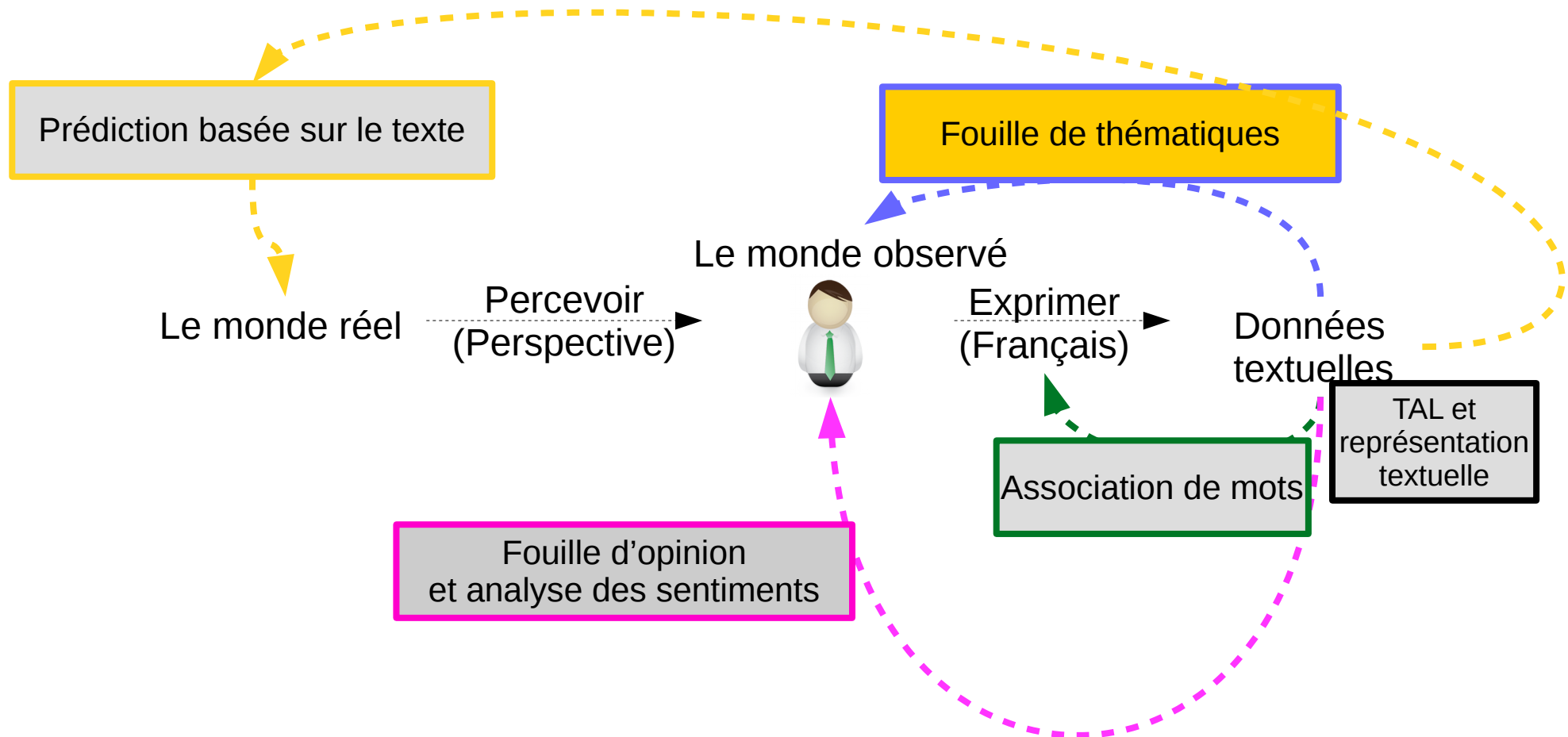


Algorithmes de classification, data mining et text mining

- M1 SID
- 2018-2019
- J. G. Moreno et Y. Pitarch

Paysage de la fouille et de l'analyse de textes



Fouille et modèles thématiques :


Motivation

- Sujet~ idée principale discutée dans une collection de textes
 - Thème/sujet d'une discussion ou d'une conversation
 - Différentes granularités (p. ex. le sujet d'une phrase, d'un article, etc.)
- De nombreuses applications nécessitent la découverte de sujets dans le texte
 - De quoi parlent les utilisateurs de Twitter aujourd'hui ?
 - Quels sont les sujets de recherche actuels en data mining ?
En quoi sont-ils différents de ceux d'il y a 5 ans ?
 - Qu'est-ce que les gens aiment de l'iPhone X ? Qu'est-ce qu'ils n'aiment pas ?
 - Quels ont été les principaux sujets débattus lors de l'élection présidentielle de 2016 ?

HomeNotificationsMessages

Search Twitter

Tweet



Jose G Moreno

@jgmorenof

Tweets

658

Following

466

Followers

208

Trends for you · Change

#soutienGW

3,726 Tweets

#MarcheRépublicaineDesLibertés

33.3K Tweets

#FoulardsRouges

23.5K Tweets

#STFCG

#AusOpenFinal

30.4K Tweets

#Djokovic

11.6K Tweets

#HolocaustMemorialDay

69.5K Tweets

#Auschwitz

25.8K Tweets


#WeRemember

60.5K Tweets

#Nadal

8,740 Tweets

What's happening?



Nando de Freitas @NandoDF · 5h

Really proud to work with these amazing researchers! This is an important must-read paper for anyone interested in multi-agent neural learning systems.



Max Jaderberg @maxjaderberg

New work on arxiv today: Open-ended Learning in Symmetric Zero-sum Games. A new framework for exploring what the optimisation objective should be in non-transitive games (think rock-paper-scissors, poker,...)

22


91



Google Ads @GoogleAds · Jan 9

Démarrez une campagne publicitaire en ligne dès aujourd'hui et développez votre activité grâce à Google Ads.

Translate Tweet



Google Ads

Essayez dès maintenant avec un avoir de 75€*.

google.com

1

9

Promoted


Leif Azzopardi Retweeted



Kyle Galbraith @kylegalbraith · 18h

There are some very handy #GitHub tricks in here that can really speed up some things... #Dev

Who to follow · Refresh · View all




Barbara Plank @barbara...

Follow



Computer Science @ Un...

Follow



Johannes Bjerva @johan...

Follow



Find people you know

Import your contacts from Gmail

Connect other address books

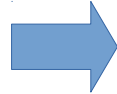
© 2019 Twitter About Help Center Terms Privacy policy Cookies Ads info Brand Blog Status Apps Jobs Marketing Businesses Developers

Advertise with Twitter

Extraction et de l'analyse du sujet

Tâche 2 : Déterminer quels documents couvrent quels sujets

Collection de textes



Thème 1

Thème 2

...

Thème k

Doc 1

Doc 2

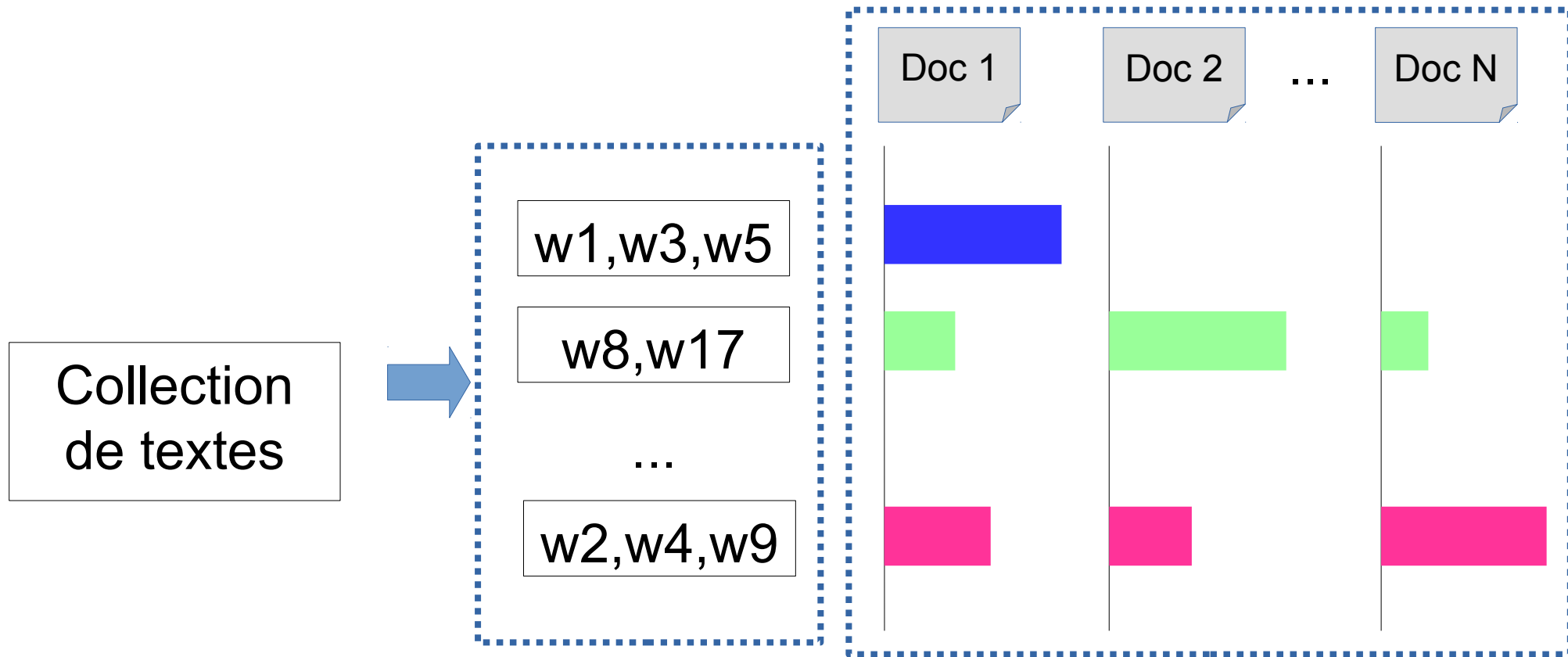
...

