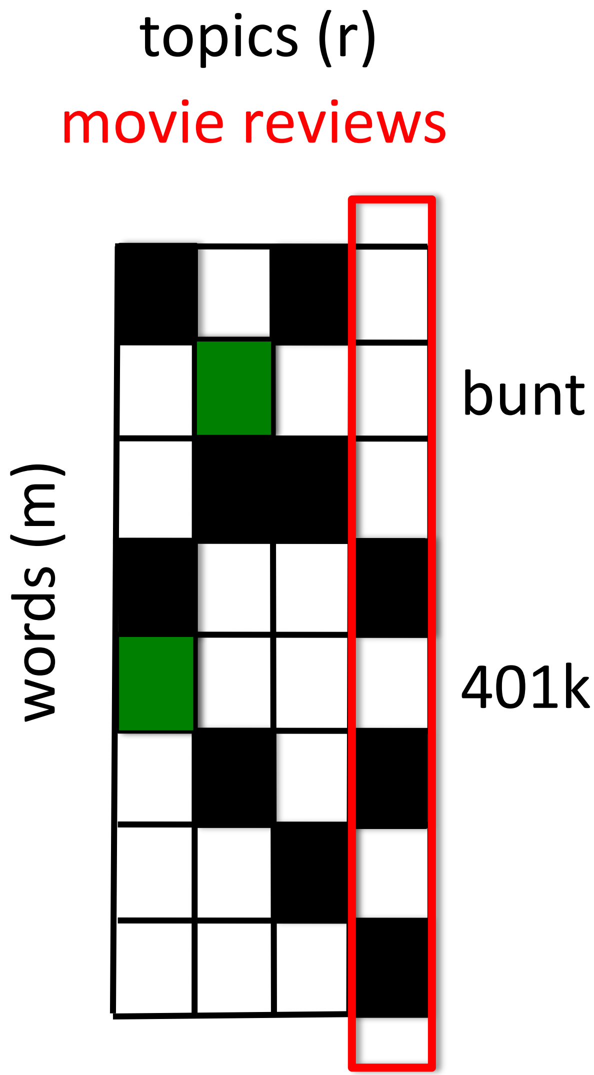

If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS



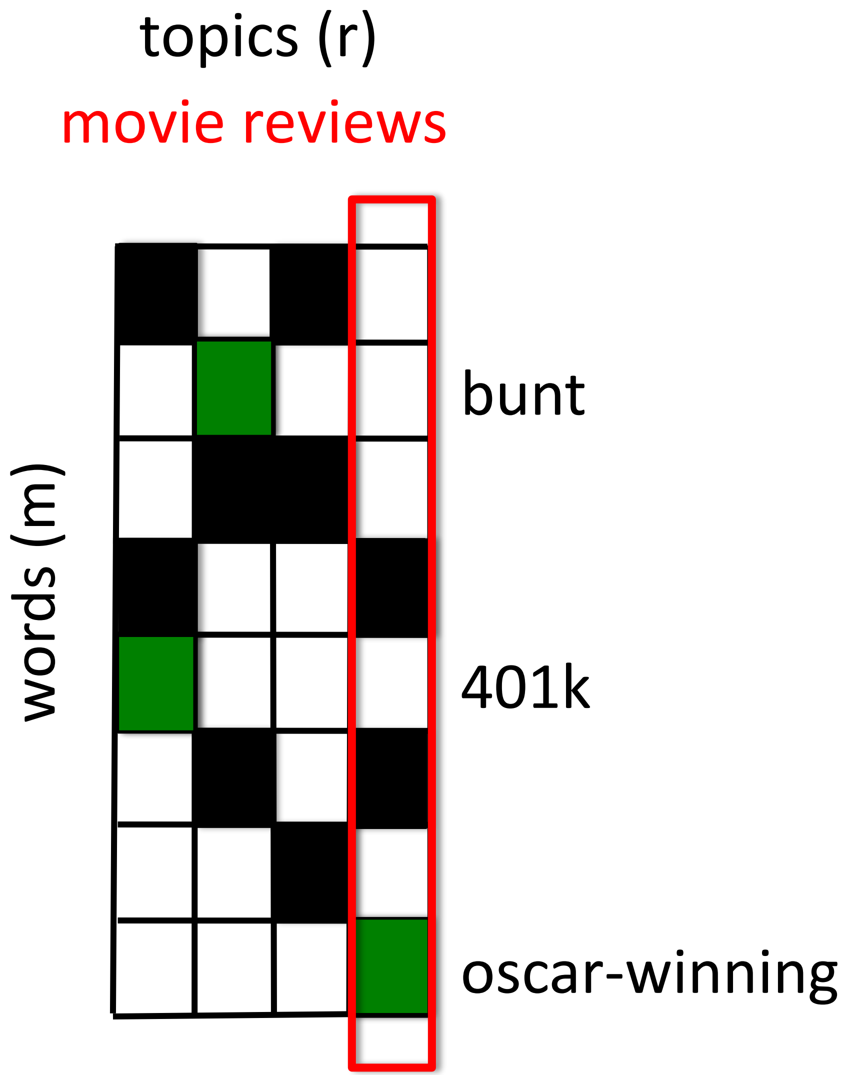

If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS



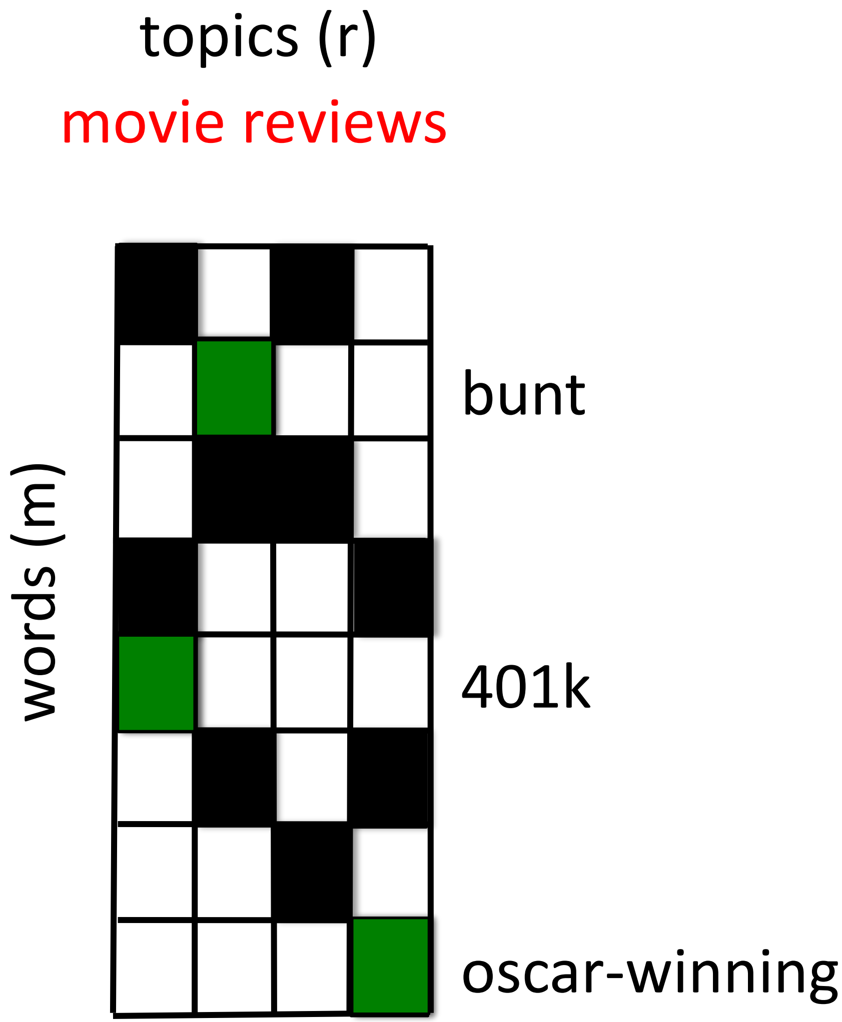

If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS



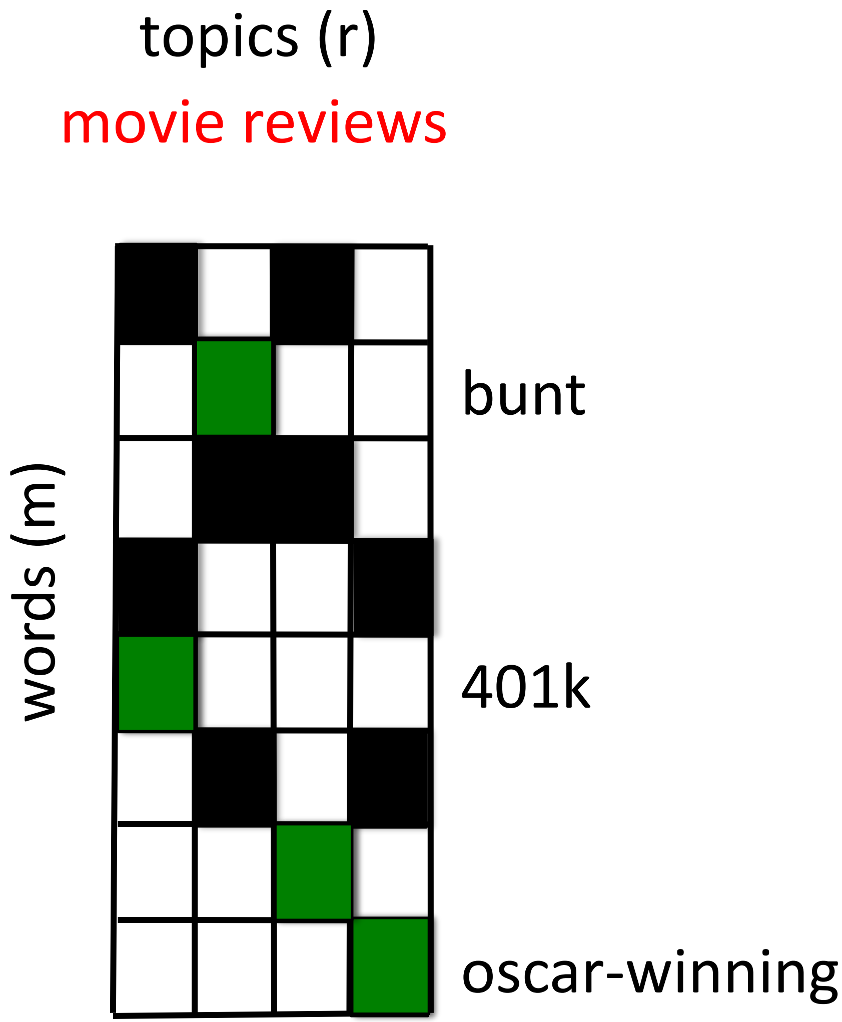

If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS




If an **anchor word** occurs then the document is at least partially about the topic

# SEPARABILITY AND ANCHOR WORDS




If an **anchor word** occurs then the document is at least partially about the topic

**A** is **p-separable** if each topic has an anchor word that occurs with probability ≥ p

**Theorem [Arora, Ge, Kannan, Moitra, STOC'12]:** There is an $O(nmr + mr^{3.5})$ time algorithm for NMF when the topic matrix **A** is separable

**Topic Models:** documents are **stochastically** generated as a convex combination of topics

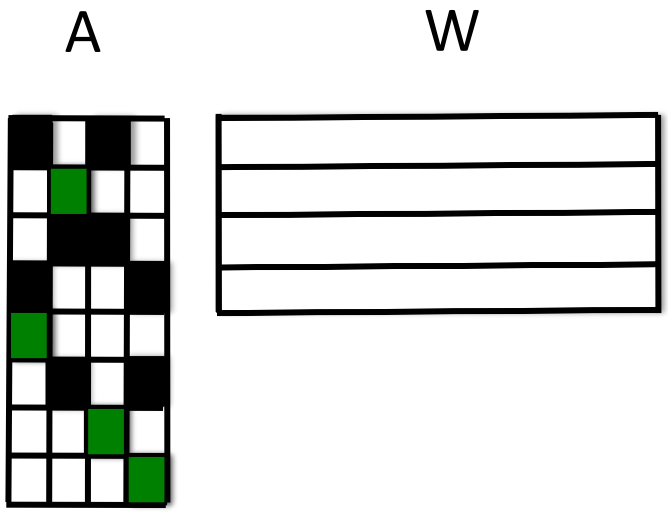
**Theorem [Arora, Ge, Moitra, FOCS'12]:** There is a polynomial time algorithm that learns the parameters of **any** topic model provided that the topic matrix **A** is p-separable.

In fact our algorithm is **highly practical**, and runs **orders of magnitude faster** with nearly-identical performance as the current best (Gibbs Sampling)

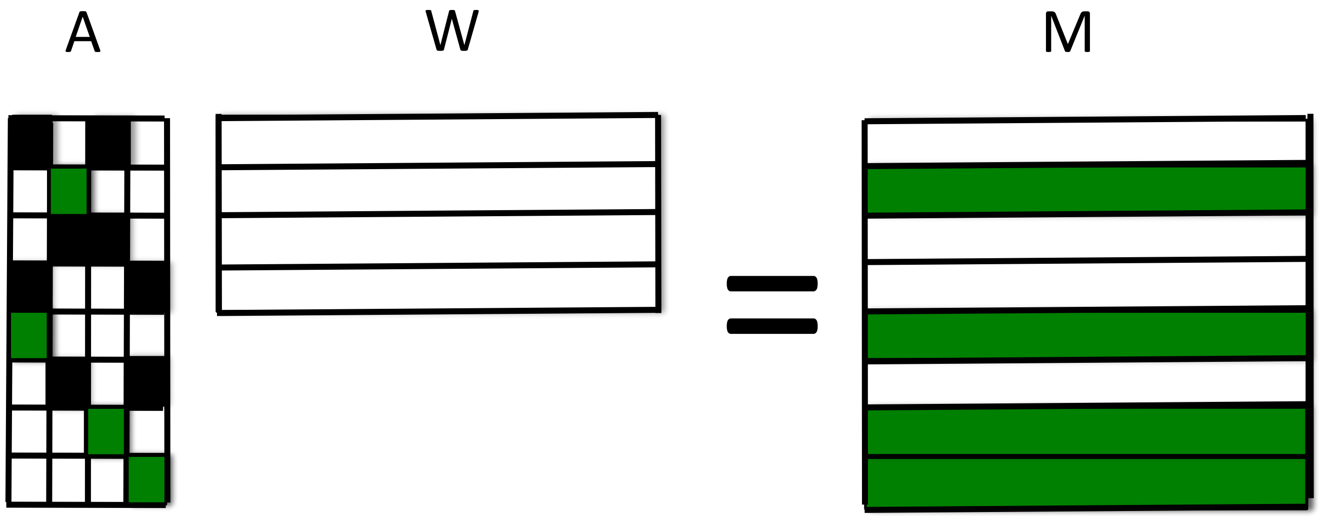
See also **[Anandkumar et al '12], [Rabani et al '12]** that give algorithms based on the method of moments

