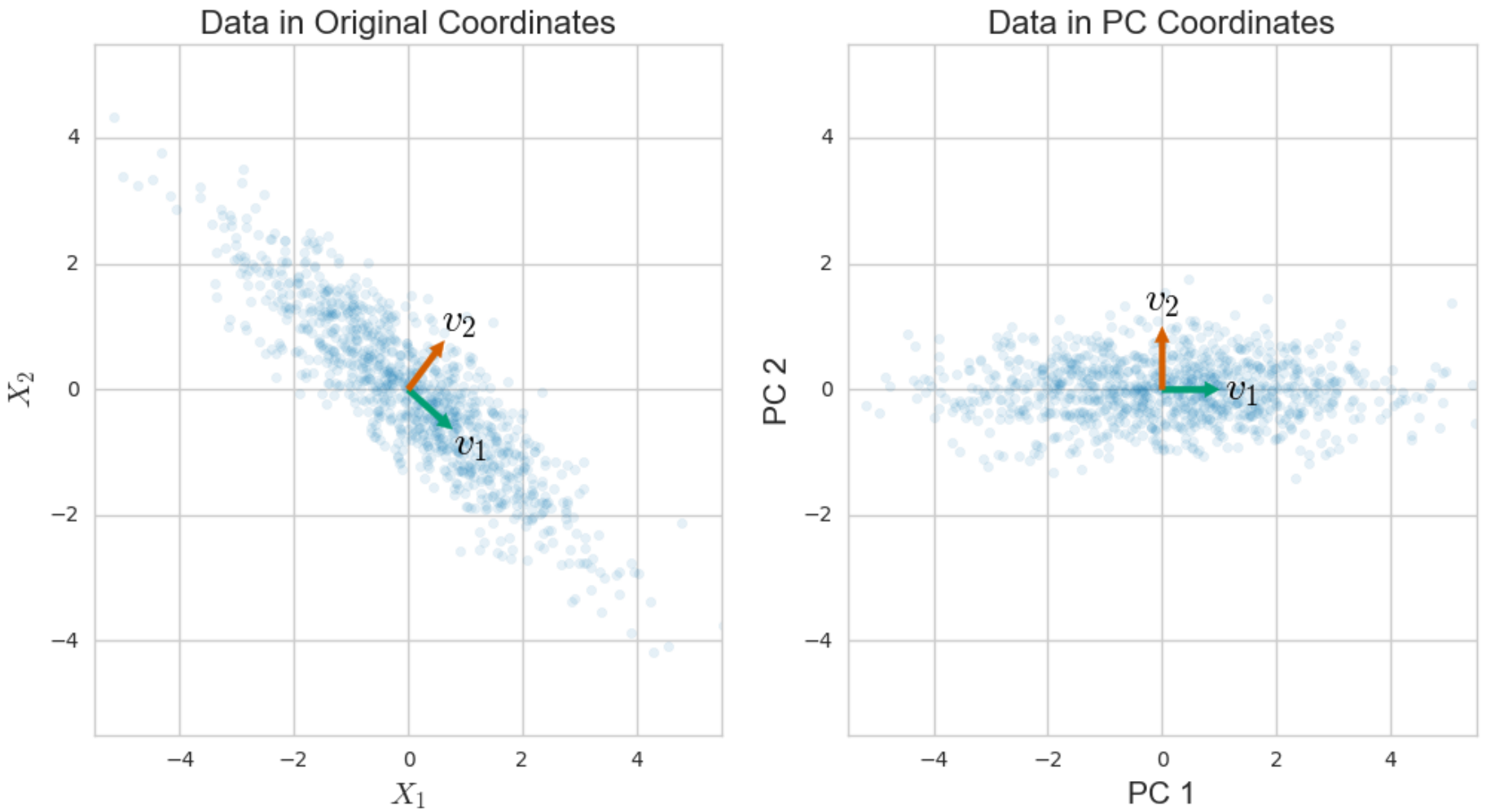
Applications of Matrix Decompositions for Machine Learning

Pavel Fakanov

What is PCA?


Data in Original Coordinates
X_2
X_1
v_2
v_1
Data in PC Coordinates
PC 2
PC 1
v_2
v_1

PCA
Step 1

- Mean normalize every feature of the dataset.

Student	Math	English	Art
1	90	60	90
2	90	90	30
3	60	60	60
4	60	60	90
5	30	30	30

$$\mathbf{A} = \begin{bmatrix} 90 & 60 & 90 \\ 90 & 90 & 30 \\ 60 & 60 & 60 \\ 60 & 60 & 90 \\ 30 & 30 & 30 \end{bmatrix}$$

$$\overline{\mathbf{A}} = [\,66 \quad 60 \quad 60\,]$$

PCA
Step 2

- Compute the covariance matrix of the dataset

	Math	*English*	*Art*
Math	504	360	180
English	360	360	0
Art	180	0	720

$$cov(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(Xi - \bar{x})(Yi - \bar{y})$$

PCA
Step 3

- Compute Eigenvectors and corresponding Eigenvalues

$$\det\begin{pmatrix} 504-\lambda & 360 & 180 \\ 360 & 360-\lambda & 0 \\ 180 & 0 & 720-\lambda \end{pmatrix}$$

$$\lambda \approx 44.81966\ldots, \lambda \approx 629.11039\ldots, \lambda \approx 910.06995\ldots$$