Doc N



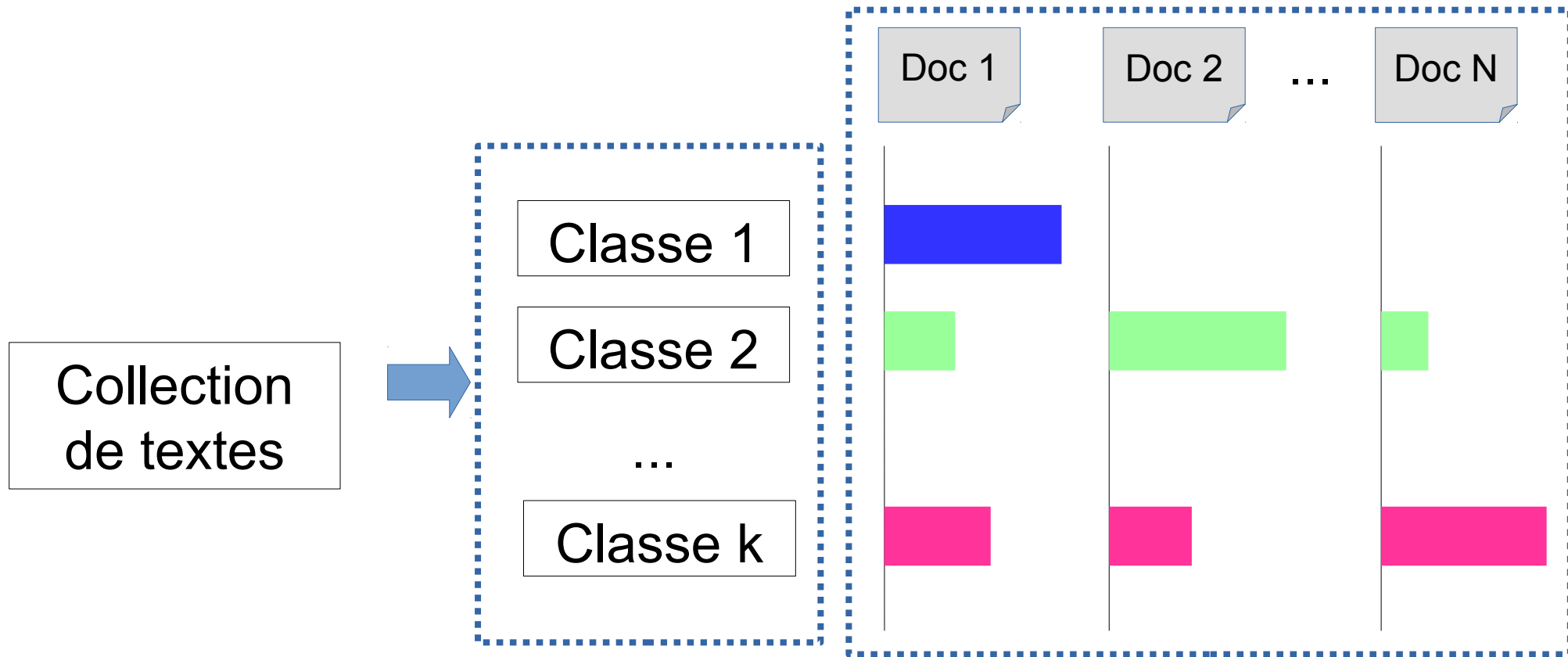
Tâche 1 : Découverte des k sujets/thème/thématiques/topiques

Extraction et de l'analyse du sujet



Problème non supervisé
(Clustering)

Extraction et de l'analyse du sujet



Problème supervisé (Classification)

Exploration et analyse du sujet

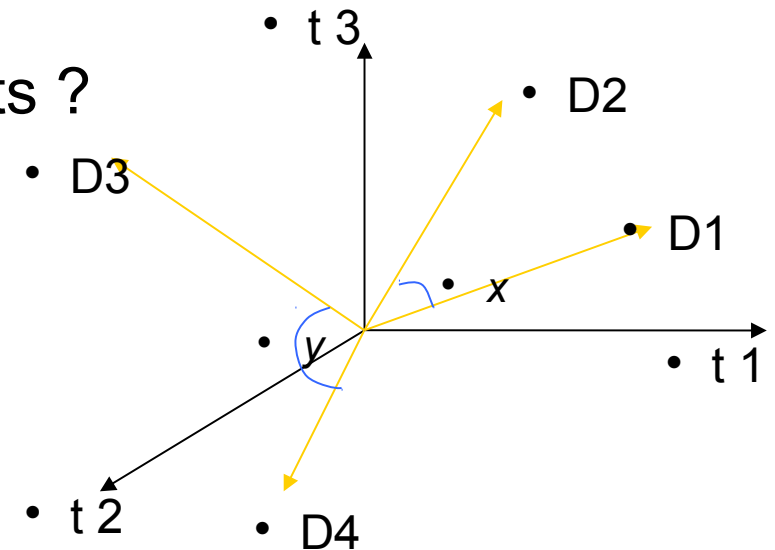
- Groupe de documents (non supervisé)
 - Regroupement
- Groupe de documents mais en classes (supervisé)
 - Classification
- ? Groupe de mots - réduction de la dimensionnalité (Nous allons les voir dans la partie non supervisée)
 - NMF, LDA, PLSA, etc.

Regroupement/Clustering

- Le regroupement de documents est un moyen de découvrir des documents et des sujets connexes.
- C'est le processus de regroupement d'un ensemble d'objets en classes d'objets similaires.
 - Les documents au sein d'une groupe doivent être similaires.
 - Les documents provenant de différents groupes devraient être dissemblables.
- De nombreux algorithmes existants
 - K-means, E.M., etc.
- Une représentation pour chaque document est exigée (ou similitude de document)
 - Matrice document-terme ou matrice terme-document
 - La fréquence est fréquemment utilisée, mais de nombreux autres modèles de pondération existent (tf-idf). Leur utilisation dépend de l'algorithme choisi (pas toujours tf-idf).

Défis du clustering

- Représentation pour le clustering
 - Quelle représentation de documents ?
 - Espace vectoriel ?
 - Normalisation ?
 - Besoin d'une notion de similarité/distance
- Combien de clusters ?
 - Défini a priori ?
 - Entièrement piloté par les données ?
 - Évitez les groupes "triviales"
 - trop grandes ou trop petites.



Matrice terme-document

	Antoine et Cléopâtre	Jules César	La Tempête	Hamlet	Othello	Macbeth
anthony	1	1	0	0	0	1
brutus	1	1	0	1	0	0
césar	1	1	0	1	1	1
calpurnia	0	1	0	0	0	0
cléopâtre	1	0	0	0	0	0
pitié	1	0	1	1	1	1
pire	1	0	1	1	1	0

Matrice binaire

	Antoine et Cléopâtre	Jules César	La Tempête	Hamlet	Othello	Macbeth
anthony	5.25	3.18	0.0	0.0	0.0	0.35
brutus	1.21	6.10	0.0	1.0	0.0	0.0
césar	8.59	2.54	0.0	1.51	0.25	0.0
calpurnia	0.0	1.54	0.0	0.0	0.0	0.0
cléopâtre	2.85	0.0	0.0	0.0	0.0	0.0
pitié	1.51	0.0	1.90	0.12	5.25	0.88
pire	1.37	0.0	0.11	4.15	0.25	1.95

TF-IDF

Les valeurs utilisées dans ces matrices sont connues sous le nom de schéma de pondération. Ils modifient les résultats obtenus (regroupement ou classification). La fréquence est souvent utilisée, mais d'autres options sont également disponibles :

Matrice terme-document

Term frequency		Document frequency		Normalization	
n (natural)	$tf_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(tf_{t,d})$	t (idf)	$\log \frac{N}{df_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2 + w_2^2 + \dots + w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times tf_{t,d}}{\max_t(tf_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N - df_t}{df_t}\}$	u (pivoted unique)	$1/u$
b (boolean)	$\begin{cases} 1 & \text{if } tf_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^\alpha$, $\alpha < 1$
L (log ave)	$\frac{1 + \log(tf_{t,d})}{1 + \log(\text{ave}_{t \in d}(tf_{t,d}))}$				

Pourquoi la fréquence n'est pas la meilleure option ?

En utilisant la fréquence, nous obtenons le vecteur {73, 157, 227, 10, 0, 0, 0, 0} pour représenter le document Jules César.

La taille du document affecte fortement les valeurs de similarité !!!!

Documents en tant que vecteurs

- Nous avons donc un espace vectoriel de taille $|V|$
- Les termes sont des axes de l'espace
- Les documents sont des points ou des vecteurs dans cet espace
- Très haute dimension : des millions de dimensions lorsque vous l'appliquez à de grandes collections.
- Ce sont des représentations creuses car la plupart des entrées sont nulles.

Notion de similitude/distance

- Idéal : similitude sémantique (dans nos rêves)
- Pratique : similitude terme-statistique
 - Similitude cosinus
 - Docs comme vecteurs
 - Pour de nombreux algorithmes, il est plus facile de penser en termes de distance (plutôt que de similarité) entre les documents.
 - Il est plus facile de parler de distance euclidienne, mais les implémentations réelles utilisent la similarité cosinus

Clustering dur vs clustering flou

- Regroupement dur : Chaque document appartient à un seul groupe de documents
 - Plus courant et plus facile à implémenter
- Regroupement flou : Un document peut appartenir à plusieurs clusters.
 - Plus utile pour les applications telles que la création de hiérarchies navigables.
 - Vous voudrez peut-être mettre une paire de baskets en deux groupes : (i) vêtements de sport et (ii) chaussures
 - Vous ne pouvez le faire qu'avec une approche de clustering flou (soft clustering).

Qu'est-ce qu'un bon regroupement ?