# How do anchor words help?

# ANCHOR WORDS ≅ VERTICES



# ANCHOR WORDS ≅ VERTICES

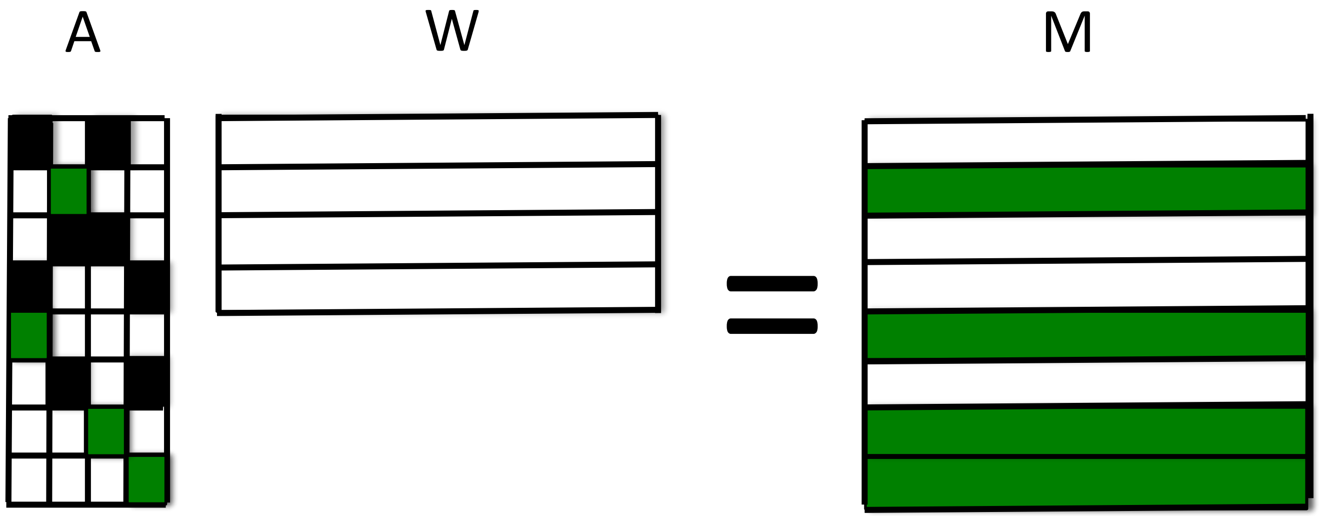


## How do anchor words help?

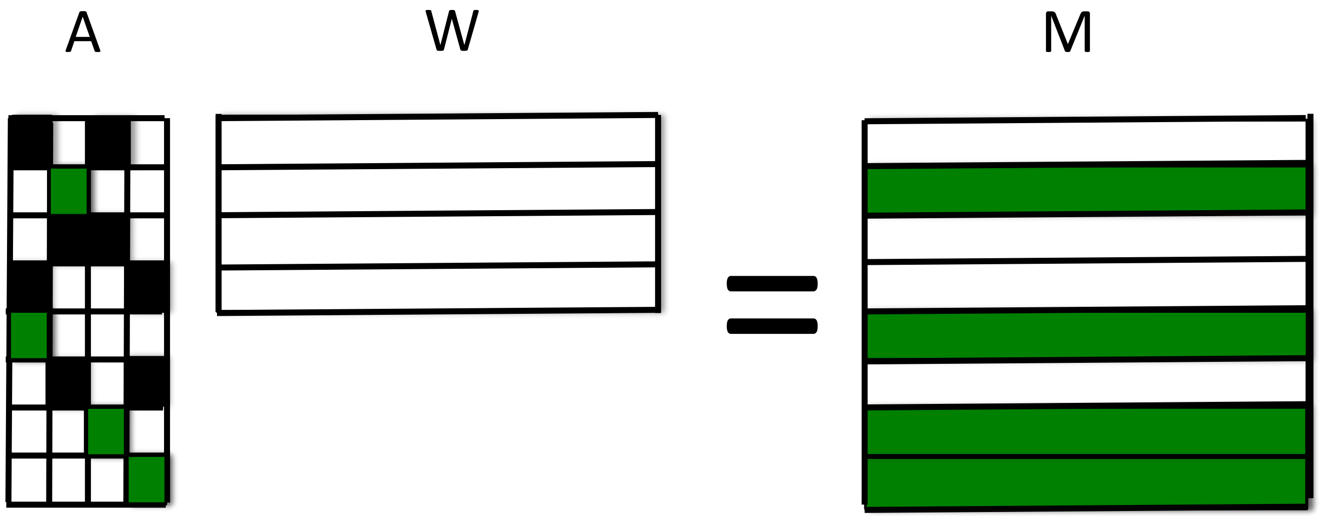
**Observation:** If **A** is separable, the rows of **W** appear as rows of **M**, we just need to find the anchor words!

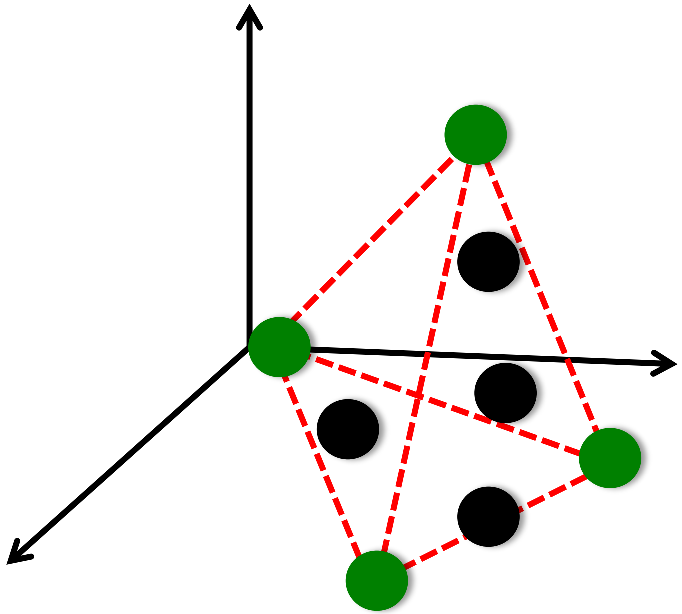
## How can we find the anchor words?