Eigenvalues

$$\begin{pmatrix} -3.75100\ldots \\ 4.28441\ldots \\ 1 \end{pmatrix}, \begin{pmatrix} -0.50494\ldots \\ -0.67548\ldots \\ 1 \end{pmatrix}, \begin{pmatrix} 1.05594\ldots \\ 0.69108\ldots \\ 1 \end{pmatrix}$$

Eigenvectors

PCA
Step 4

- Choose k eigenvectors with the largest eigenvalues to form a d × k dimensional matrix W.

$$\lambda \approx 44.81966\ldots, \lambda \approx 629.11039\ldots, \lambda \approx 910.06995\ldots$$

Eigenvalues

$$\mathbf{W} = \begin{bmatrix} 1.05594 & -0.50494 \\ 0.69108 & -0.67548 \\ 1 & 1 \end{bmatrix}$$

$$\begin{pmatrix} -3.75100\ldots \\ 4.28441\ldots \\ 1 \end{pmatrix}, \begin{pmatrix} -0.50494\ldots \\ -0.67548\ldots \\ 1 \end{pmatrix}, \begin{pmatrix} 1.05594\ldots \\ 0.69108\ldots \\ 1 \end{pmatrix}$$

Eigenvectors

PCA
Step 5

- Transform the samples onto the new subspace

$$A' = AW$$

```
[[ 24.   0.  30.]          [[ 55.34256  17.88144]
 [ 24.  30. -30.]           [ 16.07496 -62.38296]
 [ -6.   0.   0.]    →      [ -6.33564   3.02964]
 [ -6.   0.  30.]           [ 23.66436  33.02964]
 [-36. -30. -30.]]          [-88.74624   8.44224]]
```

SVD and PCA connection

$$XX^T = WDW^T$$

SVD and PCA connection

$$XX^T = WDW^T$$

$$X = U\Sigma V^T$$

SVD and PCA connection

$$XX^T = WDW^T$$

$$X = U\Sigma V^T$$

$$XX^T = (U\Sigma V^T)(V\Sigma U^T)$$

SVD and PCA connection

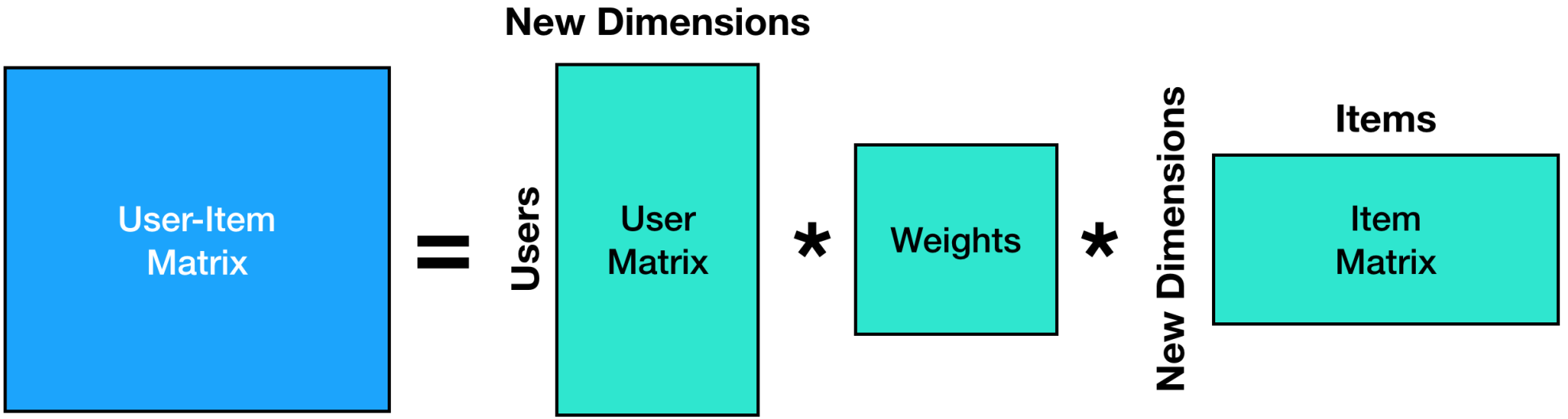
$$XX^T = WDW^T$$

$$X = U\Sigma V^T$$

$$XX^T = (U\Sigma V^T)(V\Sigma U^T)$$

$$XX^T = U\Sigma^2 U^T$$

SVD. Recommender systems



Problem Statement

- Predict a rating for a user item pair based on the history of ratings given by the user and given to the item

$$q_i \; - \; vector\; presentation\; of\; an\; item$$

$$p_u \; - \; vector\; presentation\; of\; a\; user$$

$$expected\; rating = \; \hat{r}_{ui} = q_i^T p_u$$

$$minimum(p, q) \sum_{(u,i)\epsilon K} \left(r_{ui} - q_i^T . p_u\right)^2$$

Modifying minimization equation

- Regularization:

$$minimum(p,q) \sum_{(u,i)\epsilon K} \left(r_{ui} - q_i^T . p_u\right)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2)$$