- Critère interne : Un bon clustering produira des clusters de haute qualité dans lesquels :
 - la similarité intra-classe est élevée
 - la similarité entre les classes est faible
 - La qualité mesurée d'un regroupement dépend à la fois de la représentation du document et de la mesure de similarité utilisée.
- Critère externe : La qualité d'un clustering se mesure aussi par sa capacité à découvrir tout ou partie des motifs cachés ou des classes latentes.
 - Évaluable à l'aide des données annotées

Algorithmes de clustering

- K-means
- Analyse sémantique latente (LSA)
- Factorisation matricielle non négative
- Allocation de Dirichlet latente
- Beaucoup d'autres, mais ce sont les principaux utilisés dans la mise en place d'un système de regroupement de textes

K-means

- C'est un algorithme de cluster dur
- Suppose que les documents sont des vecteurs de valeur réelle
- Regroupement basé sur des centroïdes (c'est-à-dire le centre de gravité ou la moyenne) de points d'un groupe

$$\vec{\mu}(c) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

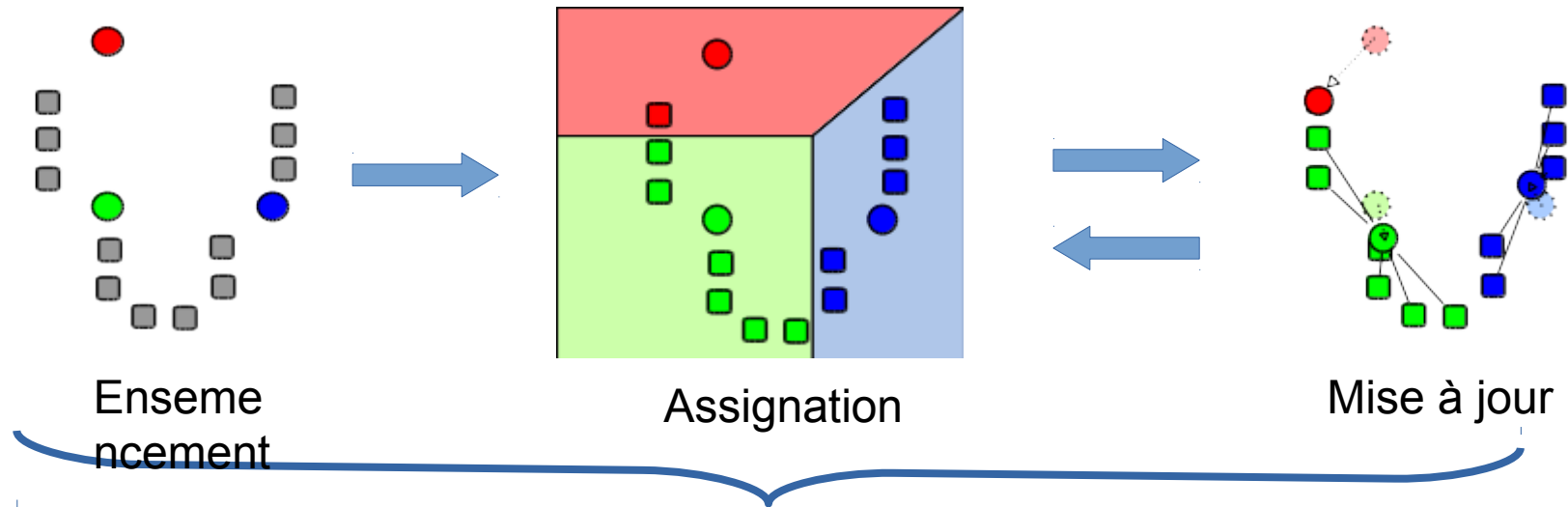
- La réaffectation des instances aux clusters est basée sur la distance par rapport aux centroïdes des clusters actuels (ou on peut l'exprimer de manière équivalente en termes de similarités).

Algorithme *K*-means

- Sélectionnez K documents aléatoires $\{s_1, s_2, \dots, s_K\}$ (ou points) comme graines ou centroïdes.
- Jusqu'à que le regroupement converge (ou un autre critère d'arrêt) :
 - Pour chaque doc d_i :
 - Affectez d_i au cluster c_j de sorte que $\text{dist}(x_i, s_j)$ soit minimal
 - Ensuite, mettre à jour les centroïdes de chaque groupe. Pour chaque groupe c_j

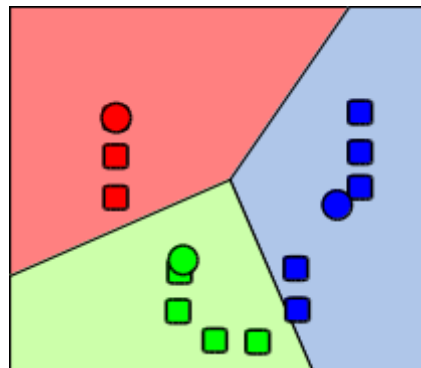
$$s_j = \vec{\mu}(c_j)$$

Itérations



Quand s'arrêter ?

- Basé sur des itérations
- Basé sur des critères d'optimisation



Partition
finale

L'implémentation par défaut dans sklearn utilise K-means++.

Message à emporter

- Facile à implémenter (déjà implémenté dans de nombreux frameworks, outils, bibliothèques, etc.)
- L'initialisation est une question importante (utiliser k-means++)
- La distance est aussi importante !
- Chaque groupe est considéré comme un sujet/thématique de la collection. Cependant, les termes sont inconnus
- Il a été étendu de nombreuses façons (soft, étiquettes, similarité, etc.)
- Comment introduire la similarité sémantique ?

Décomposition en valeurs et vecteurs propres

Pour une matrice $m \times n$, \mathbf{A} , de rang r , il existe une factorisation (Singular value decomposition = **SVD**) comme suit :

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$$

$m \times m$ $m \times n$ $n \times n$

Les colonnes de \mathbf{U} sont des vecteurs propres orthogonaux à $\mathbf{A}\mathbf{A}^T$.

Les colonnes de \mathbf{V} sont des vecteurs propres orthogonaux à $\mathbf{A}^T\mathbf{A}$.

Les valeurs propres $\lambda_1 \dots \lambda_r$ à $\mathbf{A}\mathbf{A}^T$ sont aussi des valeurs propres à $\mathbf{A}^T\mathbf{A}$.

$$\sigma_i = \sqrt{\lambda_i}$$
$$\mathbf{\Sigma} = \text{diag}(\sigma_1 \dots \sigma_r)$$

valeurs
propres

Décomposition en valeurs et vecteurs propres

- Illustration des dimensions SVD et de la faible densité

$$\underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_A = \underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_U \underbrace{\begin{bmatrix} \bullet & & & & \\ & \bullet & & & \\ & & \bullet & & \\ & & & \bullet & \\ & & & & \bullet \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_{V^T}$$

$$\underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_A = \underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_U \underbrace{\begin{bmatrix} \bullet & & & & \\ & \bullet & & & \\ & & \bullet & & \\ & & & \bullet & \\ & & & & \bullet \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_{V^T}$$

Approximation de rang inférieur

- L'UDS peut être utilisée pour calculer des **approximations** optimales **de bas rang**.
- Problème d'approximation : Trouver A_k de rang k tel que

$$A_k = \min_{X: \text{rank}(X)=k} \|A - X\|_F \longleftarrow \text{Frobenius norm}$$

$$\|A\|_F \equiv \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}.$$

A_k et X sont deux matrices $m \times n$. Il est souhaité

- $k \ll r$.

Approximation de rang inférieur

- Solution via SVD

$$A_k = U \operatorname{diag}(\sigma_1, \dots, \sigma_k, \underbrace{0, \dots, 0}_{\substack{\text{set smallest } r-k \\ \text{singular values to zero}}}) V^T$$

$$\underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_{A_k} = \underbrace{\begin{bmatrix} * & * & \text{red bar} \\ * & * & \text{red bar} \\ * & * & \text{red bar} \end{bmatrix}}_U \underbrace{\begin{bmatrix} \bullet & & \text{red bar} \\ & \bullet & \text{red bar} \\ & & \text{yellow bar} \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ \text{red bar} & \text{red bar} & \text{red bar} & \text{red bar} & \text{red bar} \\ \text{yellow bar} & \text{yellow bar} & \text{yellow bar} & \text{yellow bar} & \text{yellow bar} \\ \text{yellow bar} & \text{yellow bar} & \text{yellow bar} & \text{yellow bar} & \text{yellow bar} \end{bmatrix}}_{V^T}$$

$$A_k = \sum_{i=1}^k \sigma_i u_i v_i^T \leftarrow \text{notation par colonne : } \text{somme des matrices de rang 1}$$

Erreur d'approximation

- Dans quelle mesure cette approximation est-elle bonne (mauvaise) ?
- C'est le meilleur possible, mesuré par la norme de Frobenius de l'erreur :

$$\min_{X: \text{rank}(X)=k} \|A - X\|_F = \|A - A_k\|_F = \sigma_{k+1}$$

où les σ_i sont ordonnés de telle sorte que $\sigma_i \geq \sigma_{i+1}$. Suggère pourquoi l'erreur de Frobenius diminue à mesure que k augmente.