# ANCHOR WORDS ≅ VERTICES

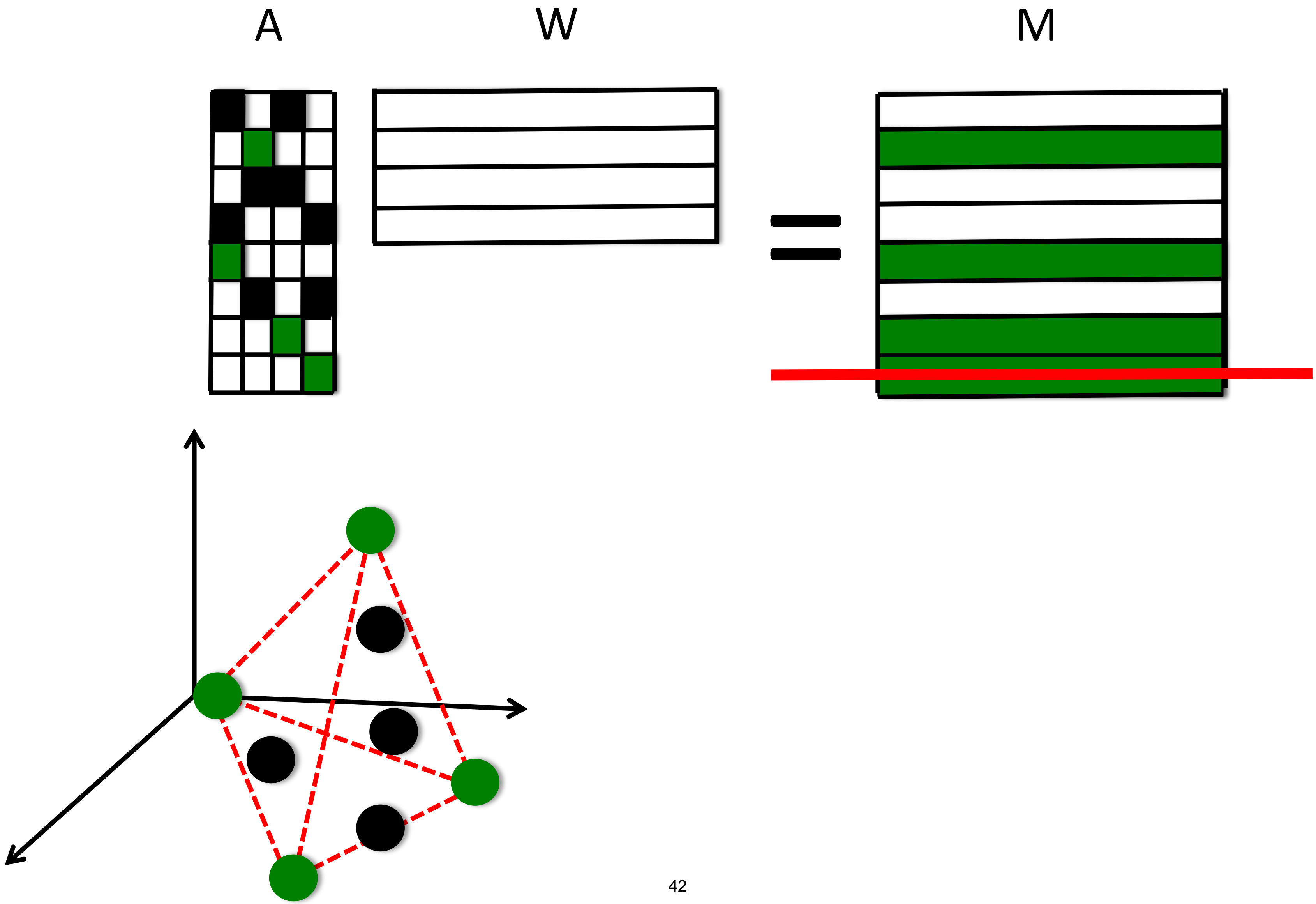


# ANCHOR WORDS ≅ VERTICES

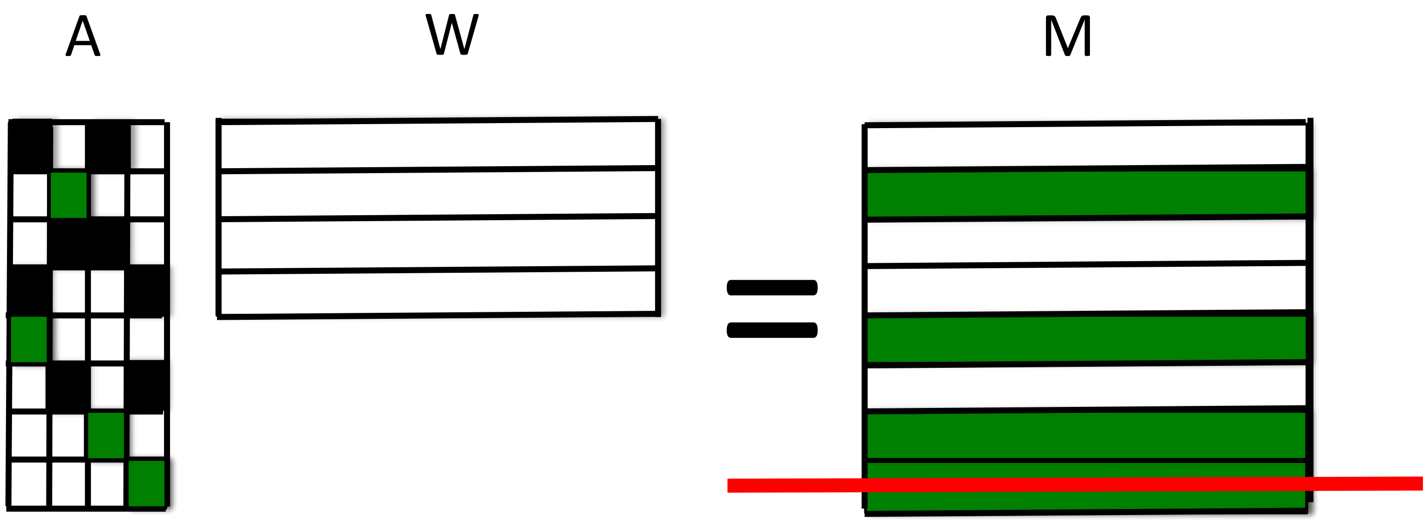


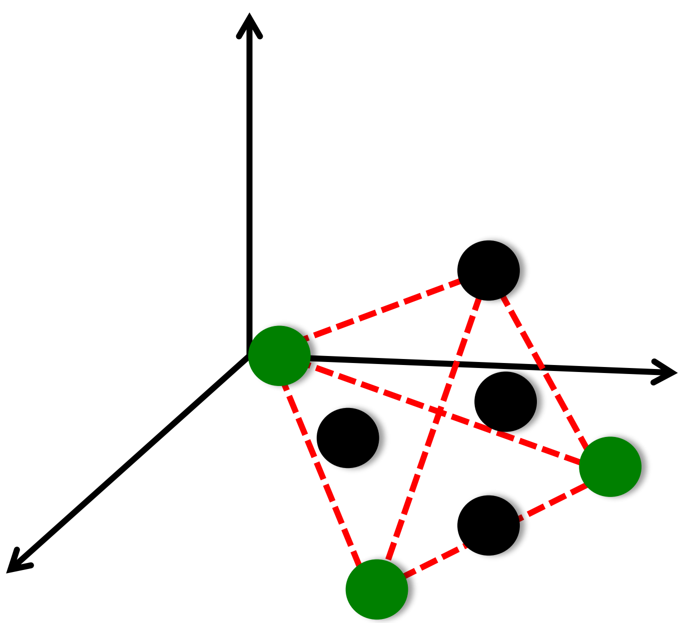



# ANCHOR WORDS ≅ VERTICES

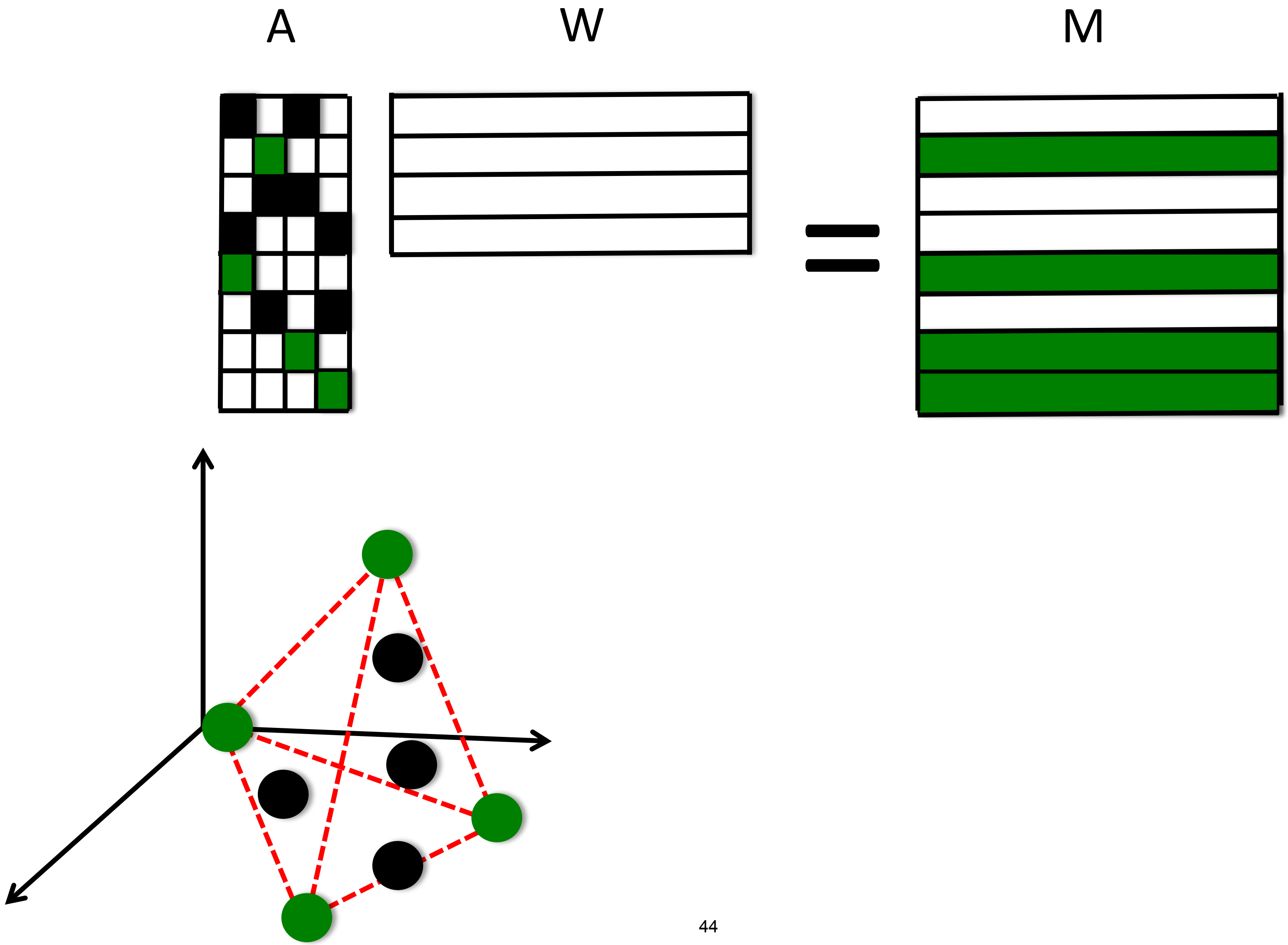


# ANCHOR WORDS ≅ VERTICES

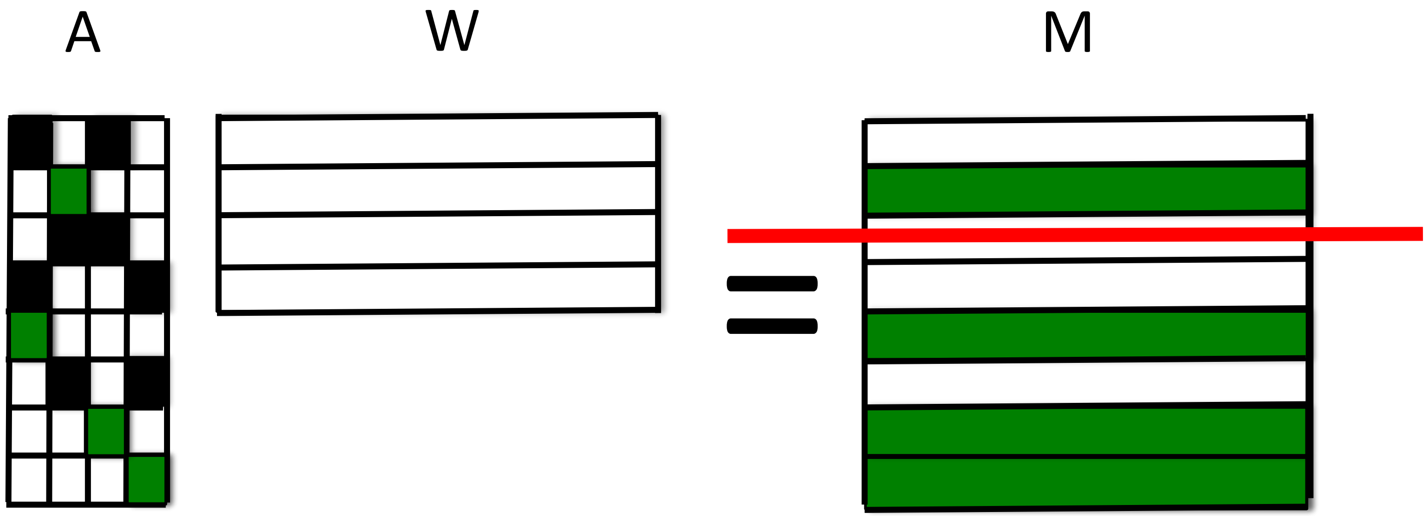


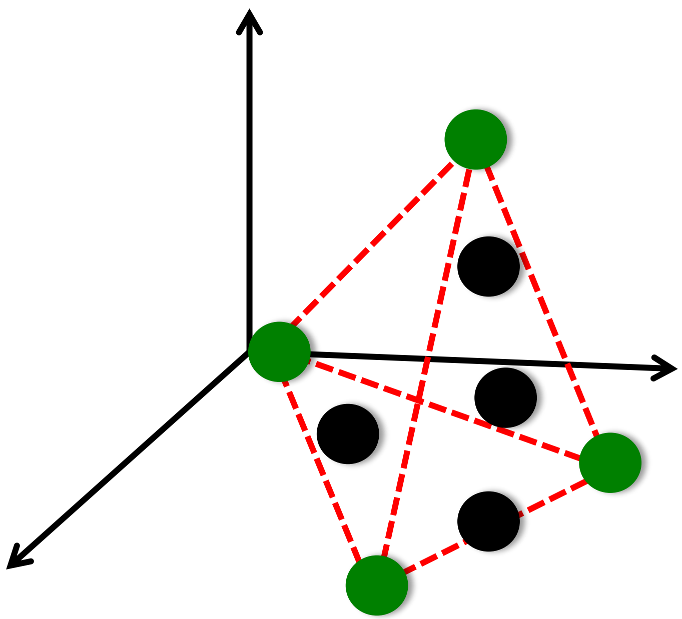



# ANCHOR WORDS ≅ VERTICES

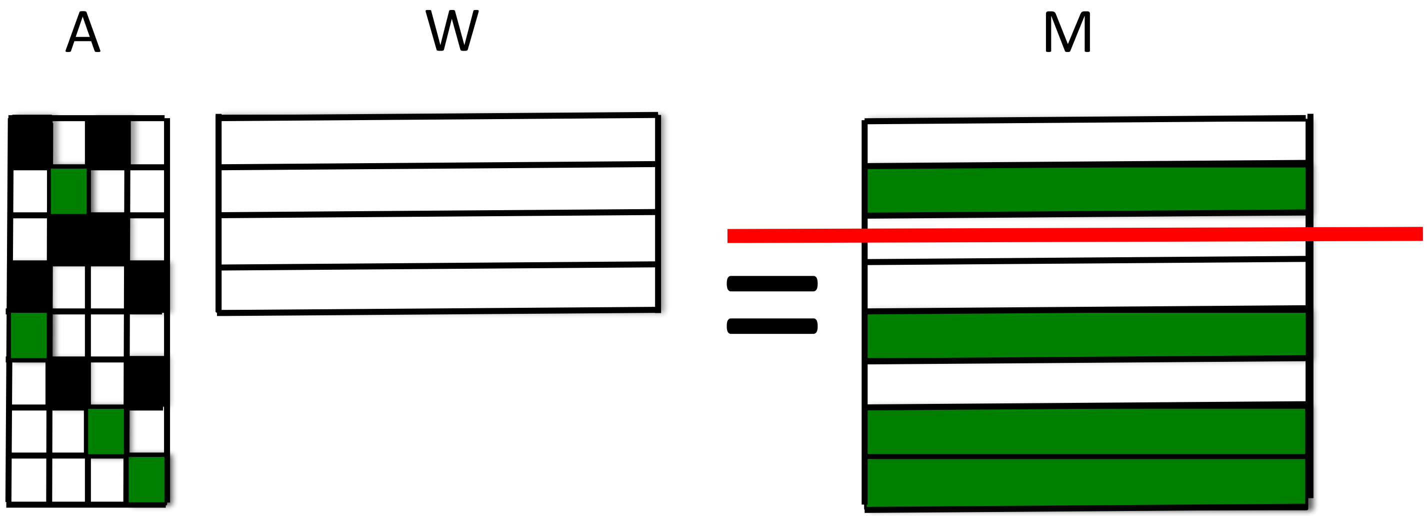


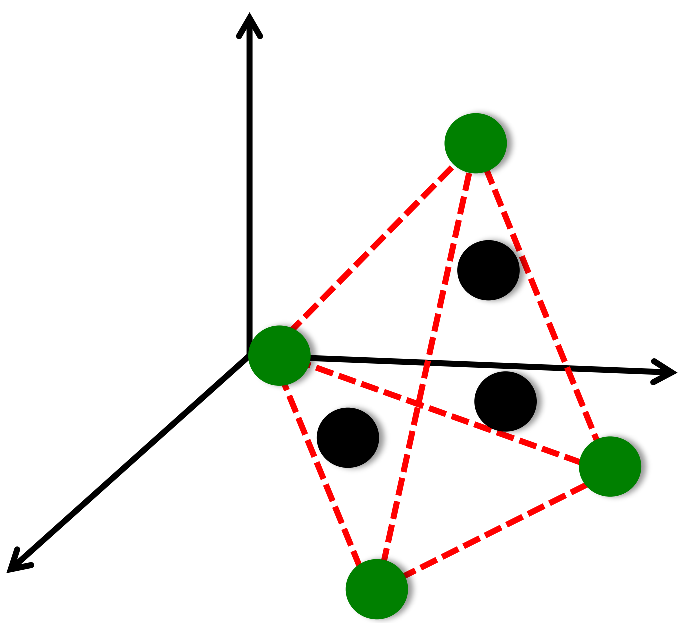
# ANCHOR WORDS $\cong$ VERTICES



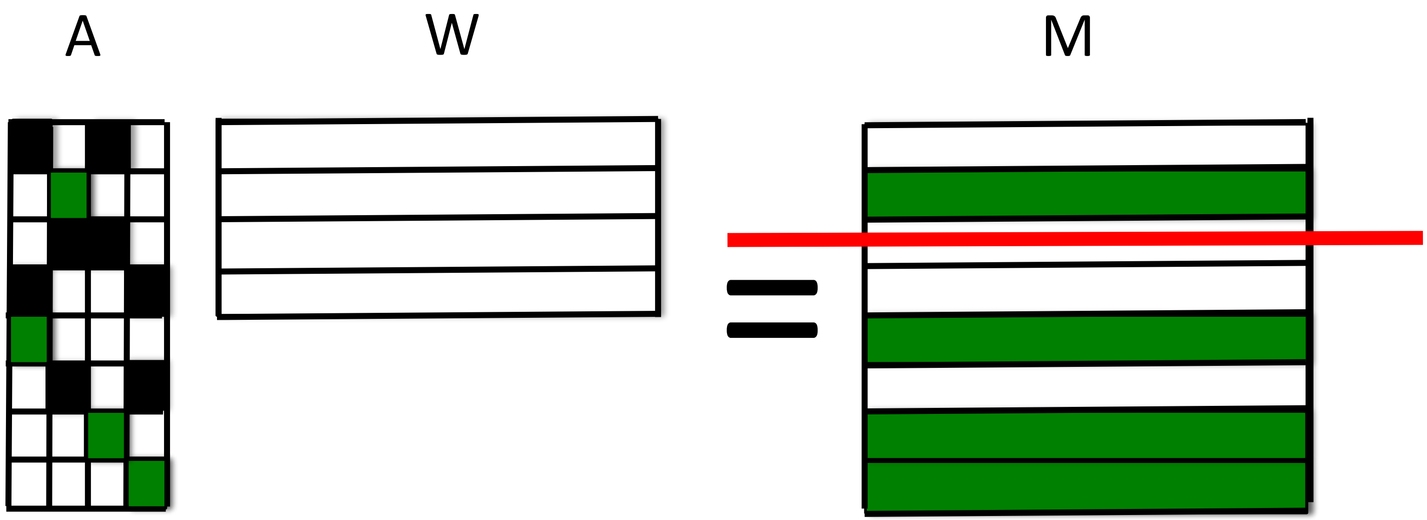



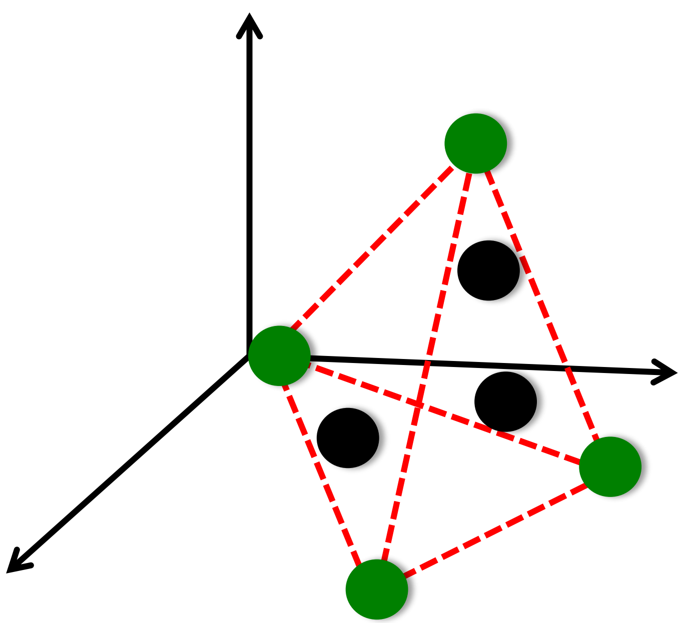
# ANCHOR WORDS ≅ VERTICES



# ANCHOR WORDS ≅ VERTICES



Deleting a word changes the convex hull

⇕

it is an anchor word

## How do anchor words help?

**Observation:** If **A** is separable, the rows of **W** appear as rows of **M**, we just need to find the anchor words!

## How can we find the anchor words?

Anchor words are extreme points; can be found by linear programming (or a combinatorial distance-based algorithm)

**The NMF Algorithm:**

- find the anchor words (linear programming)
- paste these vectors in as rows in **W**
- find the nonnegative **A** so that AW ≈ M (convex programming)

# OUTLINE

Are there efficient algorithms to find the topics?

**Challenge:** We cannot **rigorously** analyze algorithms used in practice! (When do they work? run quickly?)

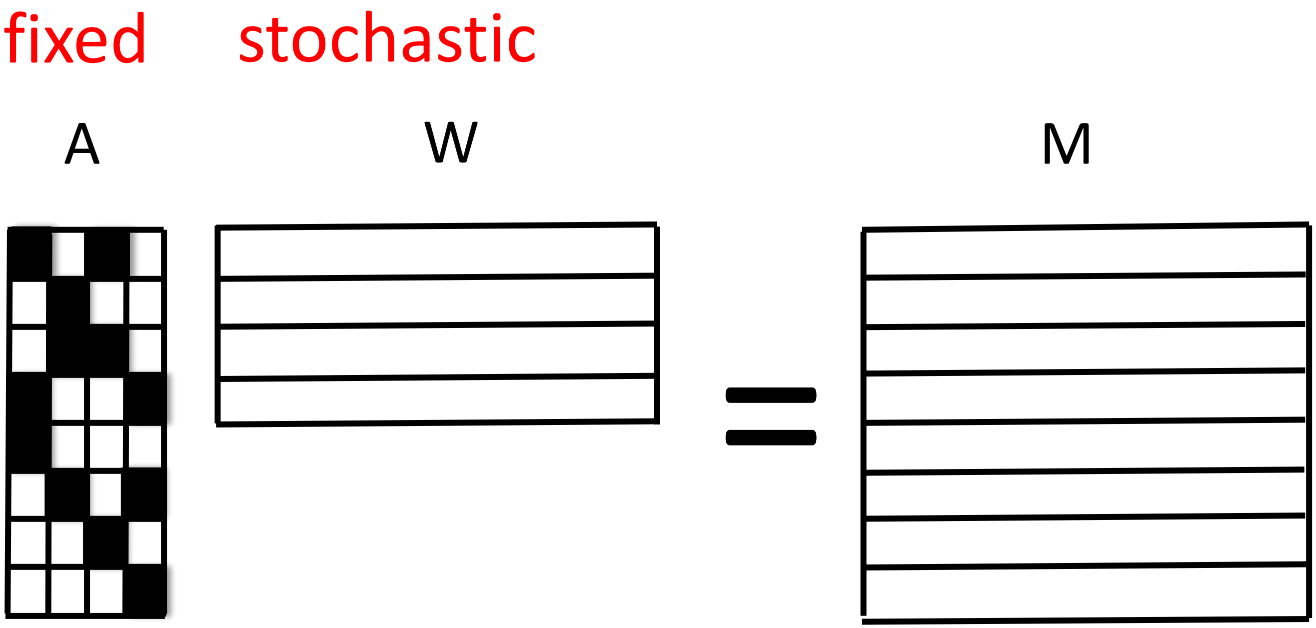
**Part I: An Optimization Perspective**

- Nonnegative Matrix Factorization
- Separability and Anchor Words
- Algorithms for Separable Instances