Modifying minimization equation

- Regularization:

$$minimum(p,q) \sum_{(u,i)\epsilon K} \left(r_{ui} - q_i^T . p_u\right)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2)$$

Bias terms:

$$\hat{r}_{ui} = q_i^T . p_u + \mu + b_i + b_u$$

Modifying minimization equation

- Regularization:

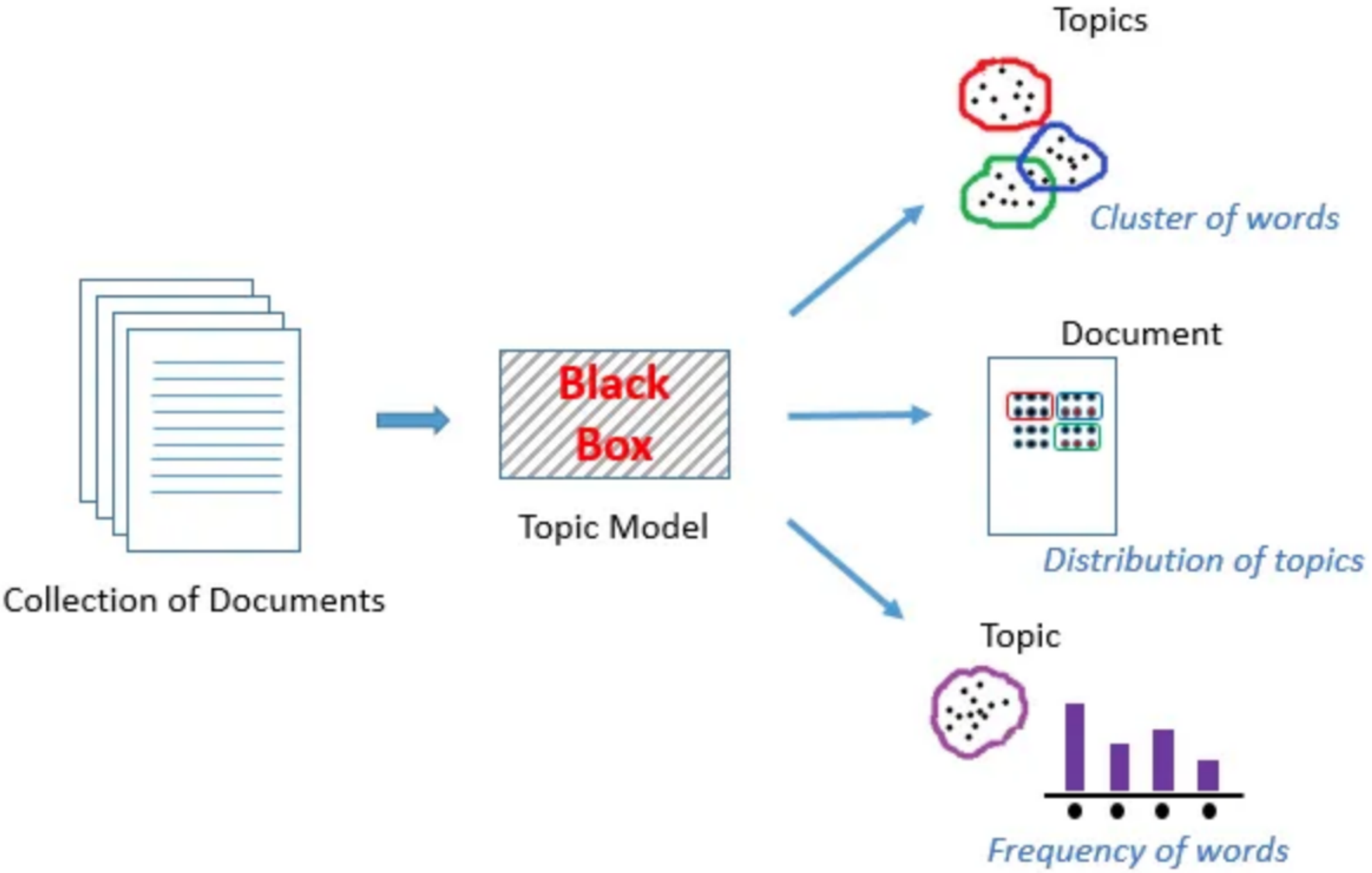
$$minimum(p,q) \sum_{(u,i)\epsilon K} \left(r_{ui} - q_i^T . p_u\right)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2)$$

Bias terms:

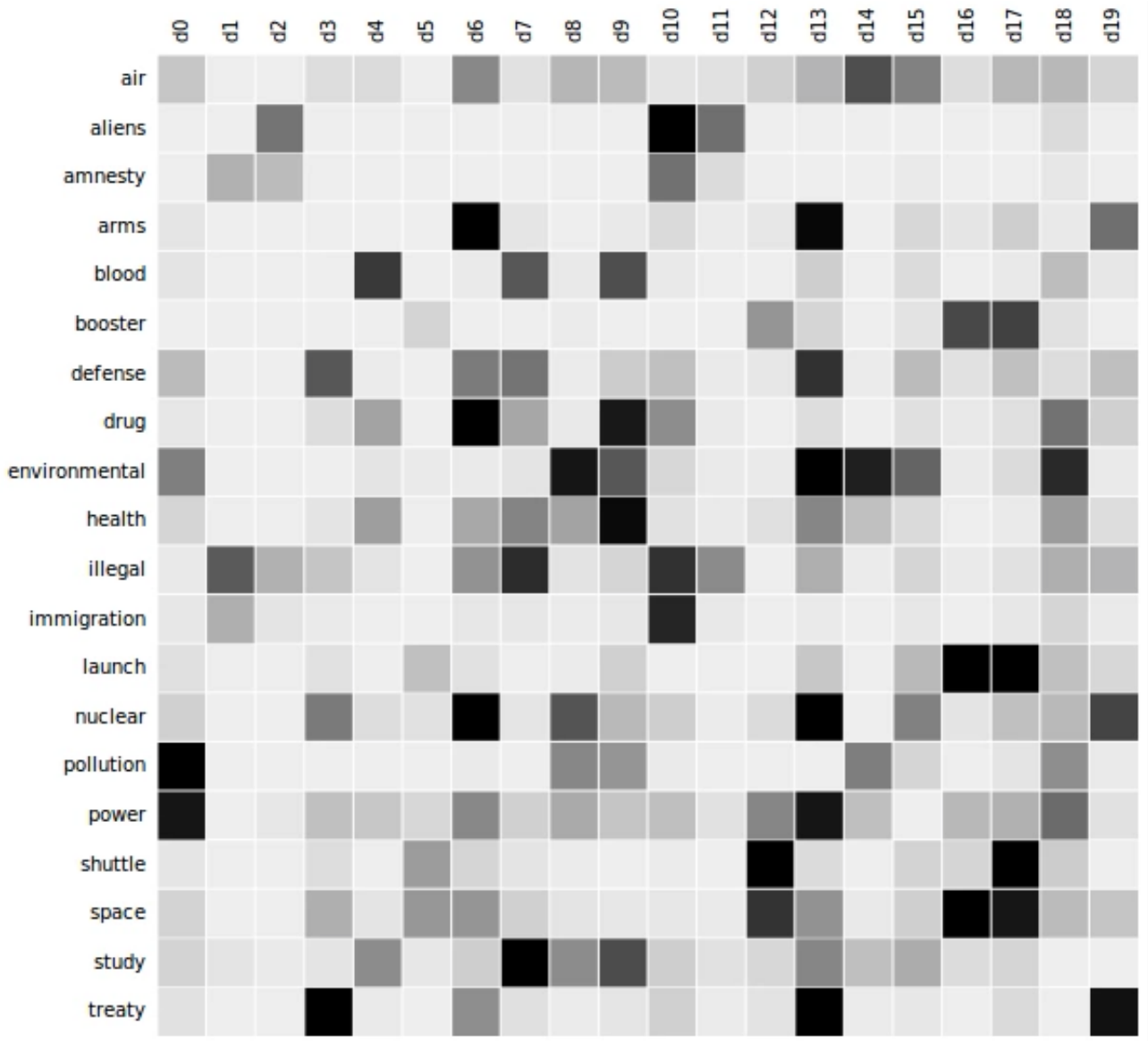
$$\hat{r}_{ui} = q_i^T . p_u + \mu + b_i + b_u$$

$$minimum(p,q,b_i,b_u) \sum_{(u,i)\epsilon K} \left(r_{ui} - q_i^T . p_u - \mu - b_i - b_u\right)^2 + \lambda\left(\|q_i\|^2 + \|p_u\|^2 + b_i^2 + b_u^2\right)$$

Topic modeling



Basic approach.
Document-word matrix



SVD approach
Step 1

Generate a document-term matrix of shape m x n having TF-IDF scores.

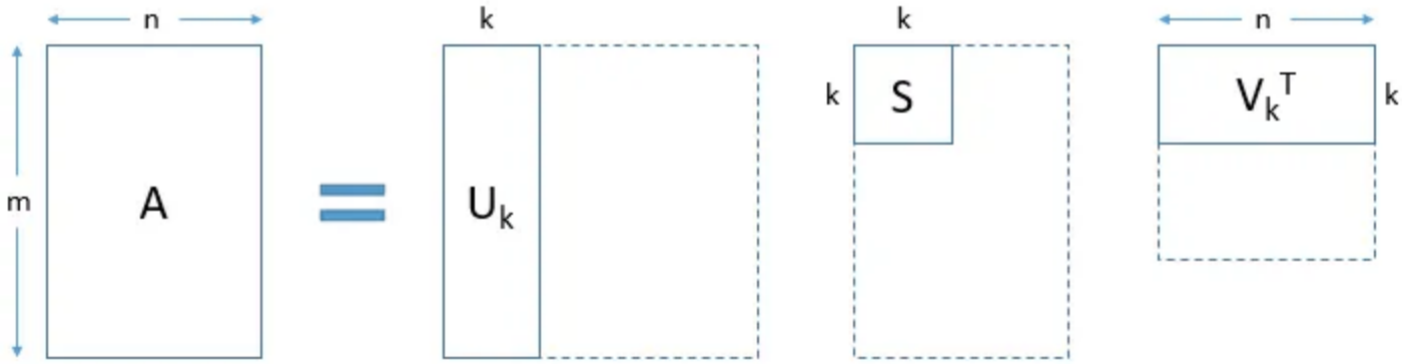
Terms

Documents

	T1	T2	T3	...	Tn
D1	0.2	0.1	0.5	...	0.1
D2	0.1	0.3	0.4	...	0.3
D3	0.3	0.1	0.1	...	0.5
...	...	...	...	...	...
Dm	0.2	0.1	0.2	...	0.1