Projections aléatoires

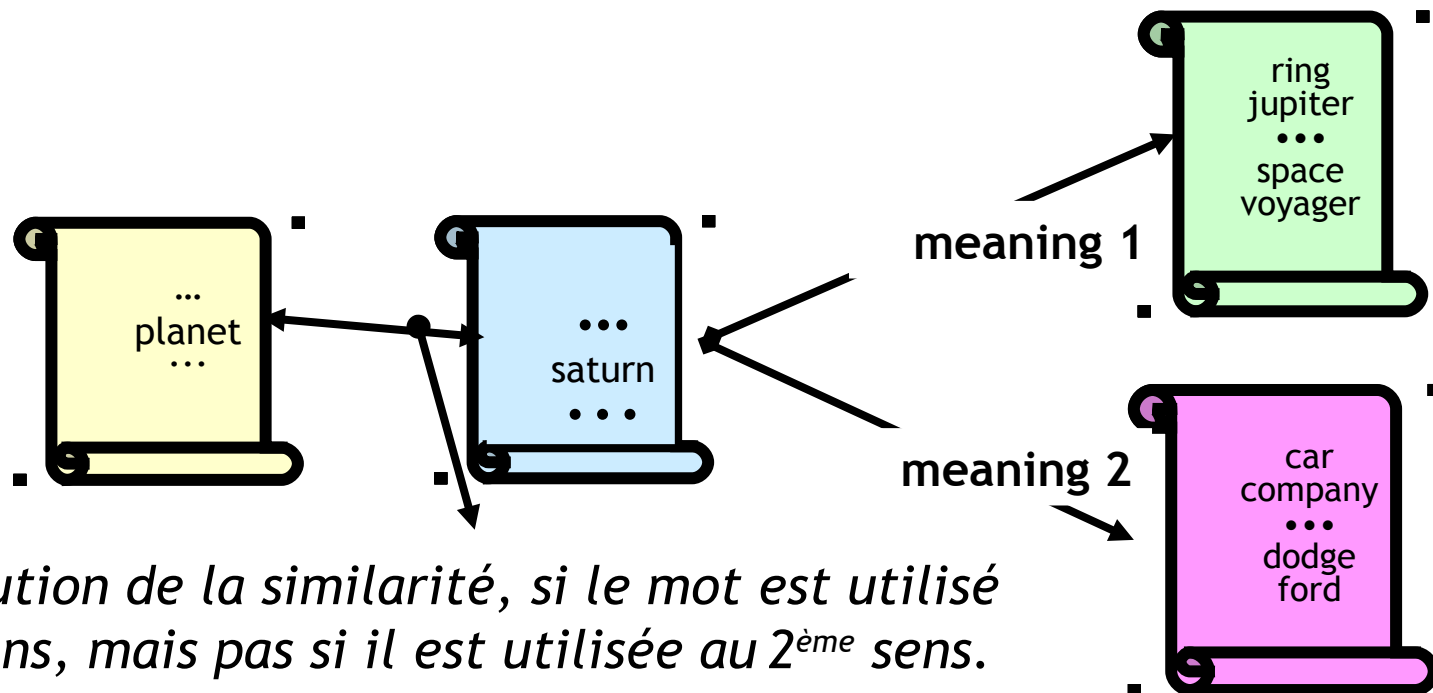
- Méthode complètement différente pour l'approximation de rang inférieur
- *Les données étaient-elles imprécises ?*
 - L'approximation basée sur le SVD *dépend des données.*
- L'erreur de projection aléatoire ne dépendait que de la dimensionnalité de début et de fin.
 - Pour chaque distance
- L'erreur pour l'approximation basée sur le SVD est pour la norme de Frobenius, et non pour les distances individuelles.

Analyse de sémantique latente

- A partir de la matrice terme-document A , nous calculons l'approximation A_k .
- Il y a une ligne pour chaque terme et une colonne pour chaque document dans A_k
- Ainsi, les documents vivent dans un espace de dimensions $k \ll r$
 - Ces dimensions ne sont pas les axes d'origine

Problèmes de sémantique lexicale

- Ambiguïté et association dans le langage naturel
 - **Polysémie** : Les mots ont souvent une **multitude de significations** et d'usages différents (*plus utile pour des collections très hétérogènes*).
 - **Synonymie** : Des termes différents peuvent avoir un **sens identique ou similaire** (mots indiquant le même sujet).



Analyse sémantique latente

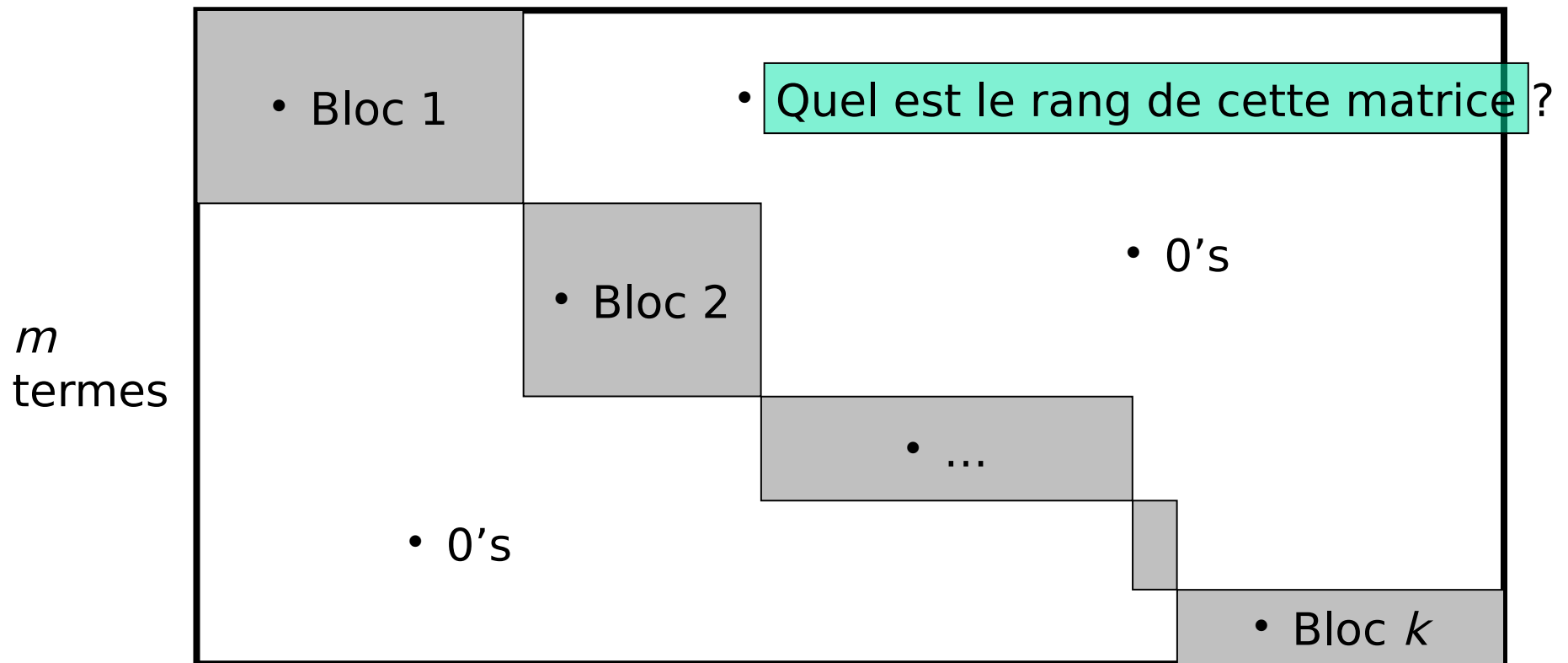
- Effectuer une **approximation de rang inférieur** de la **matrice document-terme** (rang type **100-300**)
- Idée générale
 - Projeter les documents (*et les termes*) à une représentation à **faible dimension**
 - Concevoir une projection de telle sorte que l'espace à faible dimension reflète les **associations sémantiques** (espace de sémantique latent).
- Objectifs
 - Des termes similaires correspondent à un emplacement similaire dans l'espace à faible dimension.
 - Réduction du bruit par réduction dimensionnelle

Mais pourquoi cela est du regroupement ?

- Nous avons parlé des documents et de la LSA.
- Qu'est-ce que cela a à voir avec le clustering ?
 - L'intuition : La réduction dimensionnelle par LSI permet de regrouper les axes "connexes" dans l'espace vectoriel.
- Appliquer l'algorithme k-means dans l'espace à faible dimension