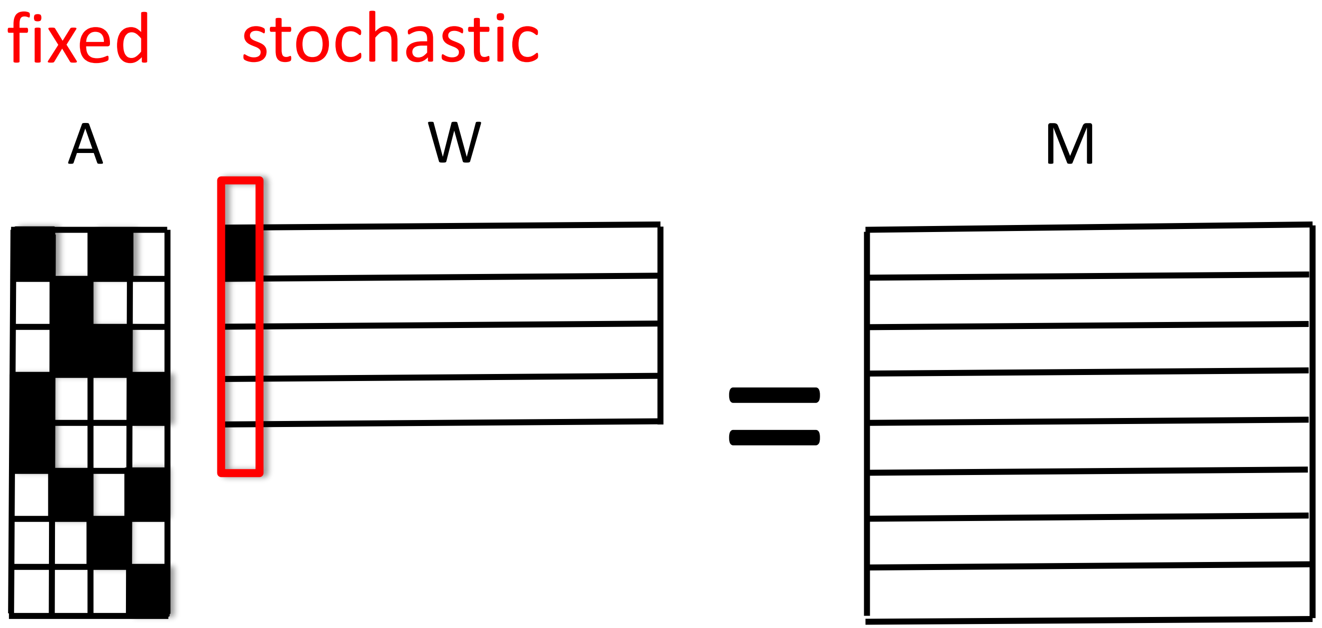
**Part II: A Bayesian Perspective**

- Topic Models (e.g. LDA, CTM, PAM, …)
- Algorithms for Inferring the Topics
- Experimental Results

# TOPIC MODELS

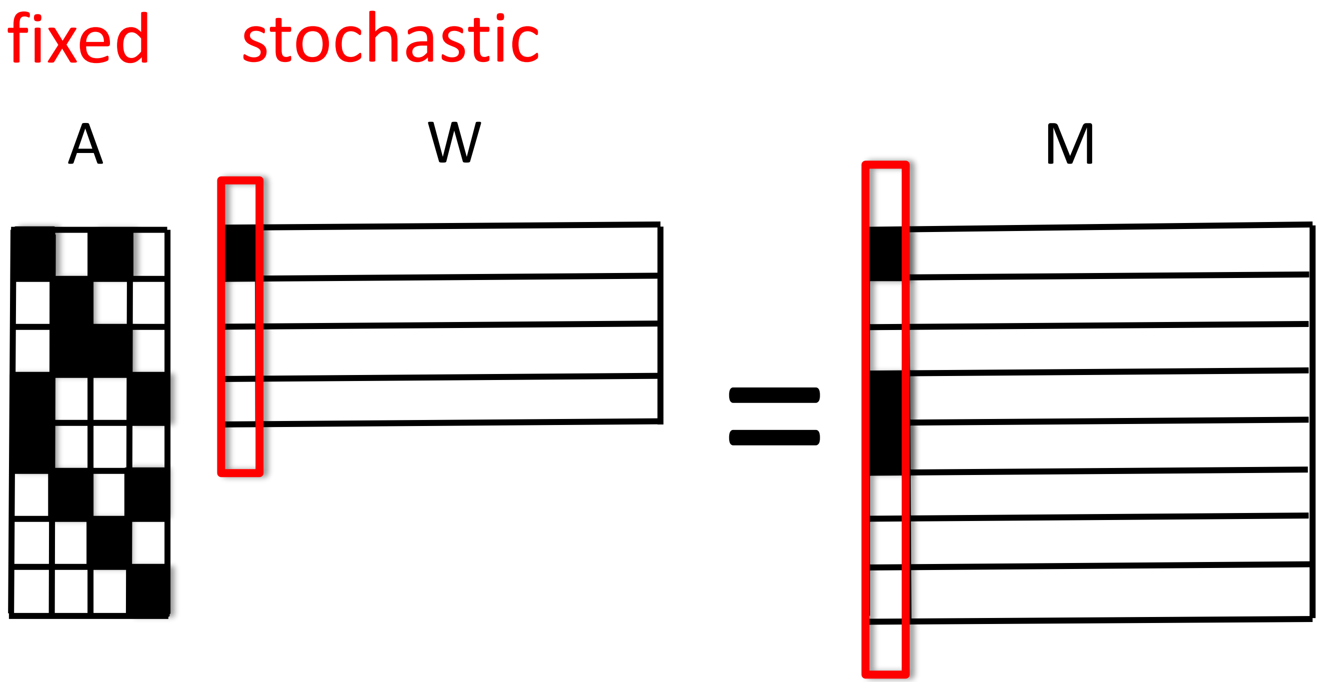


# TOPIC MODELS



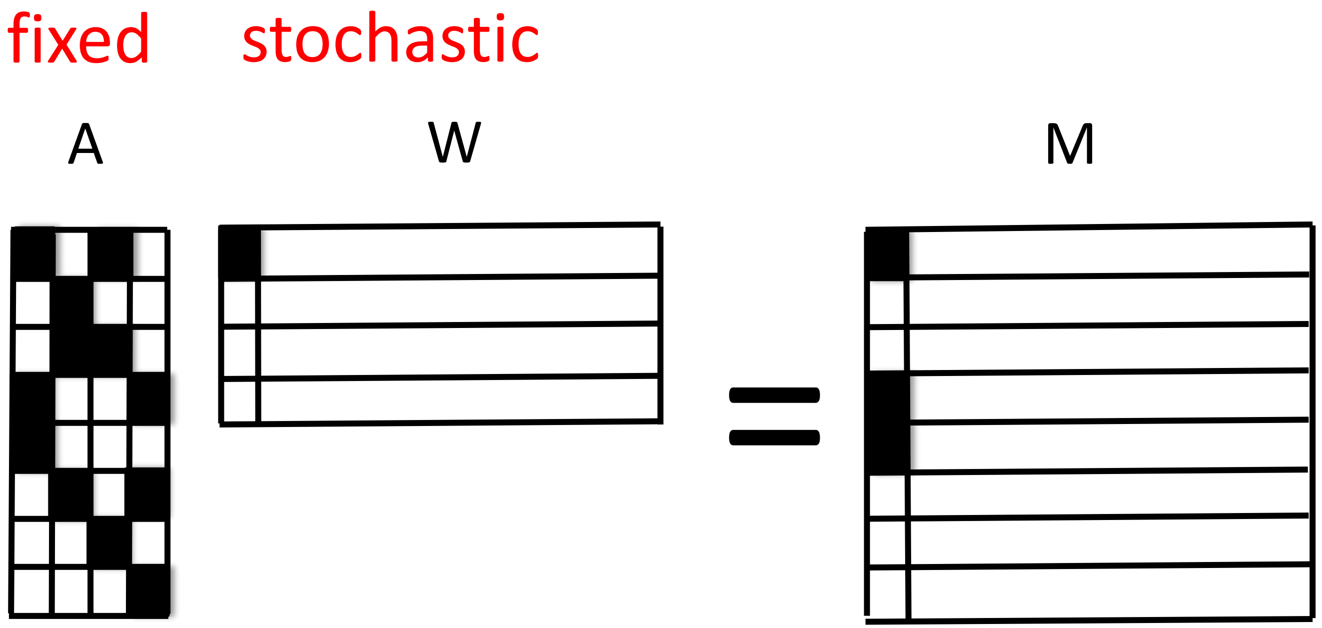

document #1: (1.0, personal finance)

# TOPIC MODELS

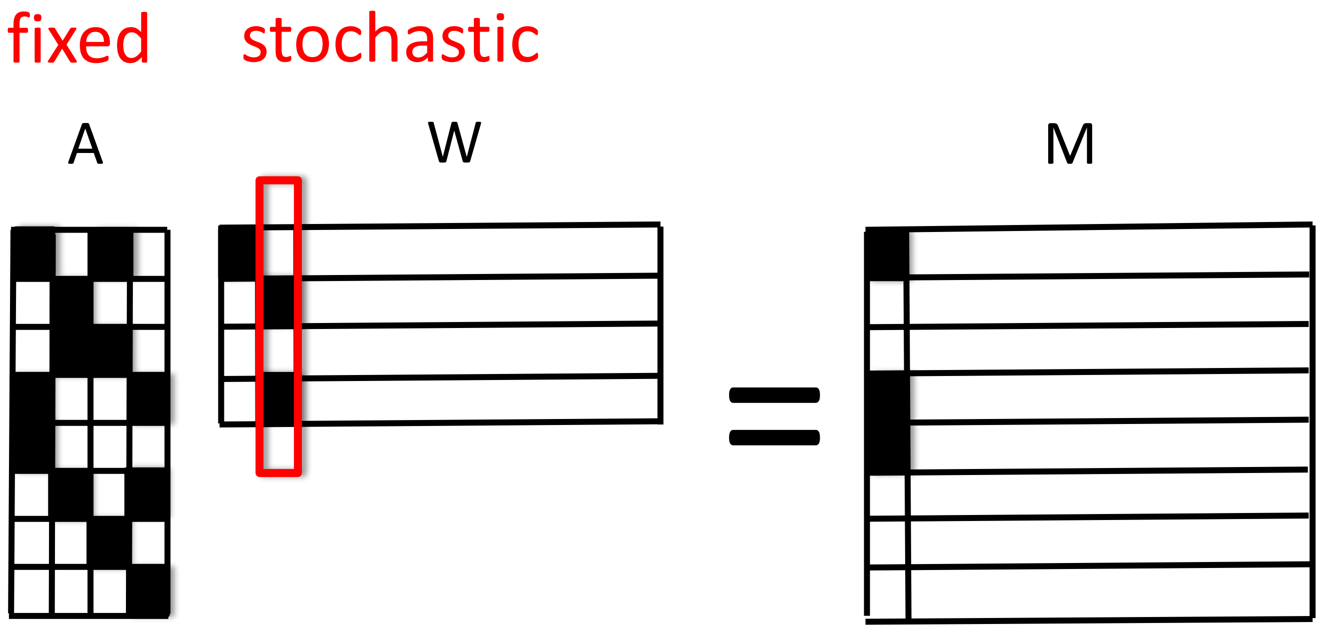



document #1: (1.0, personal finance)

# TOPIC MODELS

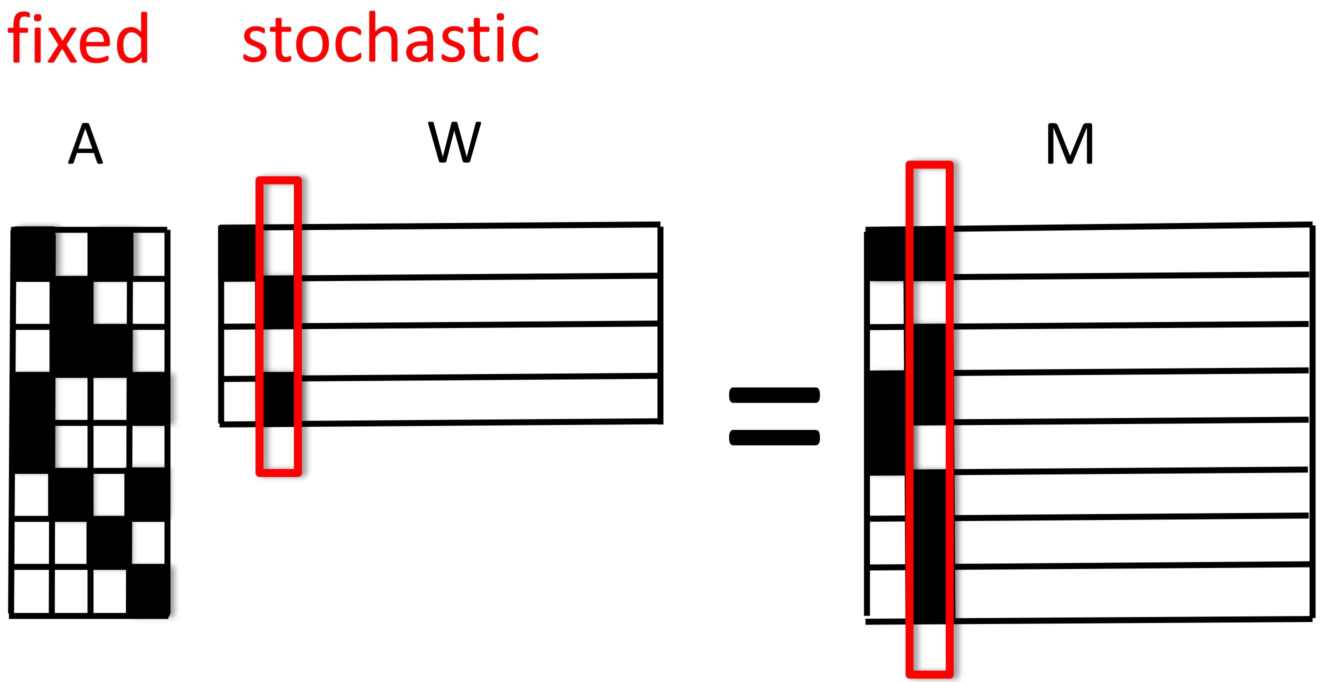


# TOPIC MODELS



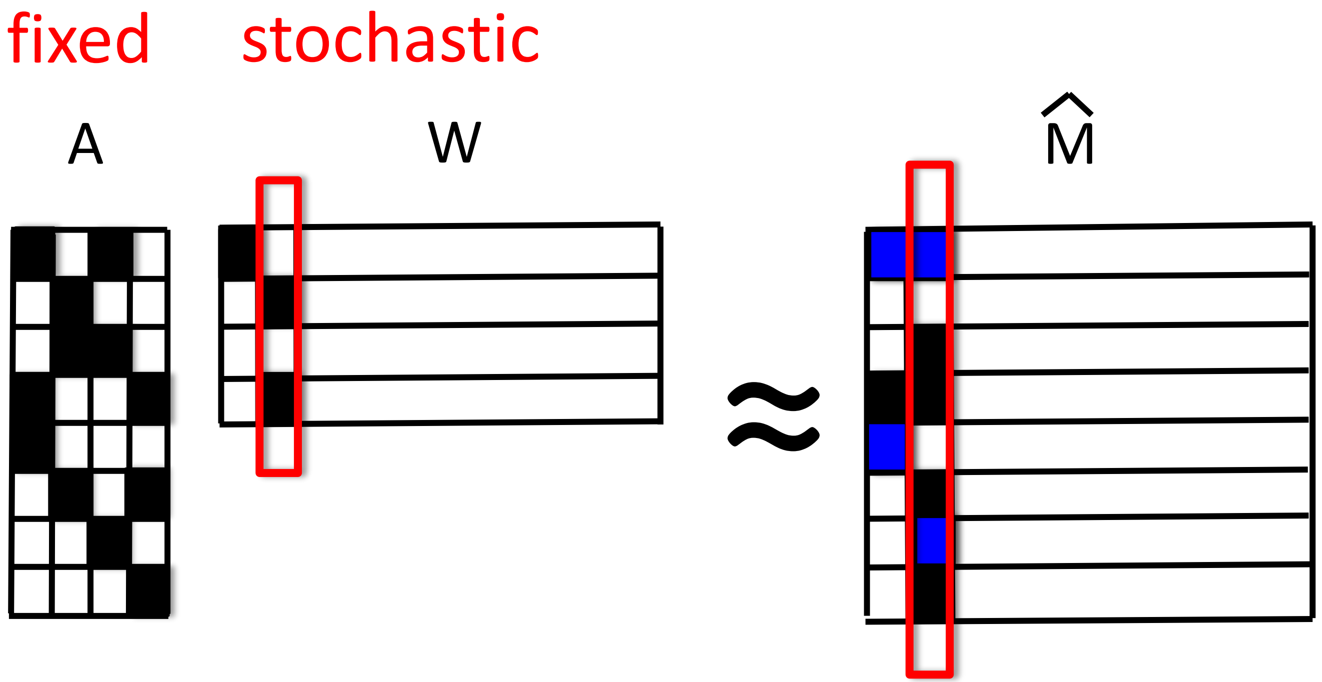

document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS




document #2: (0.5, baseball); (0.5, movie review)