SVD approach.
Step 2

$$A = USV^T$$



- K - the number of desired topics
- U row - vector representation of a document
- V column – vector representation of a word

SVD Approach.
Step 3

- Find similar words and similar documents using the cosine similarity method.

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$

Background removal

Original image

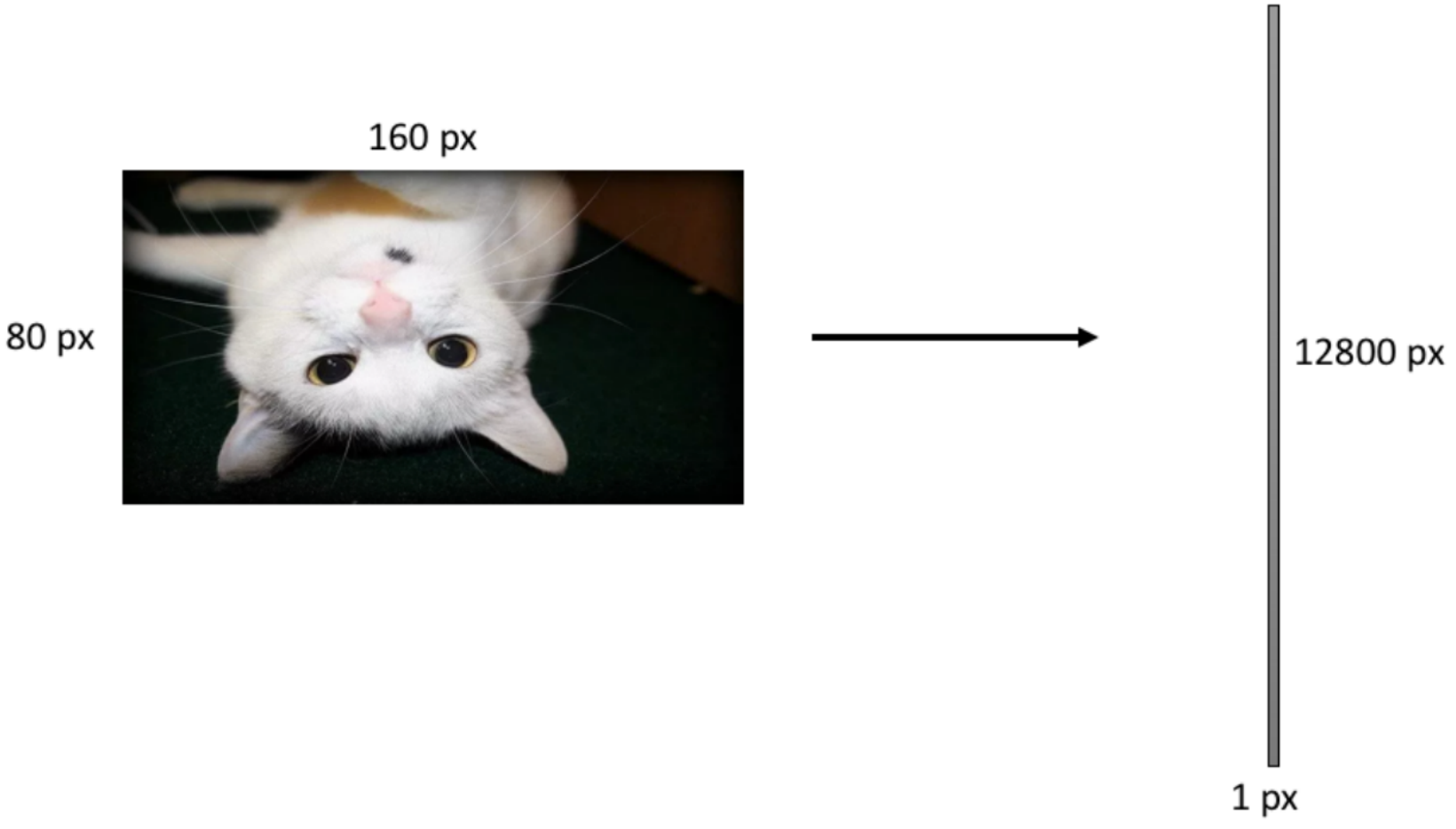
Image without background

Background removal
Step 1

- Split video into frames based on some predefined frame per second

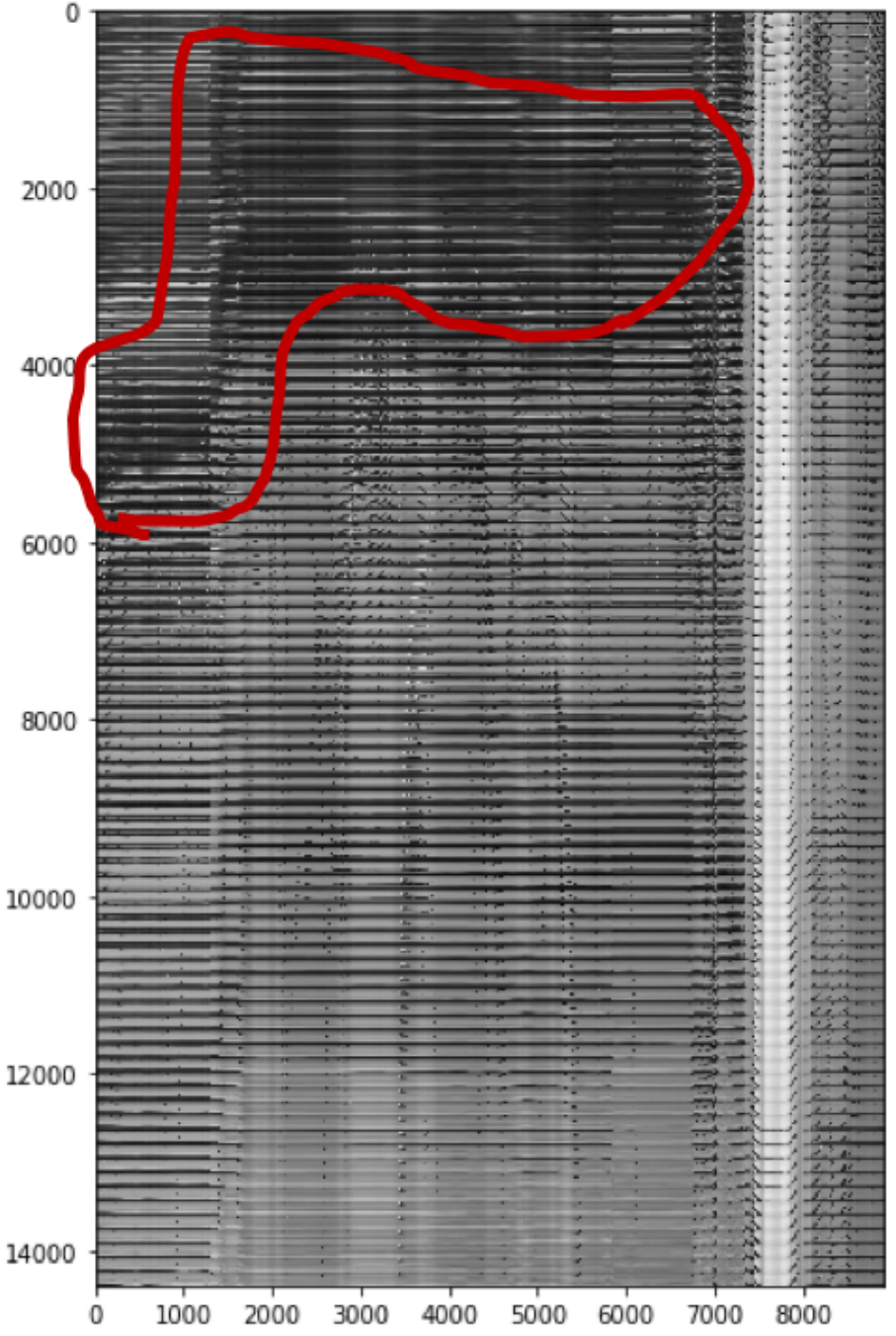
Background removal
Step 2

- Flatten each frame (convert to 1-D vector).
  Matrix of video with images as 1-D vector