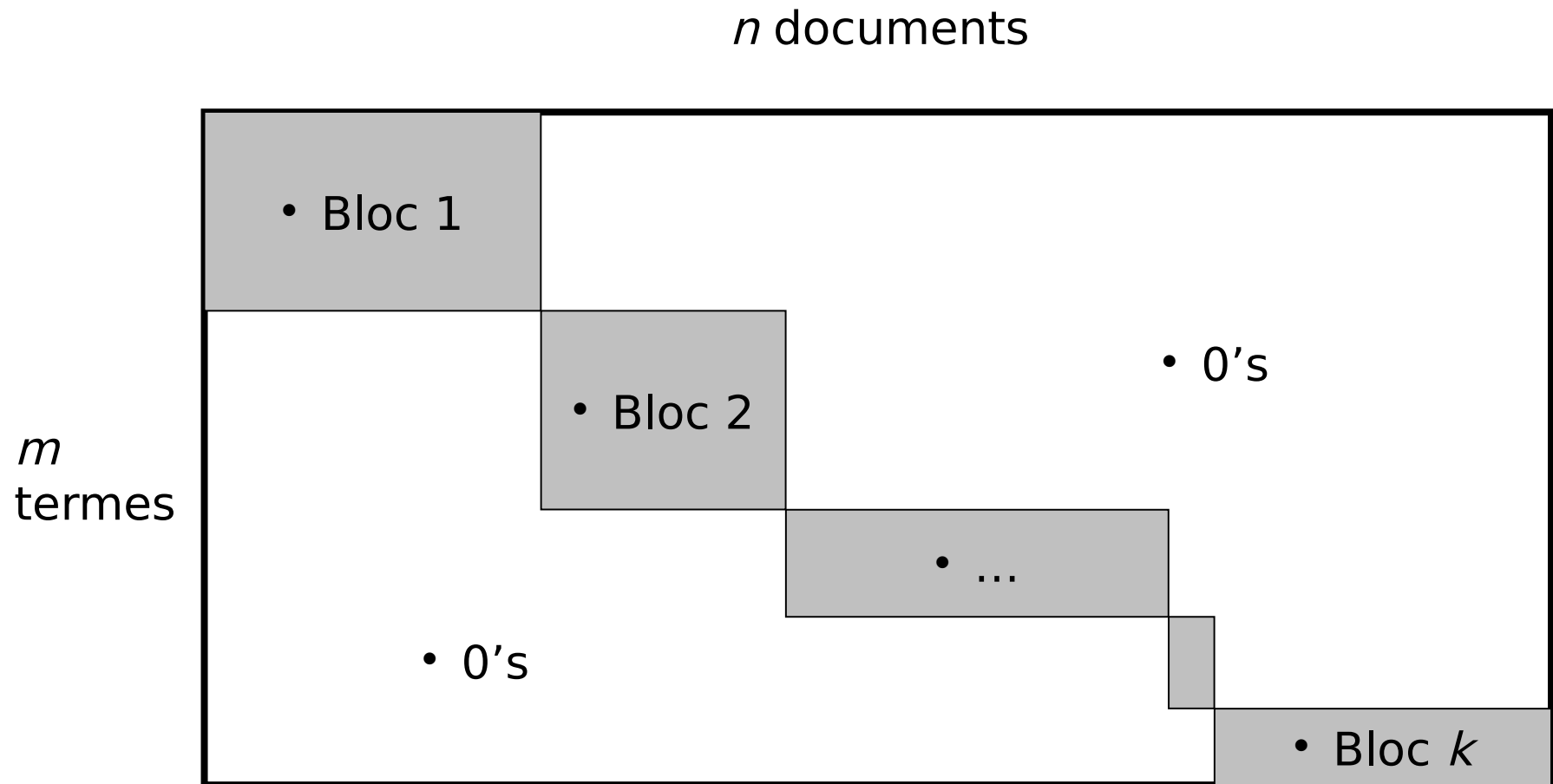
Intuition à partir de matrices de blocs

n documents



= entrées non nulles.

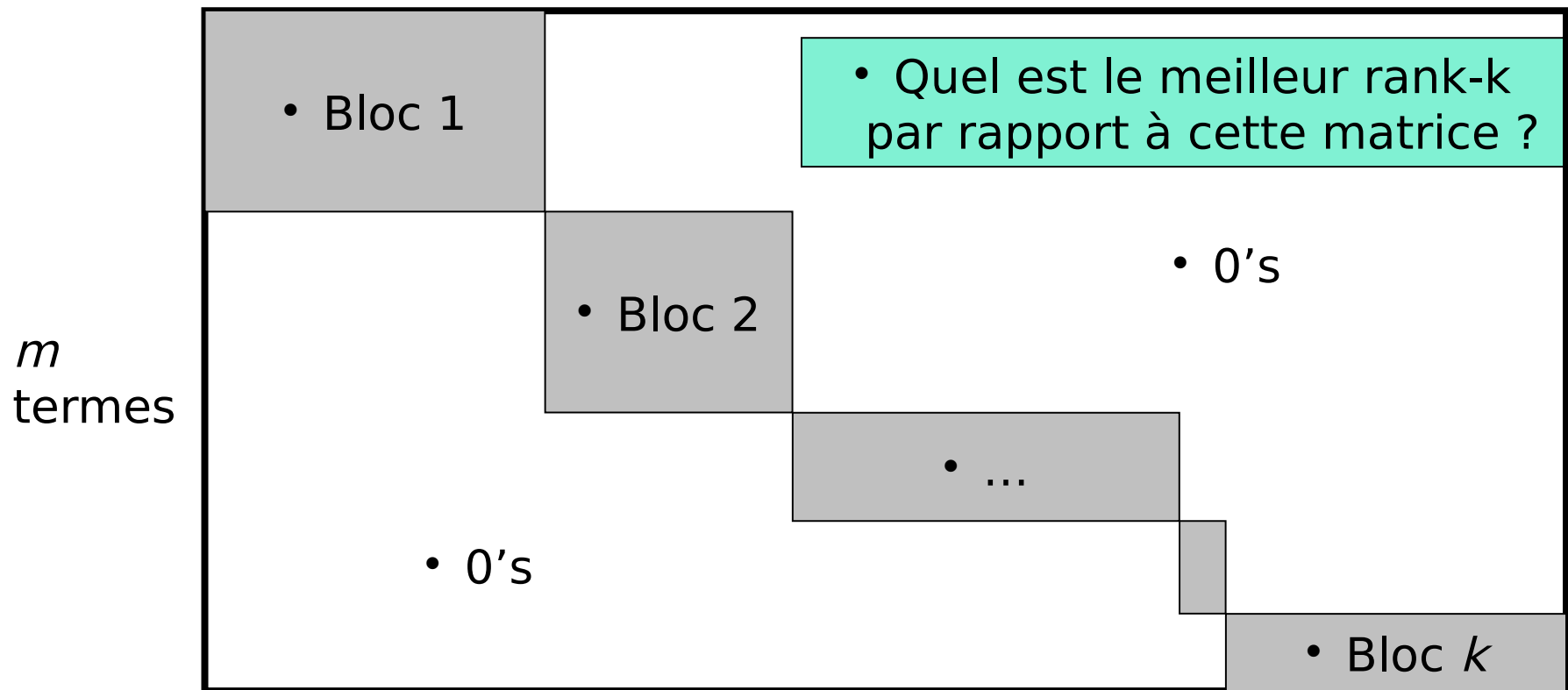
Intuition à partir de matrices de blocs



- Vocabulaire divisé en k sujets (clusters) ; chaque document ne traite que d'un seul sujet.