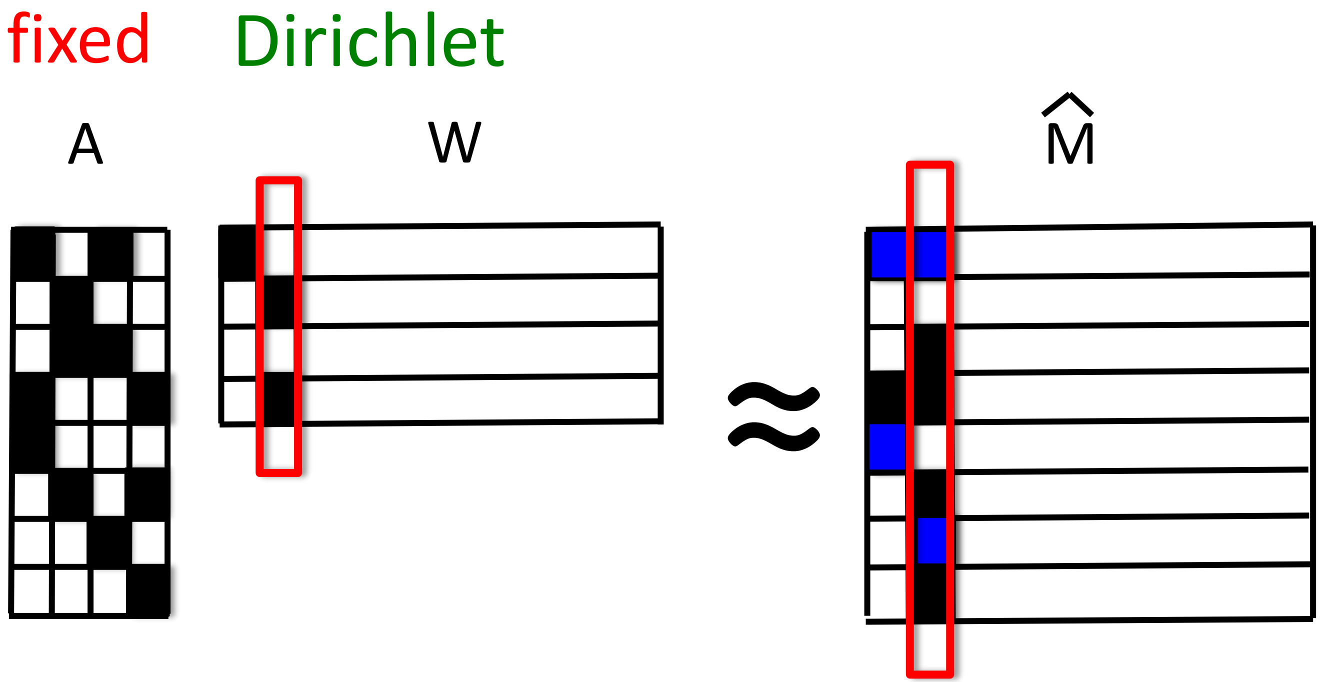
# TOPIC MODELS




document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS

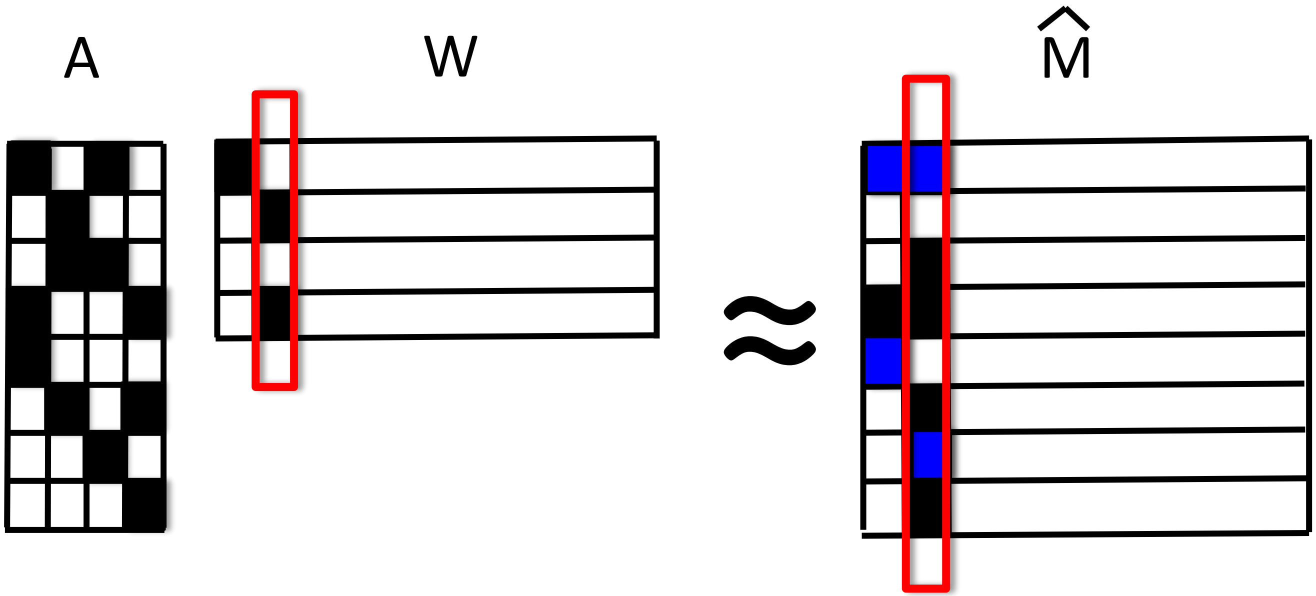
Latent Dirichlet Allocation (Blei, Ng, Jordan)




document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS

fixed

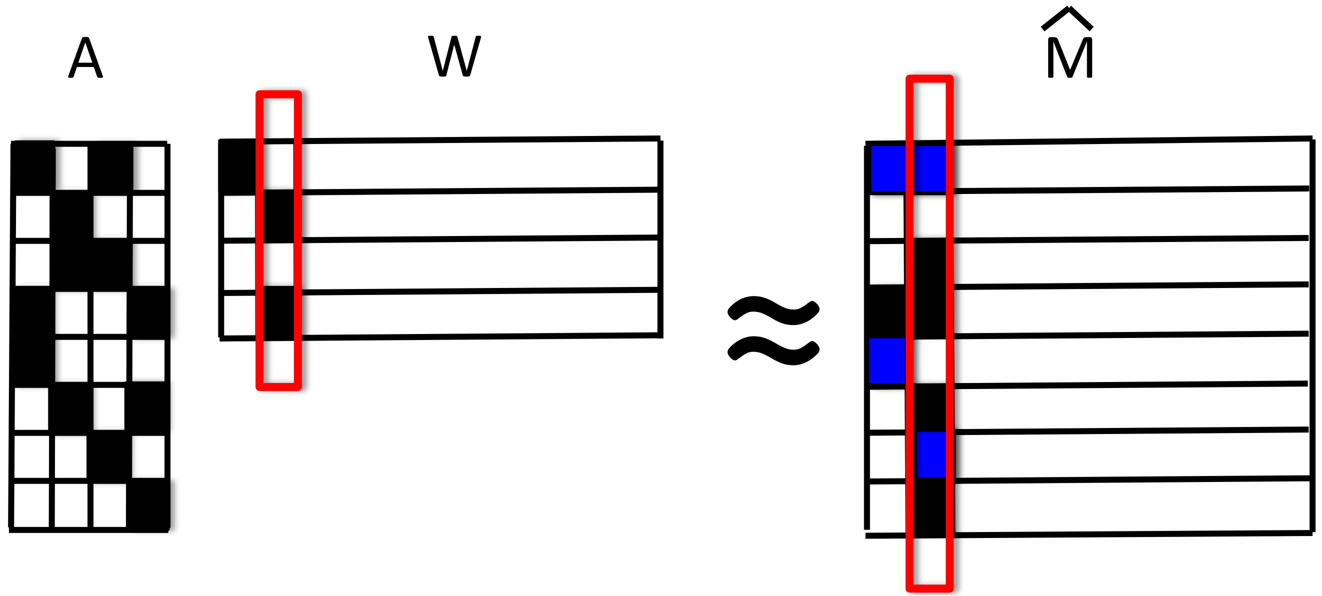



document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS

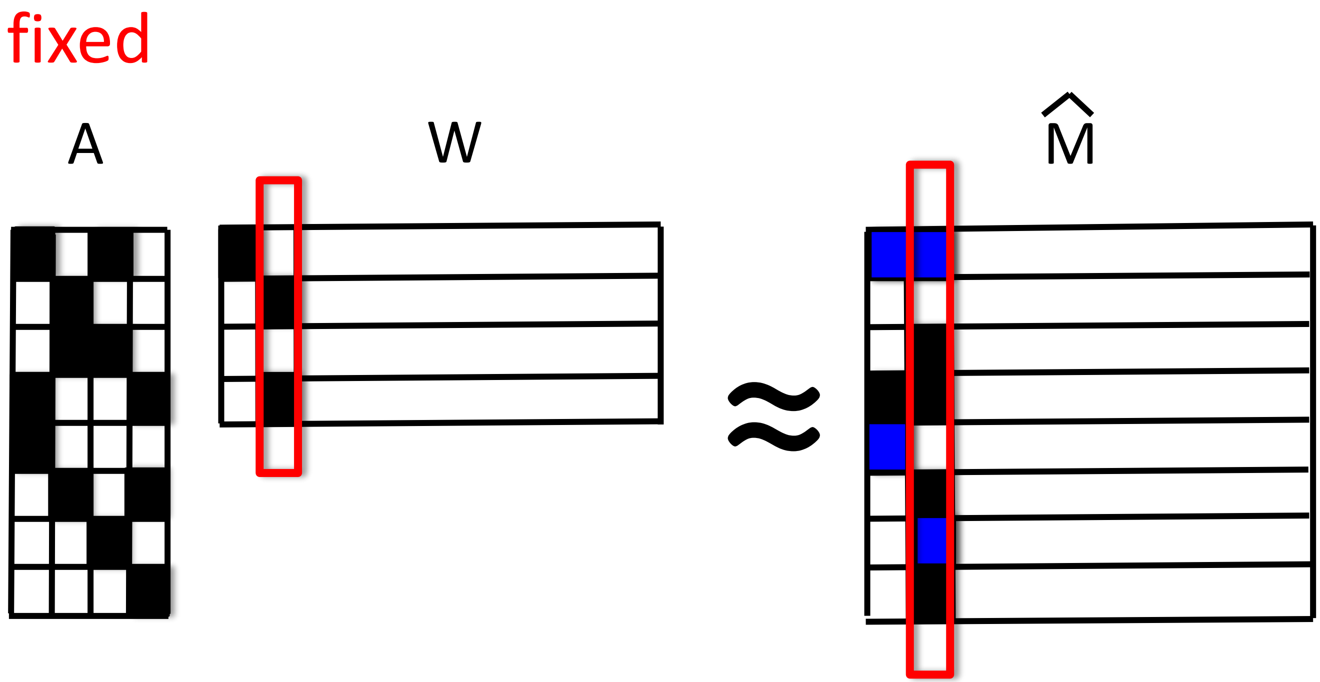
Correlated Topic Model (Blei, Lafferty)

fixed Logisitic Normal




document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS

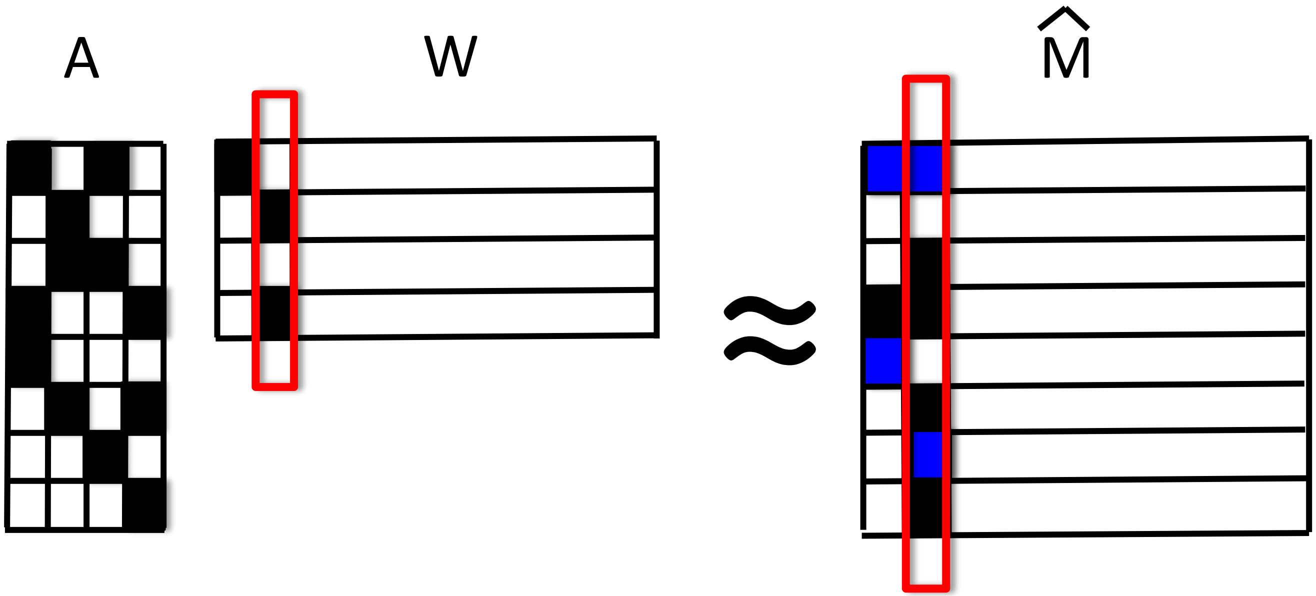



document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS

Pachinko Allocation Model (Li, McCallum)

fixed Multilevel DAG

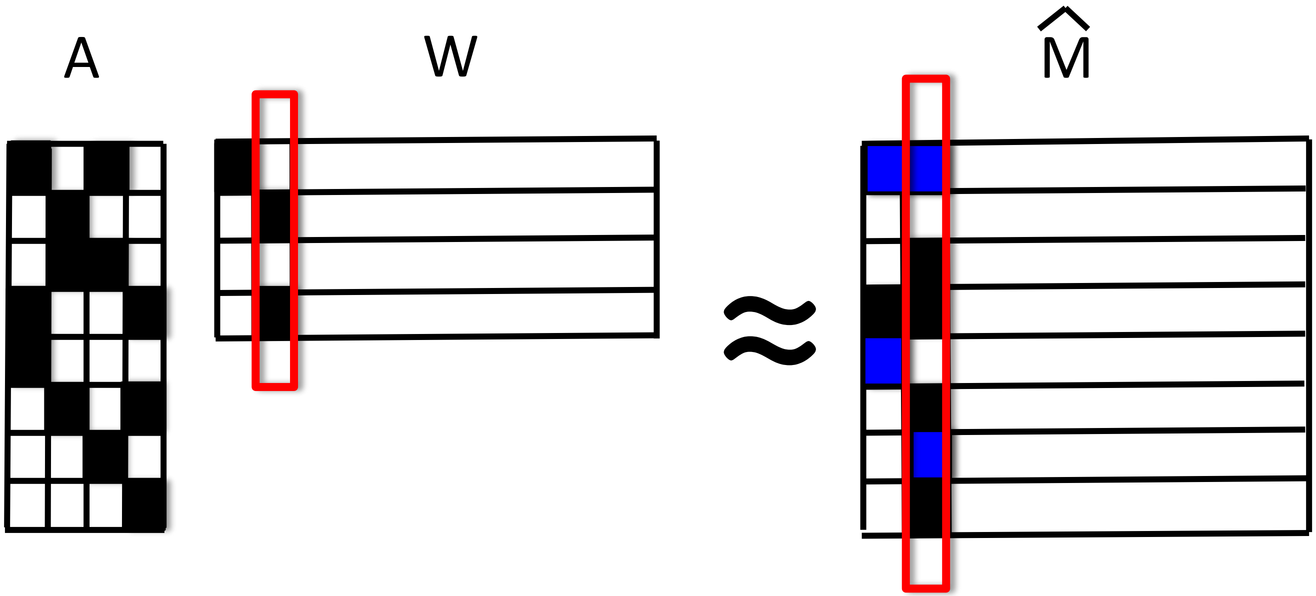



document #2: (0.5, baseball); (0.5, movie review)

# TOPIC MODELS

Pachinko Allocation Model (Li, McCallum)

fixed Multilevel DAG




document #2: (0.5, baseball); (0.5, movie review)

These models differ only in how **W** is generated

# ALGORITHMS FOR TOPIC MODELS?

What if documents are **short**; can we still find **A**?