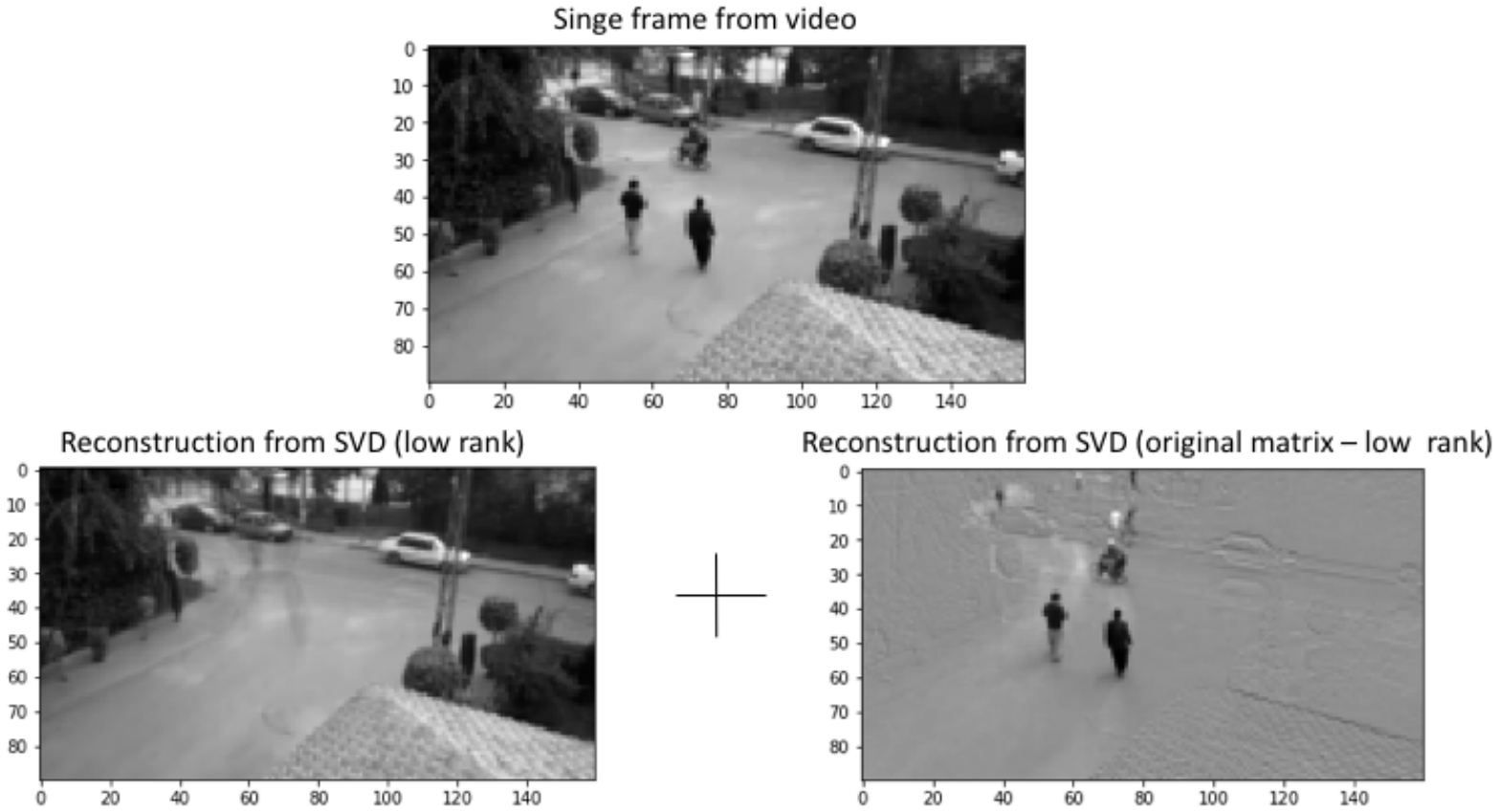
Background removal Step 3

- Do step 2 for all the frames and concatenate resultant 1-D vectors side by side



Background removal
Step 4

- Decompose the matrix from step 3 using SVD.The foreground is the difference between the original matrix and the lower rank matrix



Cholvesky decomposition

$$\mathbf{A} = \mathbf{L}\mathbf{L}^*$$

- L is a lower triangular matrix with real and positive diagonal entries
- L* denotes the Hermitian transpose of L
- The Cholesky decomposition is unique when A is positive definite

The Cholesky–Banachiewicz algorithm

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T = \begin{pmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{pmatrix} \begin{pmatrix} L_{11} & L_{21} & L_{31} \\ 0 & L_{22} & L_{32} \\ 0 & 0 & L_{33} \end{pmatrix}$$

$$= \begin{pmatrix} L_{11}^2 & & (\text{symmetric}) \\ L_{21}L_{11} & L_{21}^2 + L_{22}^2 & \\ L_{31}L_{11} & L_{31}L_{21} + L_{32}L_{22} & L_{31}^2 + L_{32}^2 + L_{33}^2 \end{pmatrix}$$

The Cholesky–Banachiewicz algorithm

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T = \begin{pmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{pmatrix} \begin{pmatrix} L_{11} & L_{21} & L_{31} \\ 0 & L_{22} & L_{32} \\ 0 & 0 & L_{33} \end{pmatrix}$$
$$= \begin{pmatrix} L_{11}^2 & & \text{(symmetric)} \\ L_{21}L_{11} & L_{21}^2 + L_{22}^2 & \\ L_{31}L_{11} & L_{31}L_{21} + L_{32}L_{22} & L_{31}^2 + L_{32}^2 + L_{33}^2 \end{pmatrix}$$

$$\mathbf{L} = \begin{pmatrix} \sqrt{A_{11}} & 0 & 0 \\ A_{21}/L_{11} & \sqrt{A_{22} - L_{21}^2} & 0 \\ A_{31}/L_{11} & \left(A_{32} - L_{31}L_{21}\right)/L_{22} & \sqrt{A_{33} - L_{31}^2 - L_{32}^2} \end{pmatrix}$$

The Cholesky–Banachiewicz algorithm

$$L_{j,j} = \sqrt{A_{j,j} - \sum_{k=1}^{j-1} L_{j,k}^2},$$

$$L_{i,j} = \frac{1}{L_{j,j}} \left(A_{i,j} - \sum_{k=1}^{j-1} L_{i,k} L_{j,k} \right) \quad \text{for } i > j.$$

The expression under the square root is always positive if A is real and positive-definite