Intuition à partir de matrices de blocs

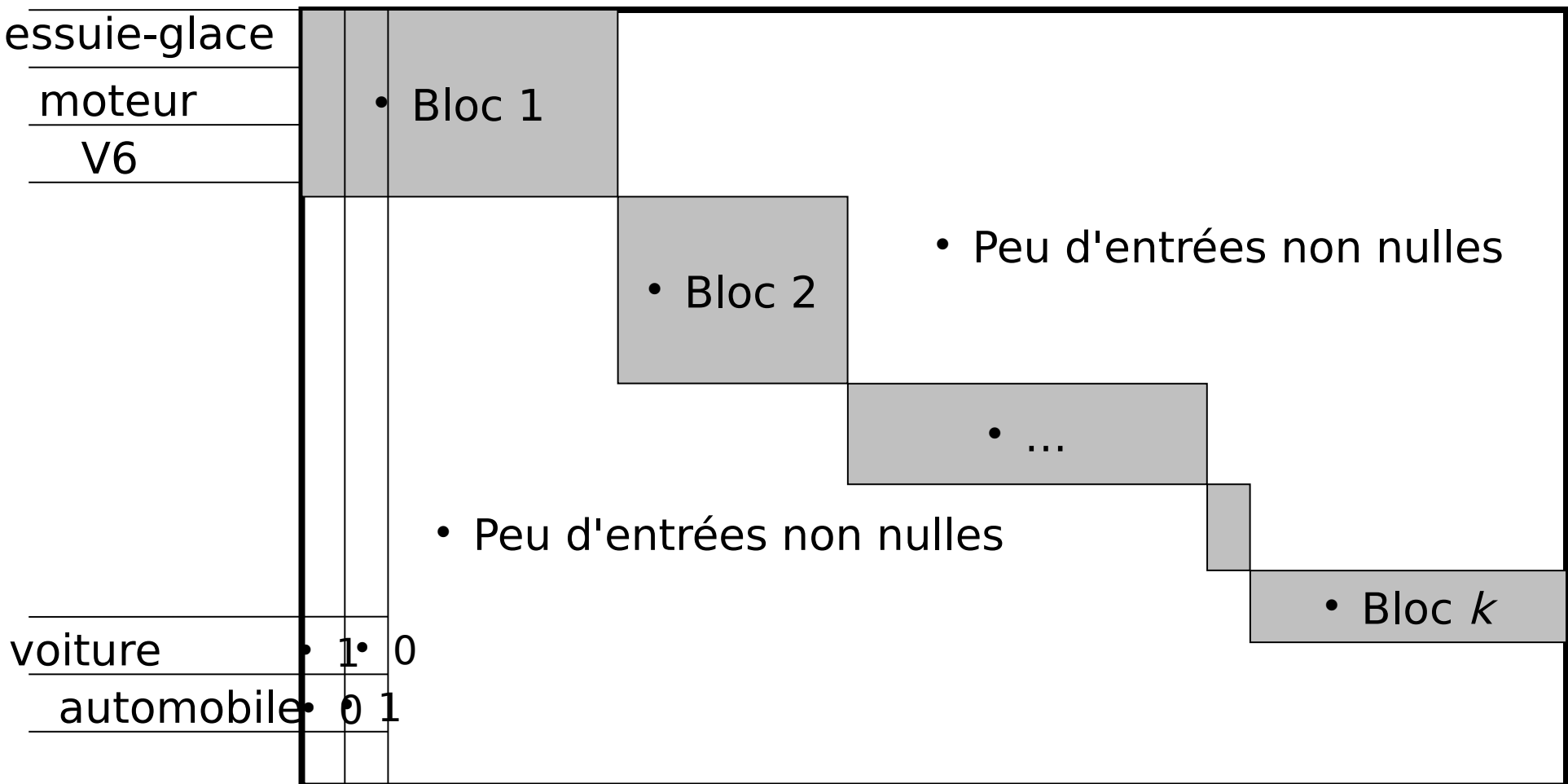
n documents



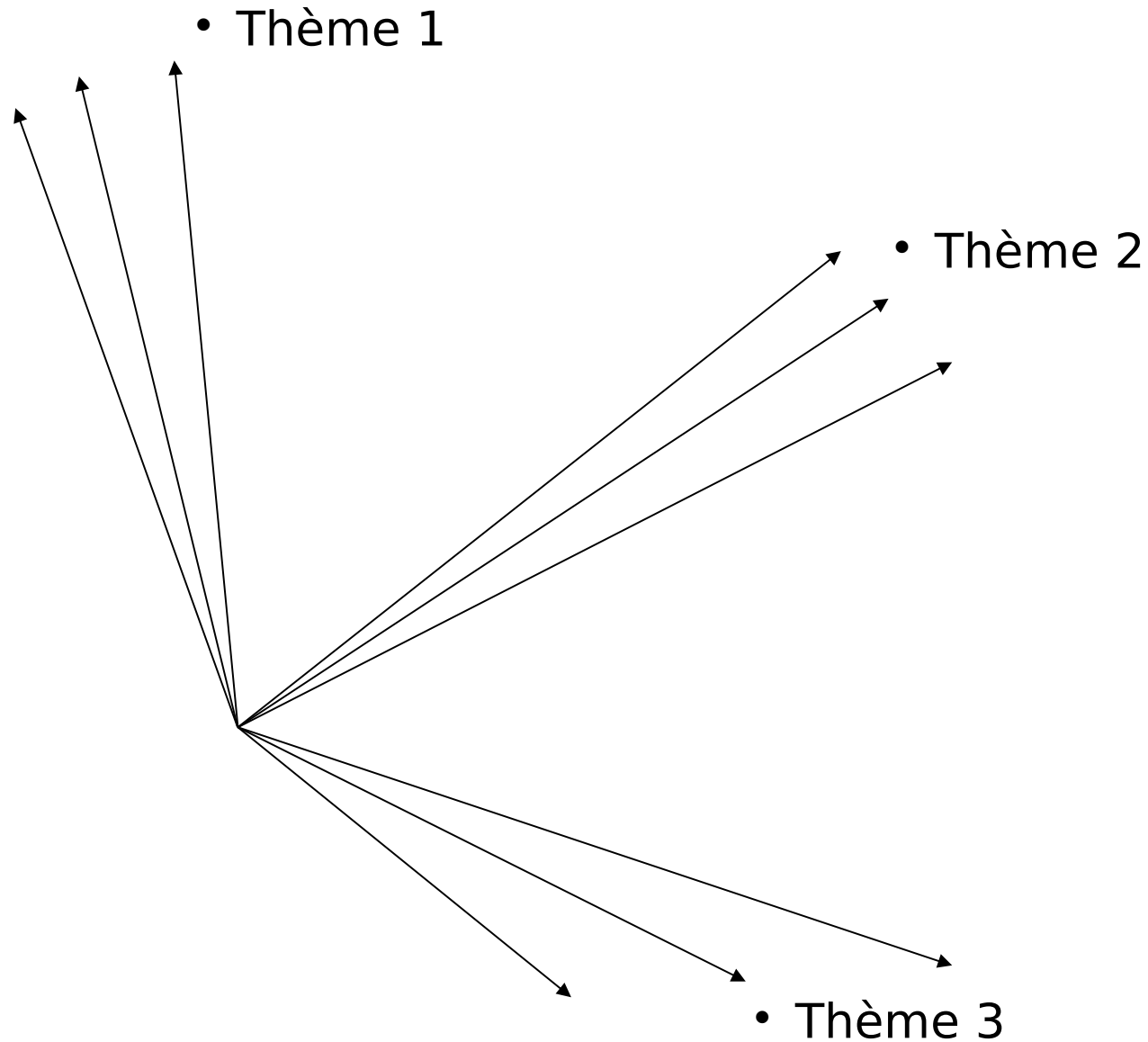
• = entrées non nulles.

Intuition à partir de matrices de blocs

Il y a probablement un bon rank-k par rapport à cette matrice.



Une image simpliste



Quelques extrapolations sauvages

- La "dimensionnalité" d'un corpus est le nombre de sujets distincts qui y sont représentés.
- Plus d'extrapolation mathématique sauvage
 - si A a une approximation de rang k de l'erreur de Frobenius basse, alors il n'y a pas plus de k sujets distincts dans le corpus.

LSA probabiliste

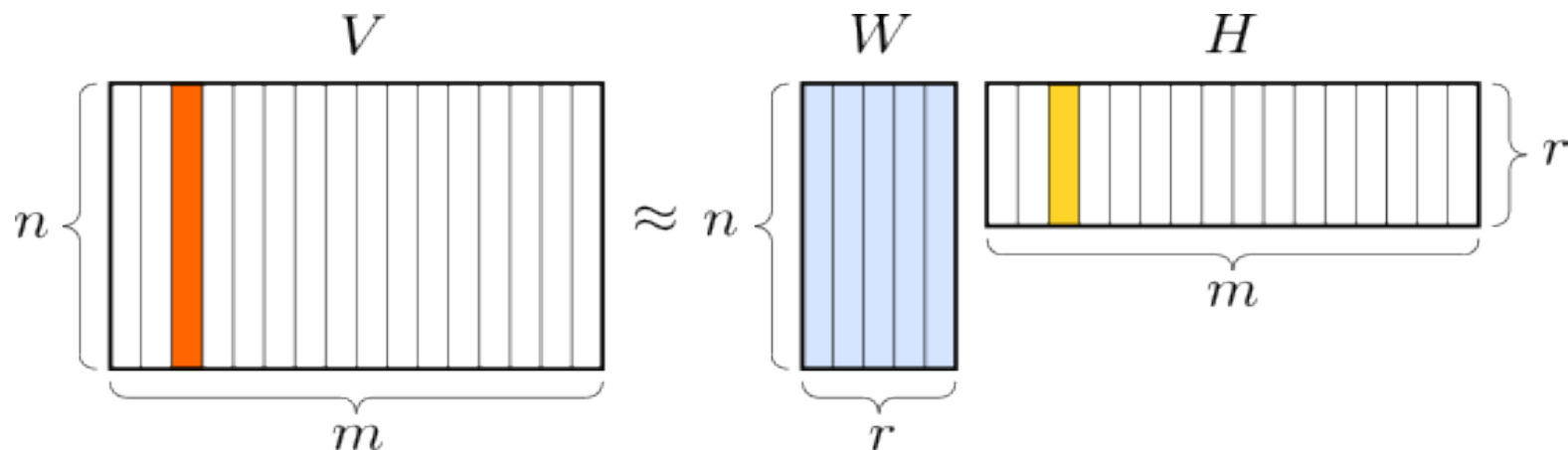
- Version probabiliste de LSA
- LSA a des facteurs négatifs, PLSA seulement positifs
- Plus facile à expliquer (qu'est-ce qu'une association de -0.8 à "maison" signifie ?)
- Sous certaines conditions spécifiques, il est similaire au NMF
- Elle n'est pas déterministe comme pourrait l'être la LSA (selon l'algorithme de SVD).

Message à emporter

- LSA réduit la collection de texte à un espace sémantique où les dimensions sont des sujets liés à la collection.
- Après l'application de la LSA, la collection peut être regroupée à l'aide de k-means
- Si le LSA est utilisé seul, la matrice terme-document doit être réduite au nombre de clusters souhaités.
- Si LSA est combiné avec K-means, le nombre de dimensions pour représenter les documents doit être plus élevé que le nombre souhaité de clusters
- Si une explication est nécessaire, mieux vaut utiliser PLSA.

Factorisation par matrices non négatives

- Approximation de rang inférieur de grandes matrices creuses
- Préserver la non-négativité des données
- Introduit le concept de la représentation basée sur des pièces



LSA vs. NMF

• LSA

- Produire des vecteurs de base avec des entrées négatives
- Les combinaisons additives et soustractives de vecteurs de base donnent des vecteurs de document originaux.

NMF

- Produit des vecteurs de base non négatifs
- Combinaison additive de vecteurs de base permettant d'obtenir un vecteur de document original
- Les vecteurs de base sont interprétés comme des caractéristiques ou des sujets sémantiques.
- Documents regroupés sur la base de fonctionnalités partagées

NMF : Définition

Le NMF est défini comme suit :

- Étant donné
 - S : Collection de textes
 - $V_{m \times n}$: matrice terme par document
 - m : nombre de termes (dictionnaire)
 - n : nombre de documents en S
 - Approximation de rang inférieur de $V_{m \times n}$ en termes de certaines métriques
- V comme le produit WH
 - $W_{m \times k}$: Contient les bases (vecteurs)
 - $H_{k \times n}$: Contient les combinaisons linéaires
 - k : nombre de sujets ou bases sélectionnés,
 - $k \ll \min(m, n)$

NMF : Approche simplifié

- Minimiser la fonction objectif :

$$\min_{W,H} \|V - WH\|_F^2 \quad V_{i\mu} \approx (WH)_{i\mu} = \sum_{a=1}^r W_{ia} H_{a\mu}$$

$W_{ia} \leftarrow W_{ia} \sum_{\mu} \frac{V_{i\mu}}{(WH)_{i\mu}} H_{a\mu}$ $W_{ia} \leftarrow \frac{W_{ia}}{\sum_j W_{ja}}$	$H_{a\mu} \leftarrow H_{a\mu} \sum_i W_{ia} \frac{V_{i\mu}}{(WH)_{i\mu}}$
--	---

NMF

Méthode multiplicative (MM)

- Basé sur des règles de mise à jour multiplicatives
- $\|V - WH\|$ est monotone non croissant et constant si et seulement si W, H sont au point fixe
- Version du schéma d'optimisation de la descente de gradient (GD)
- Encodage à faible densité
 - Basé sur l'étude des réseaux de neurones
 - Augmente la rareté statistique de la matrice H
 - Minimise la somme des valeurs non nulles de H
- Initialisation : par défaut, elle est aléatoire mais peut être améliorée en utilisant SVD

Message à emporter

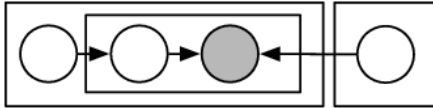
- NMF peut être vu comme une LSA mais seulement avec des valeurs positives (en effet PLSA et NMF sont des méthodes liées).
- L'association positive à las thématiques rend inutile l'étape k-means utilisée pour la LSA
- Le nombre de thématiques peut être directement le nombre de groupes désirés où l'assignation de document est définie par la valeur la plus élevée parmi les thématiques.
- L'initialisation correcte améliore fortement les résultats obtenus (cas similaire à k-means++)

Allocation de Dirichlet latente

- Les modèles thématiques sont une suite d'algorithmes permettant de découvrir les principales thématiques qui imprègnent une vaste collection de documents non structurés.
- Parmi ces algorithmes, la technique LDA, basée sur la modélisation bayésienne, est la plus couramment utilisée de nos jours.
- Les modèles thématiques peuvent être appliqués à des collections massives de documents pour organiser, comprendre, rechercher et résumer automatiquement de grandes archives électroniques.
- Particulièrement pertinent dans l'environnement "Big Data" d'aujourd'hui.