The crucial observation is, we can work with the **Gram matrix** (defined next…)

# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

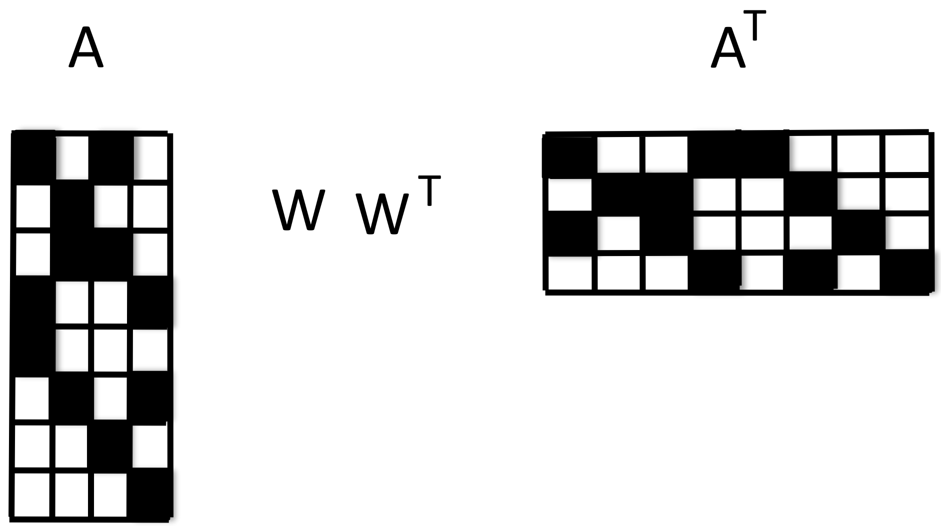
**Gram Matrix**

$$\hat{M}\hat{M}^T$$

# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

$\hat{M}\hat{M}^T$

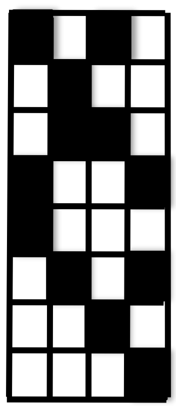


# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

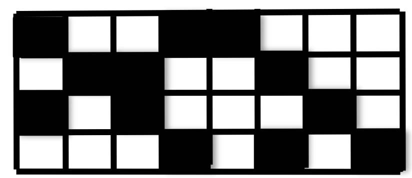
$\hat{M}\hat{M}^T \rightarrow E[M M^T]$

$A$



$W W^T$

$A^T$

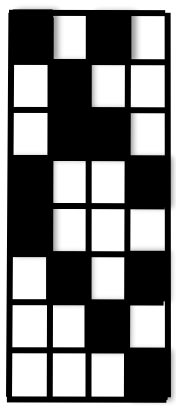


# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

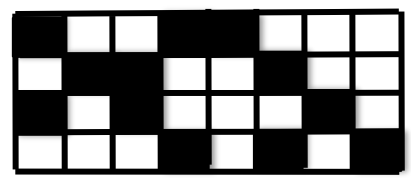
$$\hat{M}\hat{M}^T \rightarrow E[MM^T] = A\,E[WW^T]\,A^T$$

A



$W \; W^T$

$A^T$

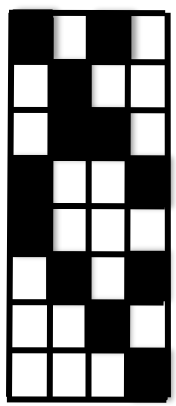


# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

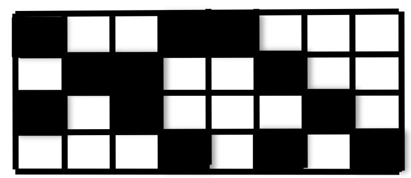
$$\hat{M}\hat{M}^T \rightarrow E[MM^T] = A\,E[WW^T]\,A^T \rightarrow ARA^T$$

A



$$\overbrace{W\ \ W^T}^{R}$$

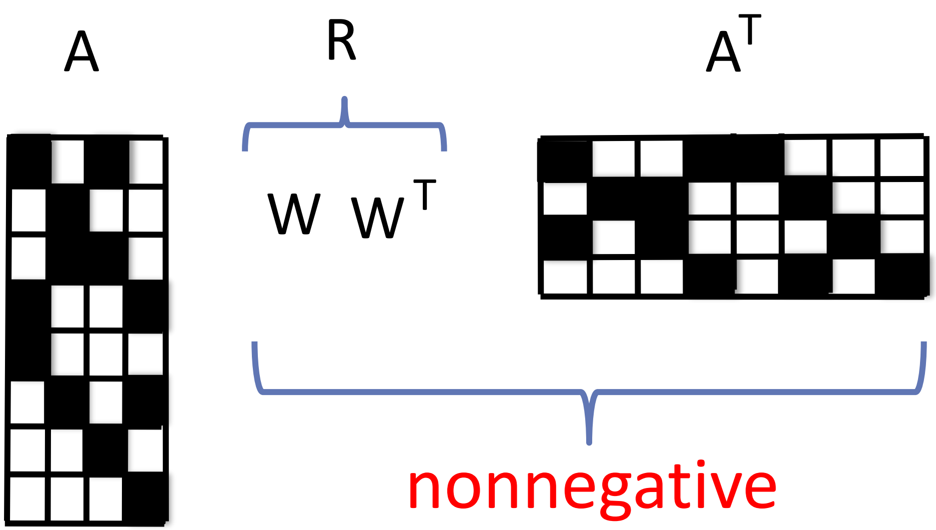
$A^T$



# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

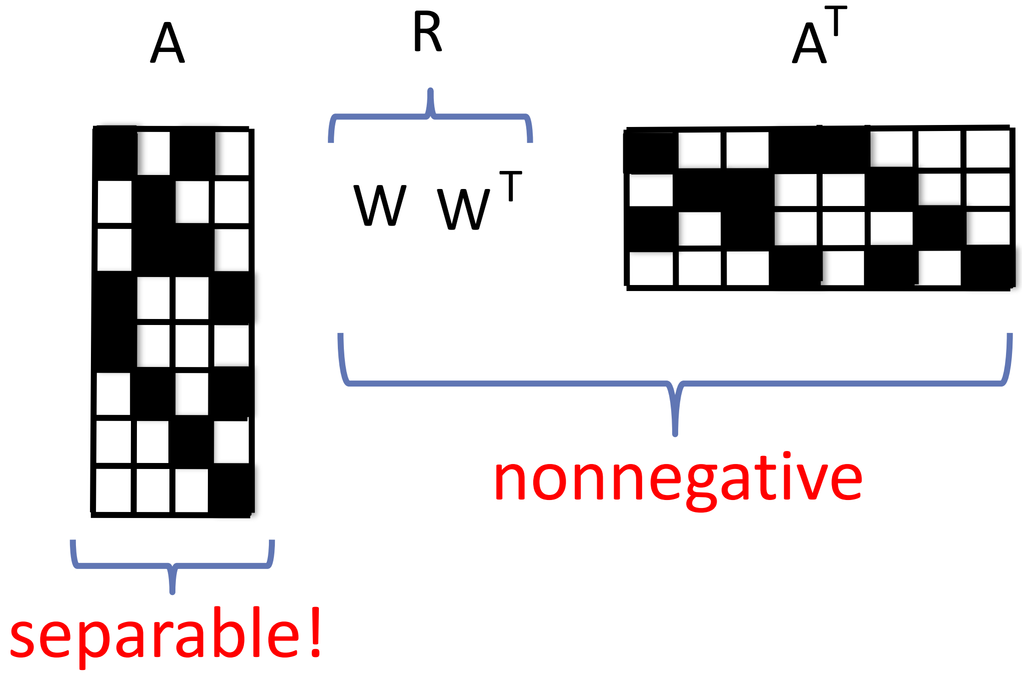
$$\hat{M}\hat{M}^T \rightarrow E[M M^T] = A\,E[W W^T]\,A^T \rightarrow ARA^T$$



# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

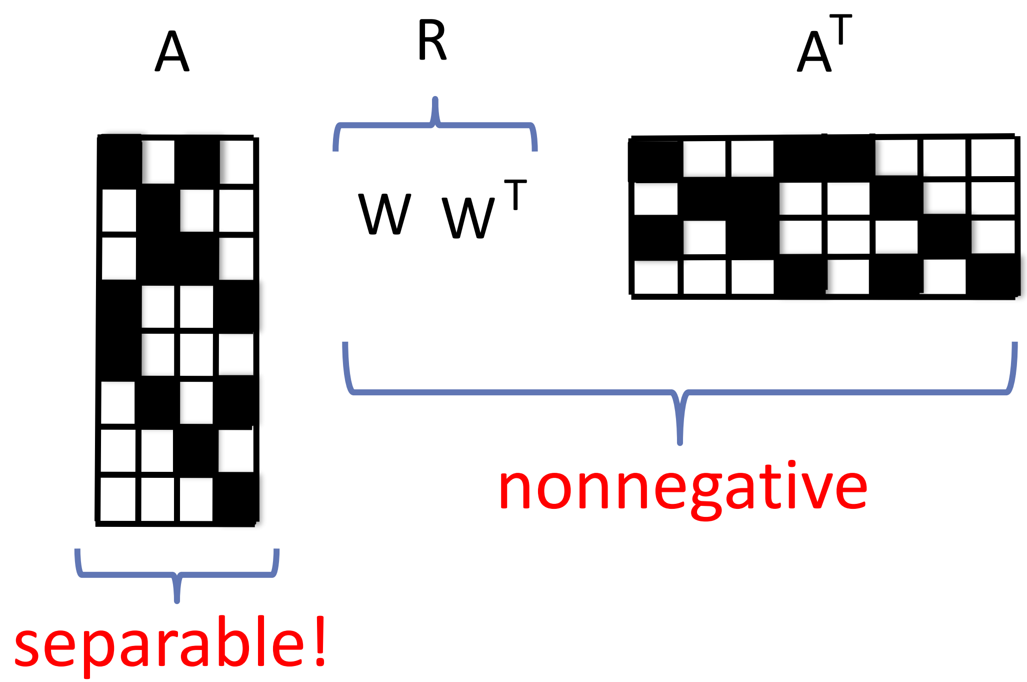
$$\hat{M}\hat{M}^T \longrightarrow E[MM^T] = A\,E[WW^T]\,A^T \longrightarrow ARA^T$$



# GRAM MATRIX (WHY? BECAUSE IT CONVERGES)

**Gram Matrix**

$$\hat{M}\hat{M}^T \rightarrow E[MM^T] = A\,E[WW^T]\,A^T \rightarrow ARA^T$$




Anchor words are extreme rows of the Gram matrix!

# ALGORITHMS FOR TOPIC MODELS?

What if documents are **short**; can we still find **A**?

The crucial observation is, we can work with the **Gram matrix** (defined next…)

Given enough documents, we can still find the anchor words!

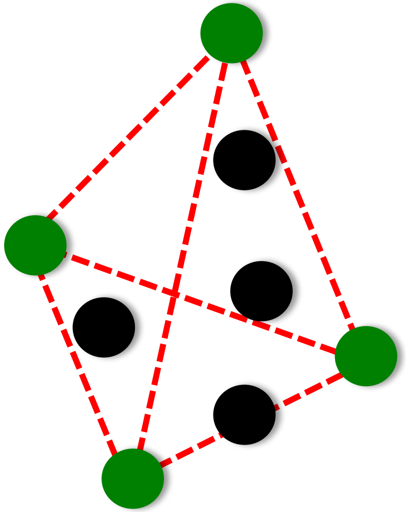
How can we use the anchor words to find the rest of **A**?

The **posterior distribution** Pr[topic|word] is supported on just one topic, for an anchor word

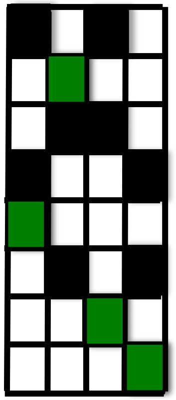
We can use the anchor words to find Pr[topic|word] for all the other words…

# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\widehat{M}\widehat{M}^T$

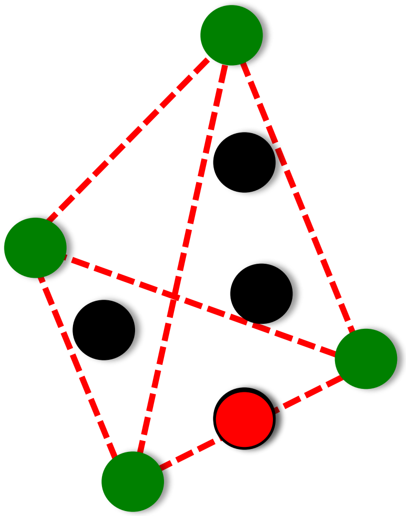


A

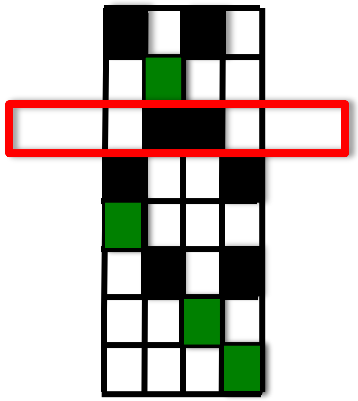


# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\widehat{M}\widehat{M}^T$

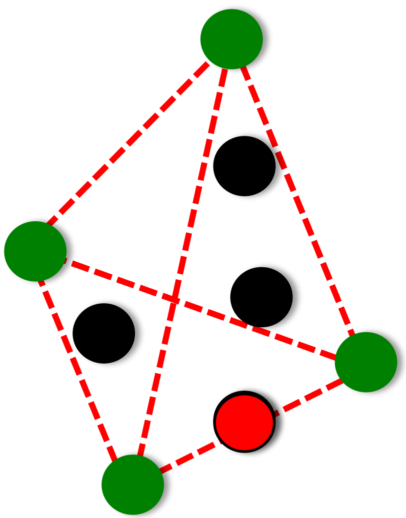


A

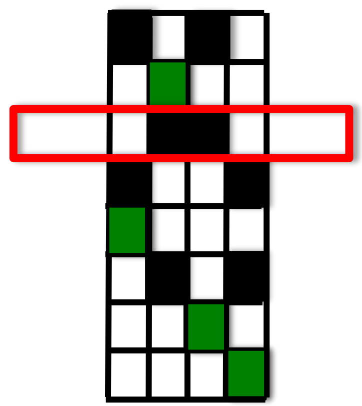


# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\hat{M}\hat{M}^T$



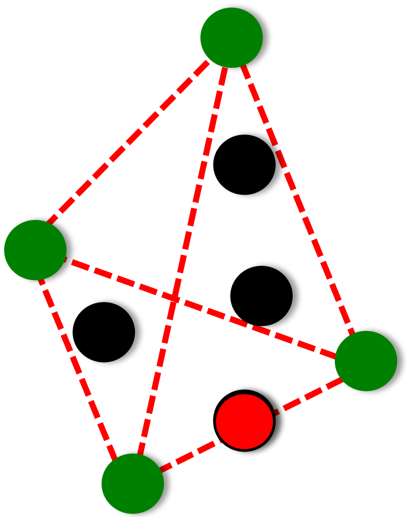
A



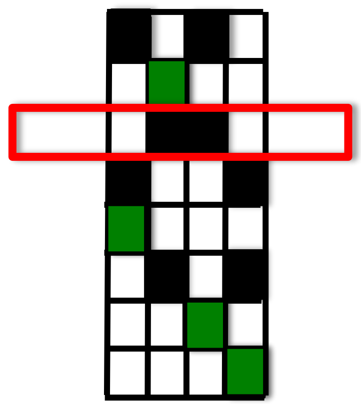
word #3: (0.5, anchor #2); (0.5, anchor #3)

# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\hat{M}\hat{M}^T$



A



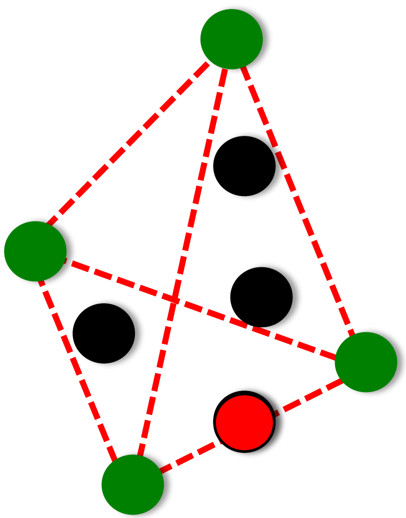
word #3: (0.5, anchor #2); (0.5, anchor #3)

$\Updownarrow$

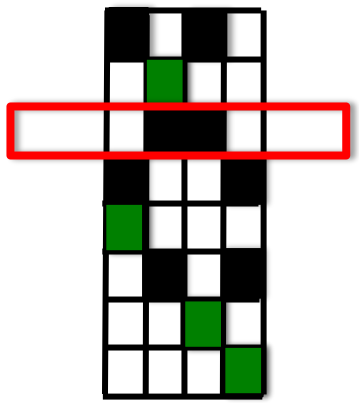
Pr[topic|word #3]: (0.5, topic #2); (0.5, topic #3)

# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\hat{M}\hat{M}^T$



A



what we have:

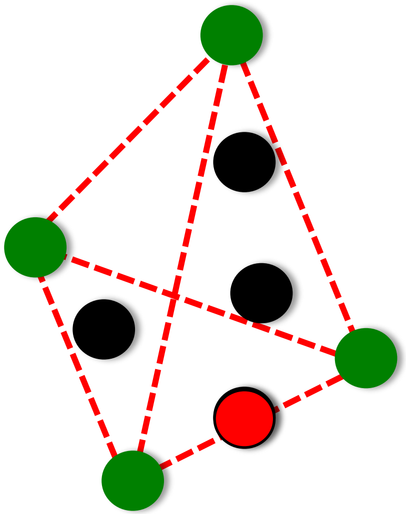
**Pr[topic|word]**

word #3: (0.5, anchor #2); (0.5, anchor #3)

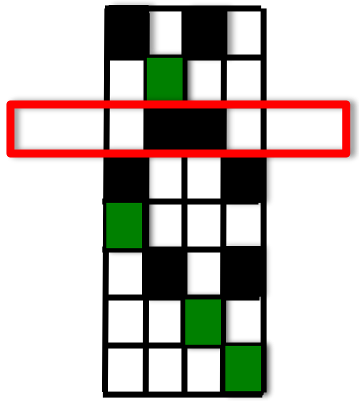
Pr[topic|word #3]: (0.5, topic #2); (0.5, topic #3)

# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\hat{M}\hat{M}^T$



A



what we have:

**Pr[topic|word]**

what we want:

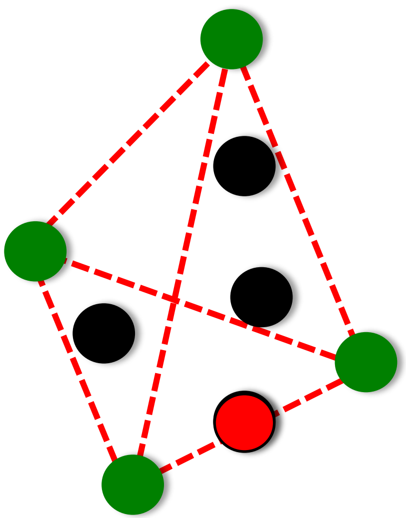
**Pr[word|topic]**

word #3: (0.5, anchor #2); (0.5, anchor #3)

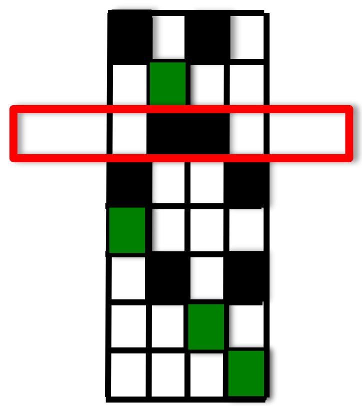
Pr[topic|word #3]: (0.5, topic #2); (0.5, topic #3)

# BAYES RULE (OR HOW TO USE ANCHOR WORDS)

points are now (normalized) rows of $\hat{M}\hat{M}^{T}$



A



what we have:

**Pr[topic|word]**

Bayes Rule

what we want:

**Pr[word|topic]**

word #3: (0.5, anchor #2); (0.5, anchor #3)

Pr[topic|word #3]: (0.5, topic #2); (0.5, topic #3)

Compute **A** using Bayes Rule:

$$\Pr[\text{word}|\text{topic}] = \frac{\Pr[\text{topic}|\text{word}]\ \Pr[\text{word}]}{\sum_{\text{word}'} \Pr[\text{topic}|\text{word}']\ \Pr[\text{word}']}$$

Compute **A** using Bayes Rule:

$$\mathbf{Pr[word|topic]} = \frac{\mathbf{Pr[topic|word]\ Pr[word]}}{\sum_{\mathbf{word'}} \mathbf{Pr[topic|word']\ Pr[word']}}$$

**The Topic Model Algorithm:**

- form the Gram matrix and find the anchor words
- write each word as a convex combination of the anchor words to find **Pr[topic|word]**
- compute **A** from the formula above

This **provably** works for **any** topic model (LDA, CTM, PAM, etc …) provided **A** is separable and **R** is non-singular

The previous algorithm was **inspired by experiments!**

Our first attempt used matrix inversion, which is noisy and unstable and can produce small **negative** values

## METHODOLOGY:

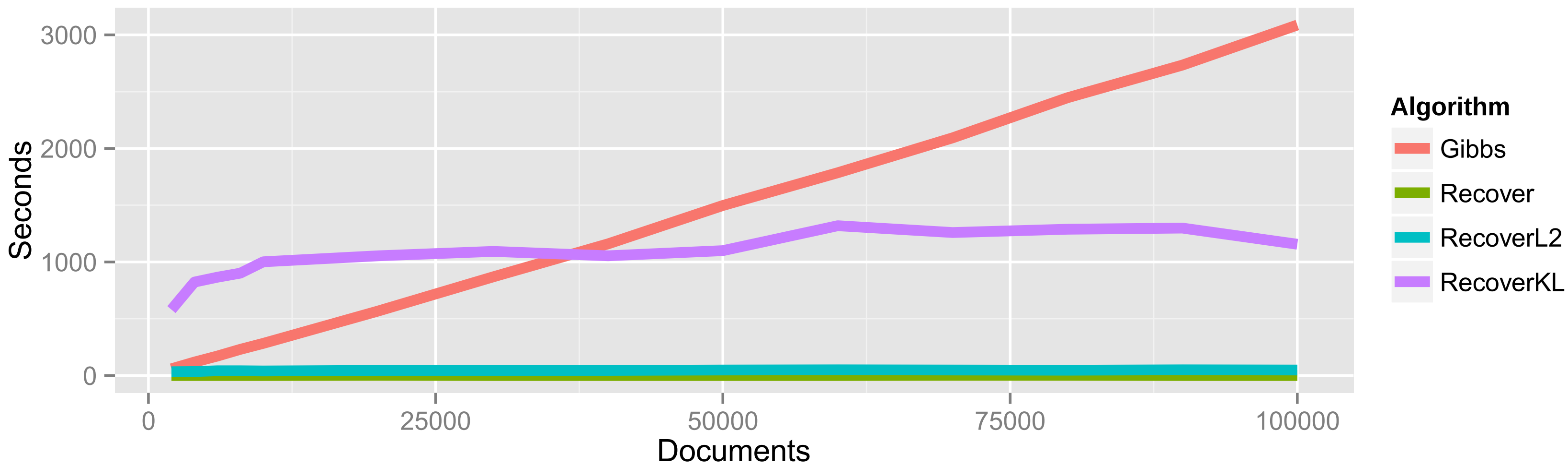
We ran our algorithm on real and synthetic data:

- synthetic data: train an LDA model on 1100 NIPS abstracts, use this model to run experiments

Our algorithm is **fifty times faster** and performs nearly the same on all metrics we tried (l_1, log-likelihood, coherence,…) when compared to MALLET

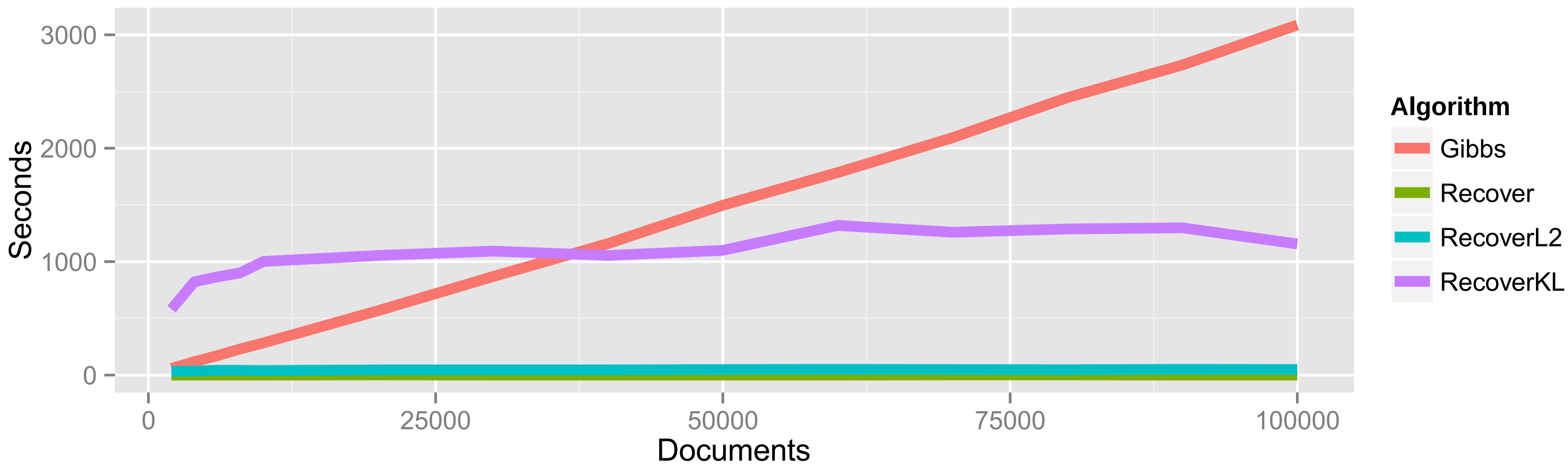
# EXPERIMENTAL RESULTS

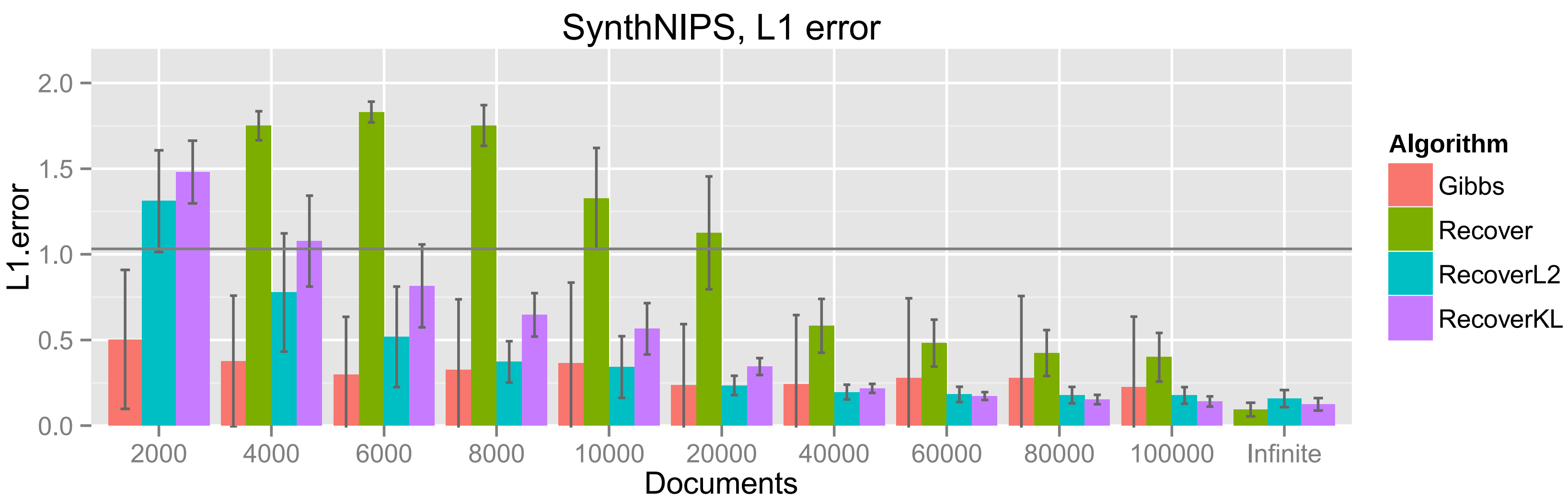
**[Arora, Ge, Halpern, Mimno, Moitra, Sontag, Wu, Zhu, ICML'13]:**



# EXPERIMENTAL RESULTS

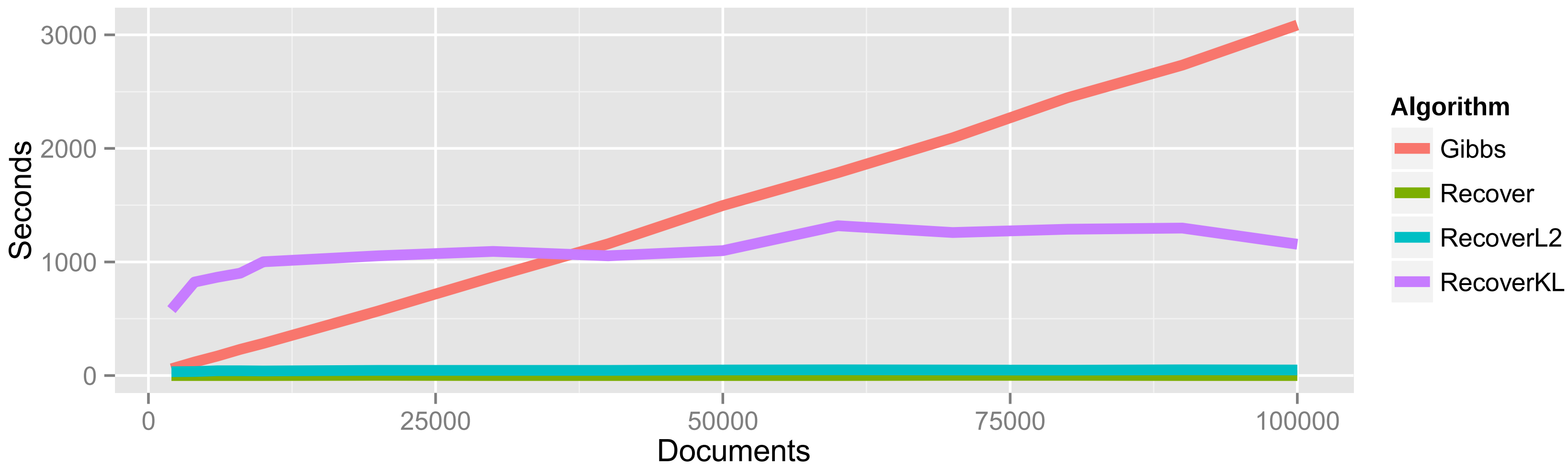
**[Arora, Ge, Halpern, Mimno, Moitra, Sontag, Wu, Zhu, ICML'13]:**