Motivation pour les modèles thématiques

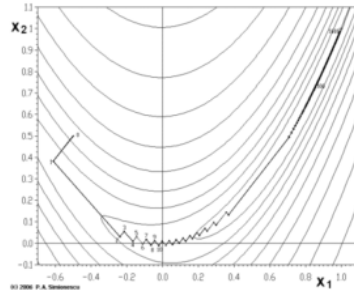
Assumptions



Data



Inference algorithm



Discovered structure

Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts

Gerard Salton, James Allan, Chris Buckley

Most amounts of text material are now available in machine-readable form. This approach is well suited for the automatic analysis and summarization of machine-readable texts. In particular, such systems can be used to extract relevant information from large volumes of text, to generate summaries, and to extract relevant information from large volumes of text.

Most kinds of texts are currently available in machine-readable form and are amenable to automatic processing. Because the available databases are large and cover many different subject areas, automatic aids must be provided to users interested in accessing the data. It has been suggested that links be placed between related pieces of text, connecting, for example, particular text paragraphs to other paragraphs covering related subject matter. Such a linked text structure, often called hypertext, makes it possible for the reader to move with particular ease through the text and to extract relevant information. In addition, it is possible to build a linked text structure that is useful for the automatic analysis and summarization of machine-readable texts. This approach is well suited for the automatic analysis and summarization of machine-readable texts. In particular, such systems can be used to extract relevant information from large volumes of text, to generate summaries, and to extract relevant information from large volumes of text.

Text Analysis and Retrieval: The Smart System

The Smart System is a sophisticated text analysis and retrieval system. It is designed to handle large volumes of text and to extract relevant information from them. It is well suited for the automatic analysis and summarization of machine-readable texts. In particular, such systems can be used to extract relevant information from large volumes of text, to generate summaries, and to extract relevant information from large volumes of text.

"Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts" (1994)

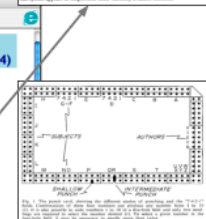
TOPIC	PROB.
data computer system information network	0.30
information library text index libraries	0.19
two three four different single	0.16

DOCUMENT	SCORE
"Global Text Matching for Information Retrieval" (1991)	0.2570
"Automatic Text Analysis" (1970)	0.3110
"Gauging Similarity with n-Grains: Language-Independent Categorization of Text" (1993)	0.3210
"Developments in Automatic Text Retrieval" (1991)	0.3680
"Simple and Rapid Method for the Coding of Punched Cards" (1962)	0.3610
"Data Processing by Optical Coincidence" (1961)	0.4290
"Pattern-Analyzing Memory" (1976)	0.4320
"The Sorting of Pamphlets" (1991)	0.4440
"A Punched-Card Technique for Computing Means, Standard Deviations, and the Product-Moment Correlation Coefficient and for Listing Scattergrams" (1946)	0.4550

Global Text Matching for Information Retrieval

James Allan, Gerard Salton

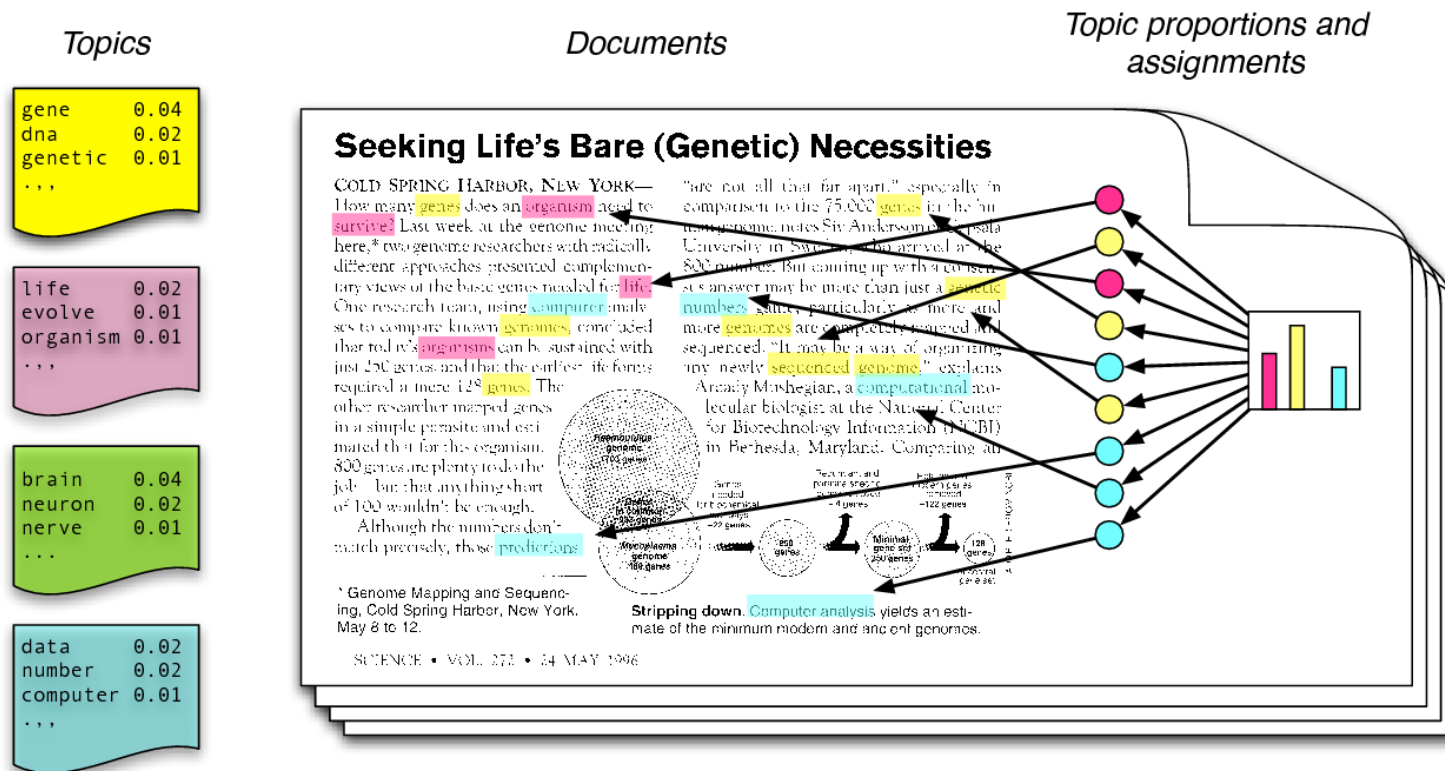
This approach is well suited for the automatic analysis and summarization of machine-readable texts. In particular, such systems can be used to extract relevant information from large volumes of text, to generate summaries, and to extract relevant information from large volumes of text.



THE HISTORY OF PAMPHLETS

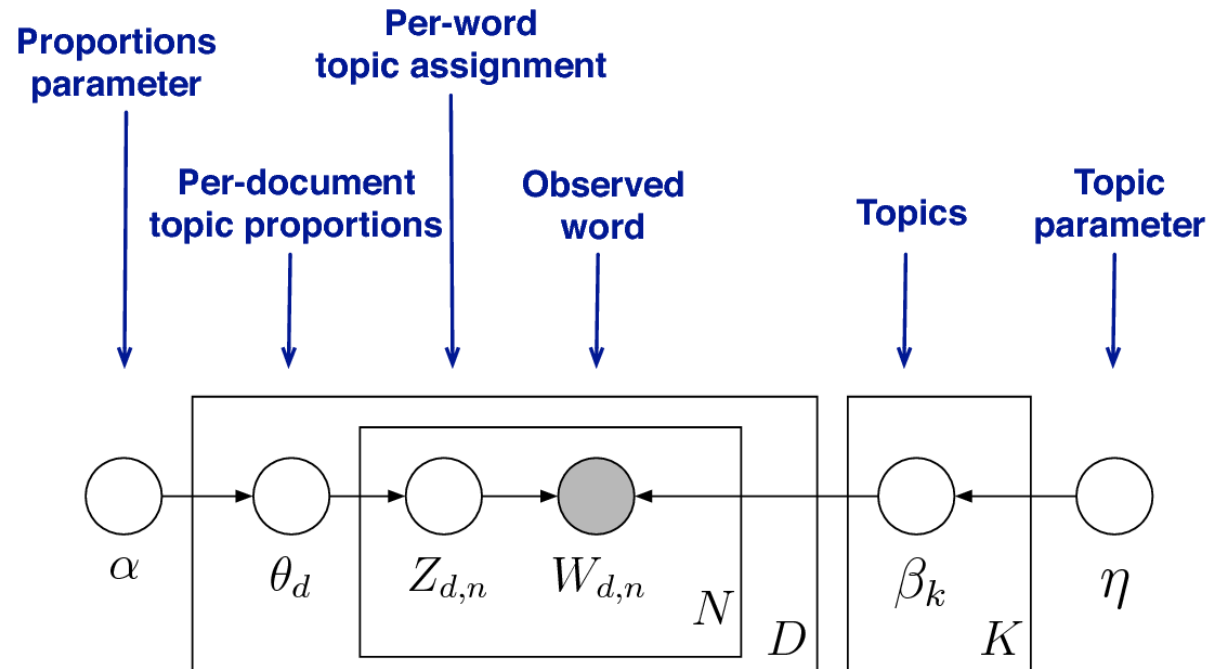
The history of pamphlets is a long and interesting one. It is well suited for the automatic analysis and summarization of machine-readable texts. In particular, such systems can be used to extract relevant information from large volumes of text, to generate summaries, and to extract relevant information from large volumes of text.