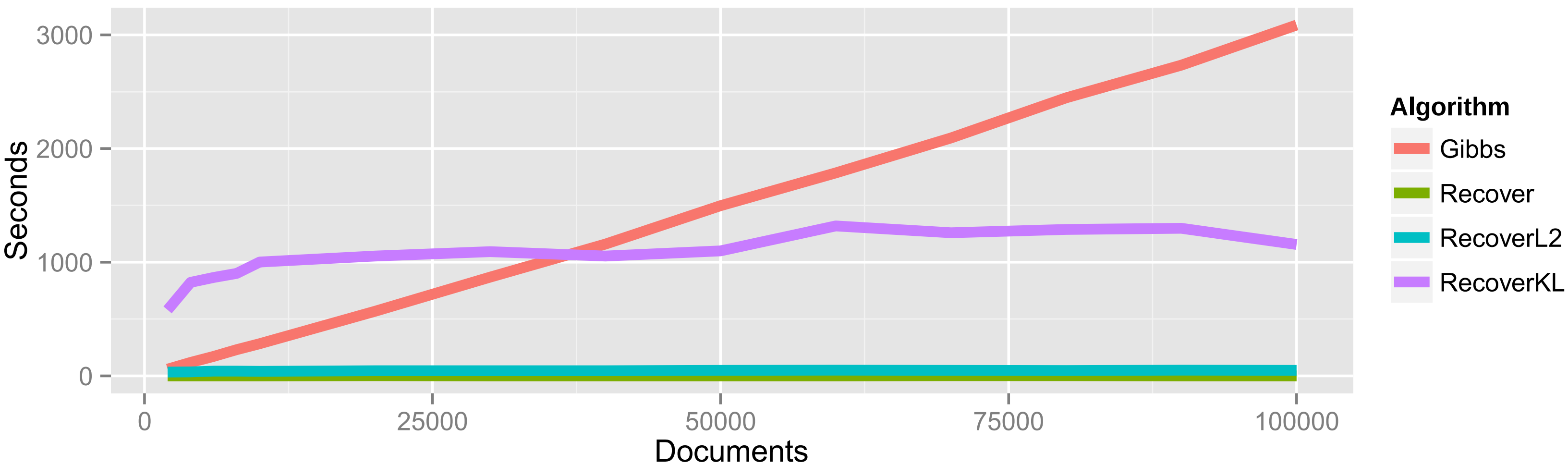
# EXPERIMENTAL RESULTS

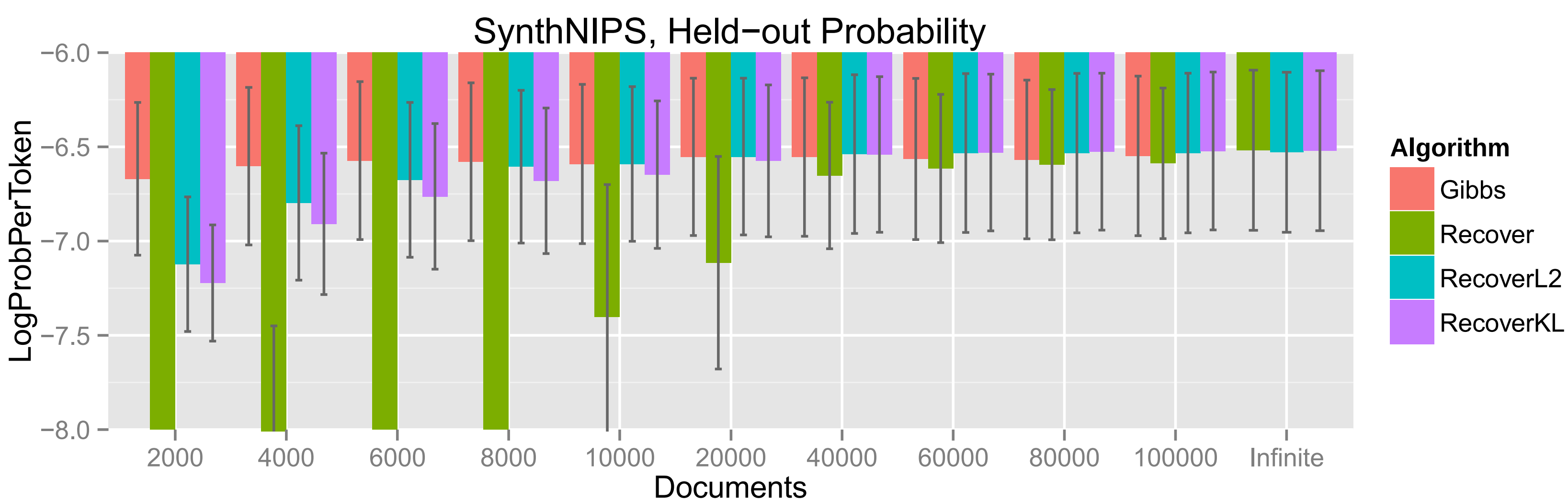
**[Arora, Ge, Halpern, Mimno, Moitra, Sontag, Wu, Zhu, ICML'13]:**



# EXPERIMENTAL RESULTS

## [Arora, Ge, Halpern, Mimno, Moitra, Sontag, Wu, Zhu, ICML'13]:



The previous algorithm was **inspired by experiments!**

Our first attempt used matrix inversion, which is noisy and unstable and can produce small **negative** values

## METHODOLOGY:

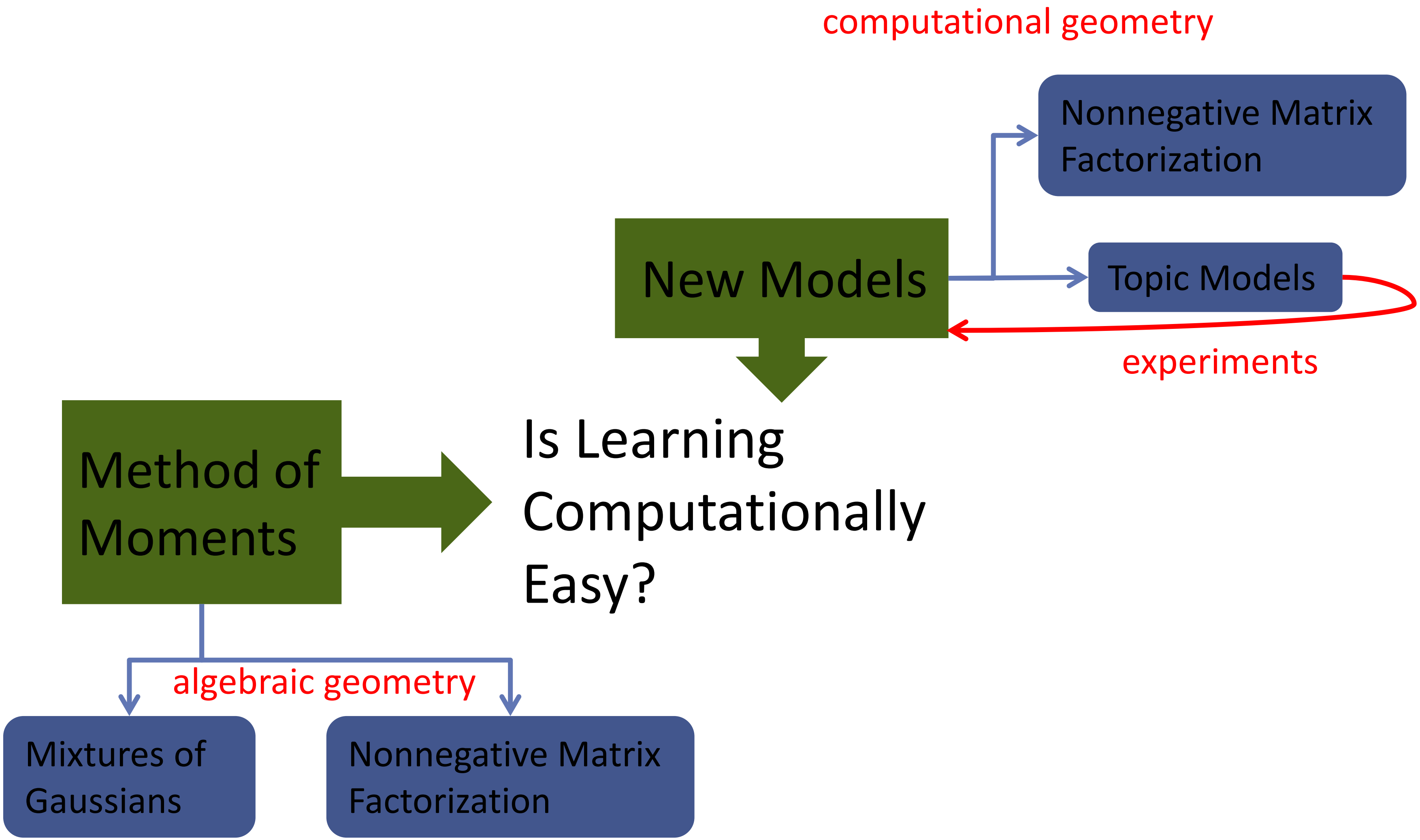
We ran our algorithm on real and synthetic data:

- synthetic data: train an LDA model on 1100 NIPS abstracts, use this model to run experiments

Our algorithm is **fifty times faster** and performs nearly the same on all metrics we tried (l_1, log-likelihood, coherence,…) when compared to MALLET

- real data: UCI collection of 300,000 NYT articles, 10 minutes!

# MY WORK ON LEARNING



# LEARNING MIXTURES OF GAUSSIANS

**Pearson (1896) and the Naples crabs:**

- Can we infer the parameters of a mixture of Gaussians from random samples?
- Introduced the **method of moments**, but no provable guarantees

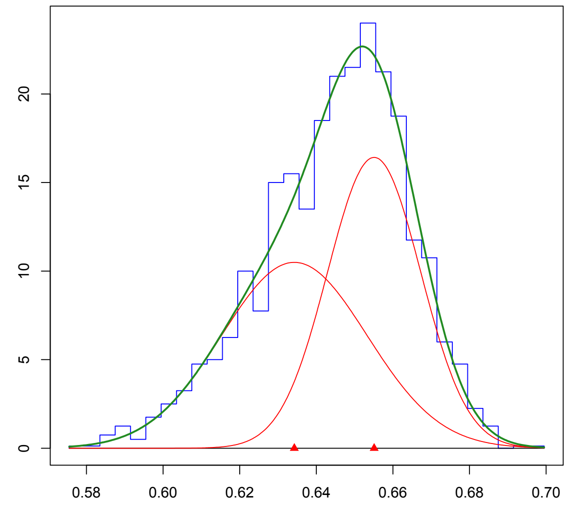
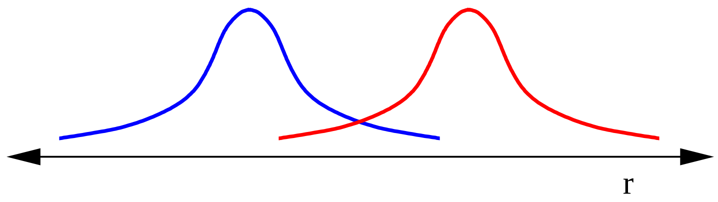


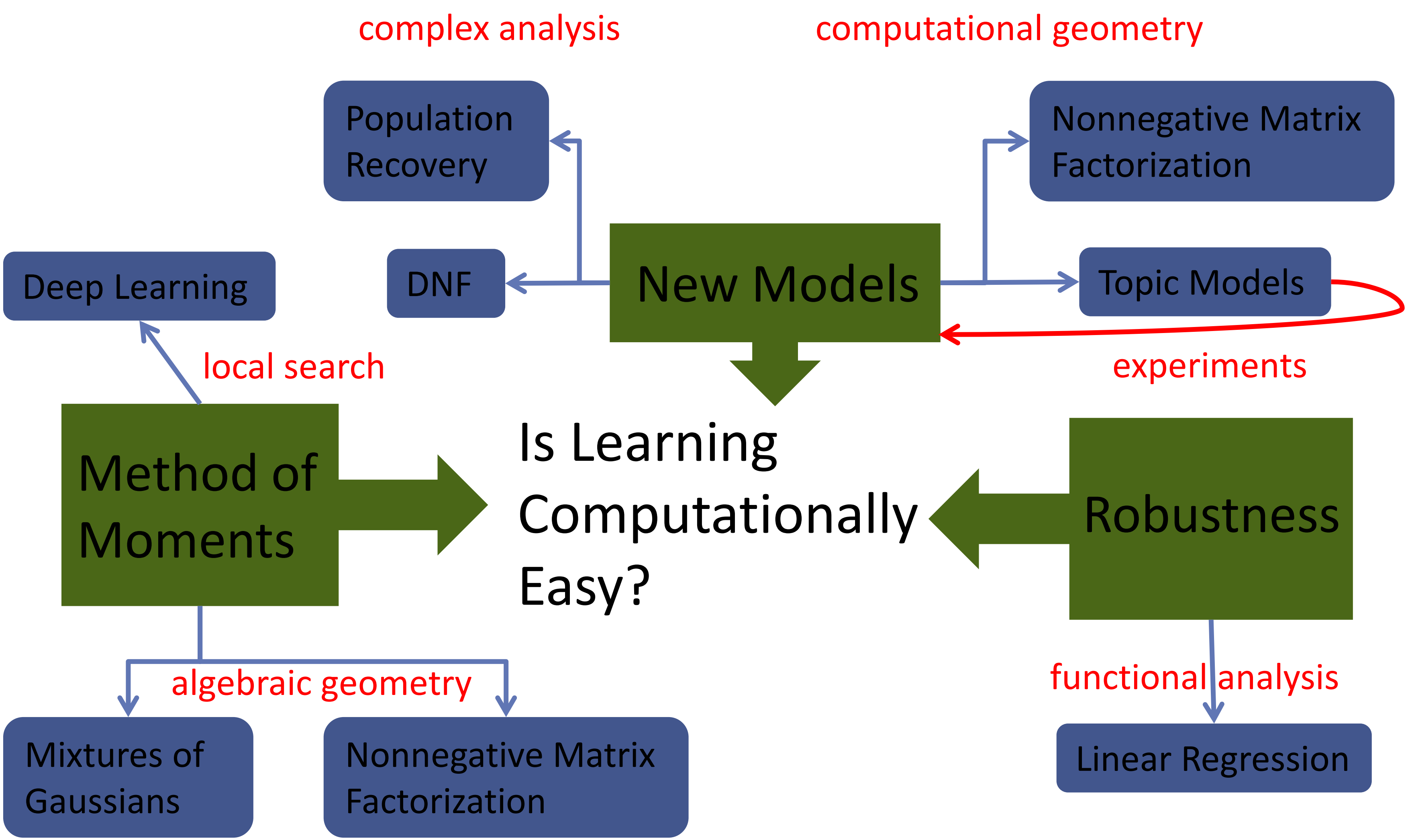

Image courtesy of Peter D. M. Macdonald. Used with permission.

**Theorem [Kalai, Moitra, Valiant STOC'10, FOCS'10]:** there is a polynomial time alg. to learn the parameters of a mixture of a constant number of Gaussians (even in high-dimensions)




This settles a long line of work starting with **[Dasgupta, '99]** that assumed **negligible overlap**. See also **[Belkin, Sinha '10]**

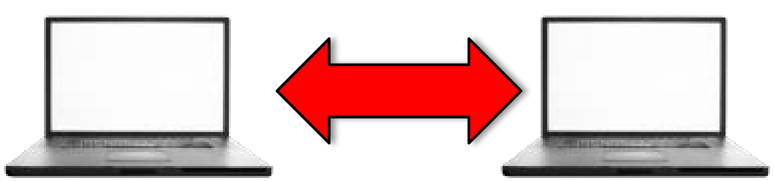
# MY WORK ON LEARNING



# MY WORK ON ALGORITHMS

Approximation Algorithms,
Metric Embeddings

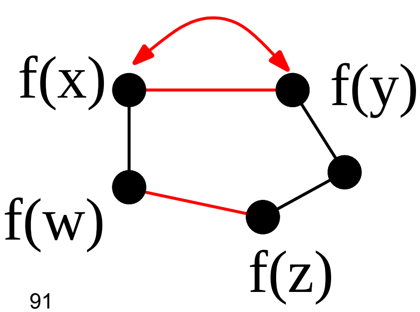


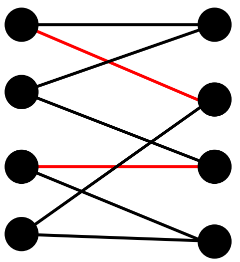


Information Theory,
Communication Complexity

Combinatorics,
Smooth Analysis

$$f(x+y) \stackrel{?}{=} f(x) + f(y)$$



# Any Questions?

**Summary:**

- Often optimization problems abstracted from learning are **intractable**!
- Are there new models that better capture the instances we actually want to solve in practice?
- These new models can lead to interesting **theory** questions and highly practical and **new** algorithms
- There are **many** exciting questions left to explore at the intersection of algorithms and learning