LDA



- Chaque thématique est une distribution de mots ; chaque document est un mélange de thématiques à l'échelle du corpus ; et chaque mot est tiré d'un de ces thématiques.
- Le but est de déduire les variables cachées
Calculer leur distribution conditionnée par les documents
 $p(\text{thématique, proportions, affectations} \mid \text{documents})$

modèle LDA



$$\prod_{i=1}^K p(\beta_i | \eta) \prod_{d=1}^D p(\theta_d | \alpha) \left(\prod_{n=1}^N p(z_{d,n} | \theta_d) p(w_{d,n} | \beta_{1:K}, z_{d,n}) \right)$$

- Notation
 - Les nœuds sont des variables aléatoires ; les arcs indiquent une dépendance
 - Les nœuds ombragés sont observés
 - Les plaques indiquent les variables répliquées

Histoire générative de LDA

- Si θ_d et β_k sont connus (définis par les paramètres), nous pouvons construire n'importe quel document en suivant :
 1. Sélectionnez autant de thématiques que la taille du document selon une distribution multinomiale définie par θ_d
 2. Pour chaque thématique sélectionnée, sélectionnez un mot pour représenter la thématique selon une distribution multinomiale définie par β_k

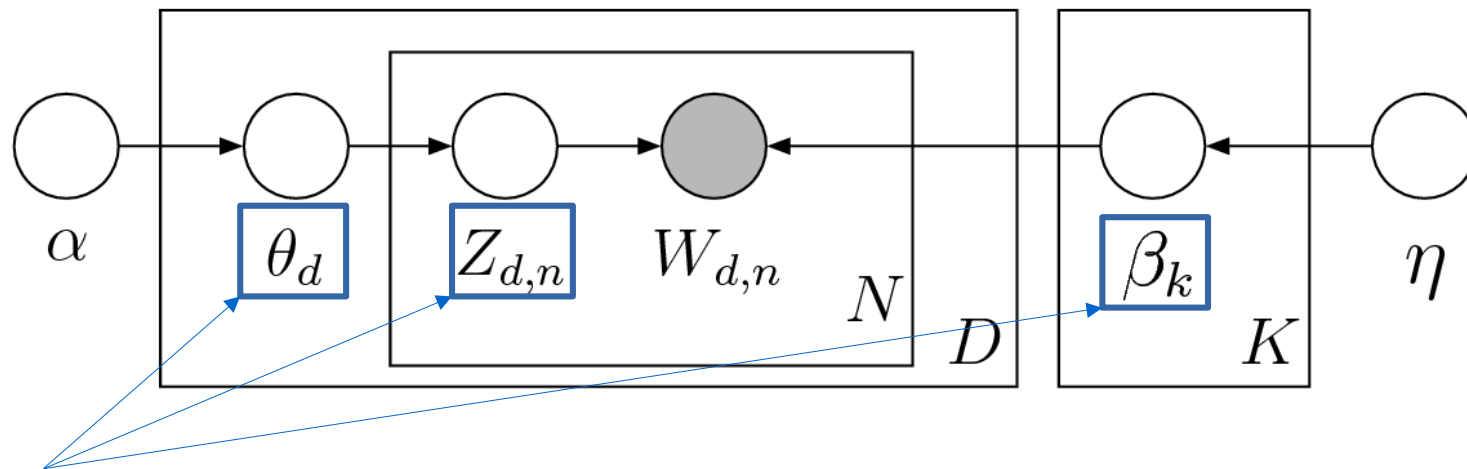
Calcul analytique

$$p(Z, \theta, \beta | w; \alpha, \eta) = \frac{p(w, Z, \theta, \beta; \alpha, \eta)}{p(w; \alpha, \eta)}$$

$$p(w, Z, \theta, \beta; \alpha, \eta) = \left\{ \prod_{d=1}^D \prod_{n=1}^N p(w_{dn} | \beta_{Z_{dn}}) p(Z_{dn} | \theta_d) \right\} \left\{ \prod_{d=1}^D p(\theta_d; \alpha) \right\} \left\{ \prod_{k=1}^K p(\beta_k; \eta) \right\} +$$

$$p(w; \alpha, \eta) = \int_{\theta} \int_{\beta} \sum_Z p(w, Z, \theta, \beta; \alpha, \eta) d\beta d\theta \quad \times$$

Estimation du modèle



- Algorithmes approximatifs d'inférence postérieure
 - Méthodes variationnelles de champ moyen
 - Propagation des attentes
 - Échantillonnage de Gibbs effondré*
 - Inférence variationnelle effondrée
 - **Inférence variationnelle en ligne**

Message à emporter

- LDA est un modèle thématique basé sur la version probabiliste de LSI
- L'estimation des paramètres dans le modèle est effectuée sur la base des données observées et des paramètres.
- Le regroupement peut être effectué à l'aide de LDA en choisissant le sujet le plus représentatif et en fixant le nombre désiré de sujets comme le nombre de regroupements.

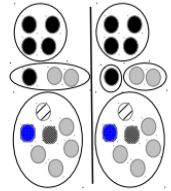
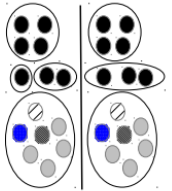
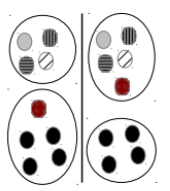
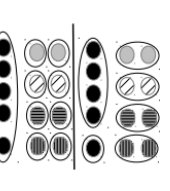
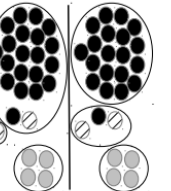
En savoir plus sur le clustering

- De nombreux algorithmes sont disponibles pour le regroupement, mais beaucoup d'entre eux ne sont pas adaptés au texte
- Les algorithmes classiques fonctionnent généralement bien, mais les algorithmes de l'état de l'art peuvent fonctionner beaucoup mieux s'ils sont bien paramétrés (le théorème no-free-lunch).
- Le regroupement est une tâche difficile et peu d'algorithmes s'adaptent bien à de grandes collections de textes. Si des hiérarchies sont nécessaires, des stratégies ascendantes ou descendantes peuvent être appliquées (combinées avec les algorithmes présentés).
- Il existe des problèmes pour lesquels des étiquettes sont nécessaires !!!! LDA est une bonne solution mais d'autres algorithmes plus simples peuvent faire l'affaire (STC, Lingo, etc.)
- Comme toute autre tâche d'exploration de données, le meilleur algorithme de clustering sera dessiné par le problème et non en choisissant le plus populaire.

Comment évaluer le clustering ?

- L'évaluation est une tâche difficile.
- Meilleur scénario : une collection annotée est disponible
 - Des paires de documents sont utilisées pour évaluer les partitions en demandant si elles appartiennent à la même partition dans les données annotées et dans la partition obtenue (pour tout algorithme de clustering).
- Les cas extrêmes sont difficiles à évaluer
 - Tous les documents appartiennent à un seul cluster
 - Tous les documents appartiennent à des clusters individuels

Comparaison des métriques d'évaluation de clustering

	C. Homogeneity			C. Completeness			Rag Bag			C. size vs q.			Unbalanced			4 + 1 F.C.
																
Purity	0.71	0.79	✓	0.79	0.79	✗	0.56	0.56	✗	1.00	1.00	✗	0.96	0.96	✗	✗
Inv. Purity	0.79	0.79	✗	0.79	0.79	✗	1.00	1.00	✗	0.69	0.92	✓	0.96	0.96	✗	✗
F&M	0.47	0.49	✓	0.47	0.53	✓	0.61	0.61	✗	0.85	0.85	✗	0.95	0.94	✗	✗
RandIndex	0.68	0.70	✓	0.68	0.70	✓	0.72	0.72	✗	0.95	0.95	✗	0.94	0.94	✗	✗
Adj.RandIndex	0.25	0.28	✓	0.24	0.31	✓	0.40	0.40	✗	0.80	0.80	✗	0.79	0.79	✗	✗
Jaccard	0.31	0.33	✓	0.31	0.36	✓	0.38	0.38	✗	0.71	0.71	✗	0.90	0.89	✗	✗
F-measure	0.71	0.79	✓	0.79	0.79	✗	0.56	0.56	✗	1.00	1.00	✗	0.96	0.96	✗	✗
P_{b^3}	0.60	0.69	✓	0.69	0.69	✗	0.49	0.56	✓	1.00	1.00	✗	0.93	0.95	✓	✗
R_{b^3}	0.70	0.70	✗	0.71	0.76	✓	1.00	1.00	✗	0.69	0.88	✓	0.96	0.93	✗	✗
F_{b^3}	0.64	0.69	✓	0.70	0.72	✓	0.55	0.71	✓	0.82	0.93	✓	0.94	0.93	✗	✗
$P_{b^3}^{mod}$	0.60	0.69	✓	0.69	0.69	✗	0.49	0.56	✓	1.00	1.00	✗	0.93	0.95	✓	✗
$R_{b^3}^{mod} (\vec{x} = 3)$	0.45	0.45	✗	0.56	0.57	✓	1.00	1.00	✗	0.46	0.77	✓	0.93	0.86	✗	✗
$F_{b^3}^{mod\&0.9}$	0.58	0.66	✓	0.67	0.68	✓	0.52	0.58	✓	0.90	0.97	✓	0.93	0.95	✓	✓
$F_{b^3}^{0.9}$	0.61	0.70	✓	0.69	0.70	✓	0.52	0.58	✓	0.96	0.99	✓	0.93	0.94	✓	✓

Message à emporter

- Évaluer si vous avez des données annotées
 - La sélection de la métrique est une question importante
 - Essayez de comprendre le problème abordé pour sélectionner une métrique adaptée à celui-ci.
- Les outils implémentent également des métriques d'évaluation
 - Module d'évaluation des performances de clustering dans scikit-learn
 - Clusteval en R