System of linear equations

$$\begin{pmatrix} A_{11} & A^*_{21} & A^*_{31} & A^*_{41} \\ A_{21} & A_{22} & A^*_{32} & A^*_{42} \\ A_{31} & A_{32} & A_{33} & A^*_{43} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{pmatrix} * \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}$$

System of linear equations

$$\begin{pmatrix} L_{11} & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} \end{pmatrix} * \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}$$

$$\begin{pmatrix} L_{11} & L^*_{21} & L^*_{31} & L^*_{41} \\ 0 & L_{22} & L^*_{32} & L^*_{42} \\ 0 & 0 & L_{33} & L^*_{43} \\ 0 & 0 & 0 & L_{44} \end{pmatrix} * \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}$$

Least-squares regression

$$\hat{\beta} \quad = \quad (X^T X)^{-1} X^T Y$$

Least-squares regression

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

$$X^T X = L L^T$$

Least-squares regression

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

$$X^T X = L L^T$$

$$L L^T \hat{\beta} = X^T Y$$

Least-squares regression

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

$$X^T X = L L^T$$

$$L L^T \hat{\beta} = X^T Y$$

1. $Solve\ for\ x = L^T \beta$

Least-squares regression

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

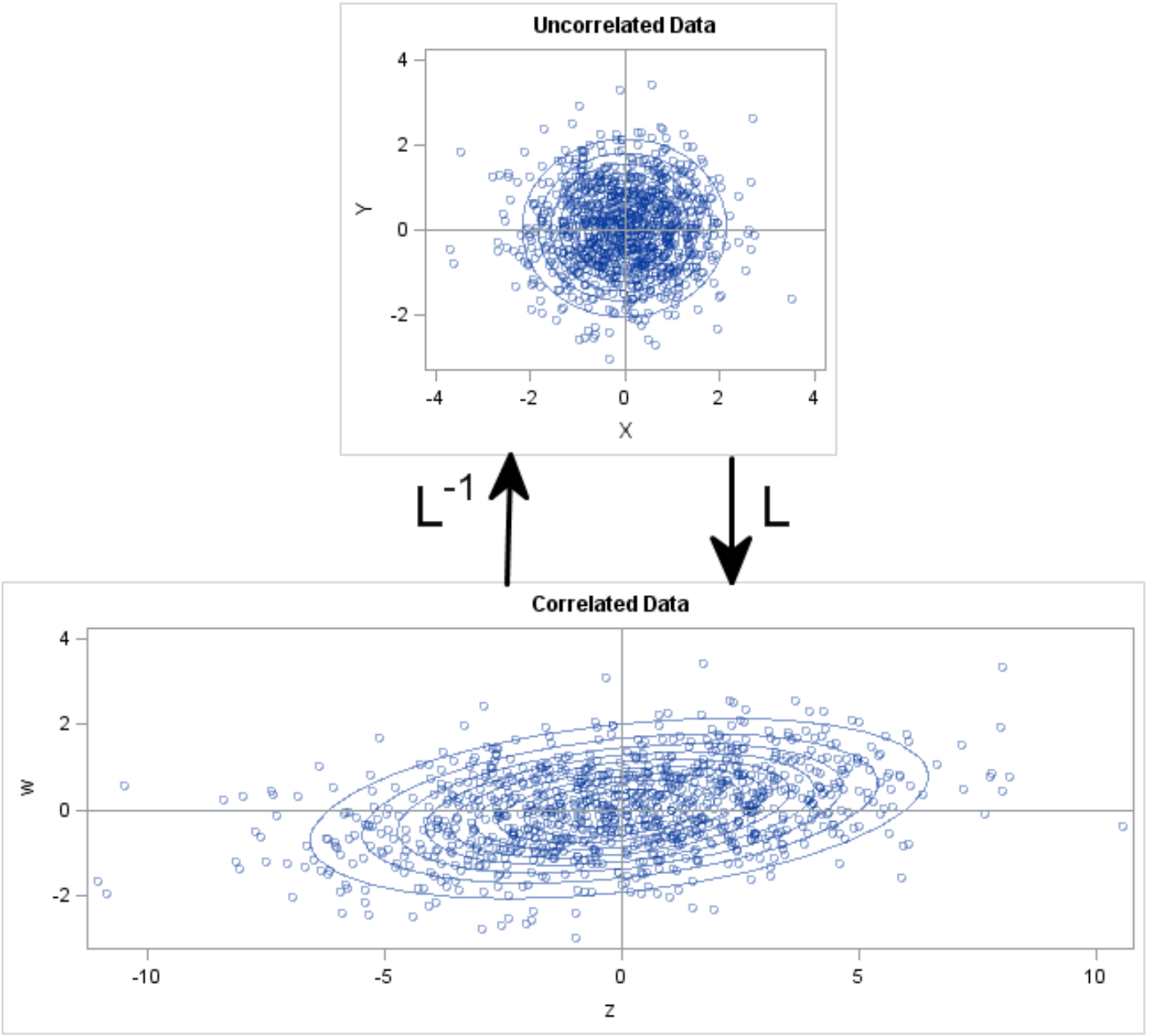
$$X^T X = L L^T$$

$$L L^T \hat{\beta} = X^T Y$$

1. $Solve\ for\ x = L^T \beta$

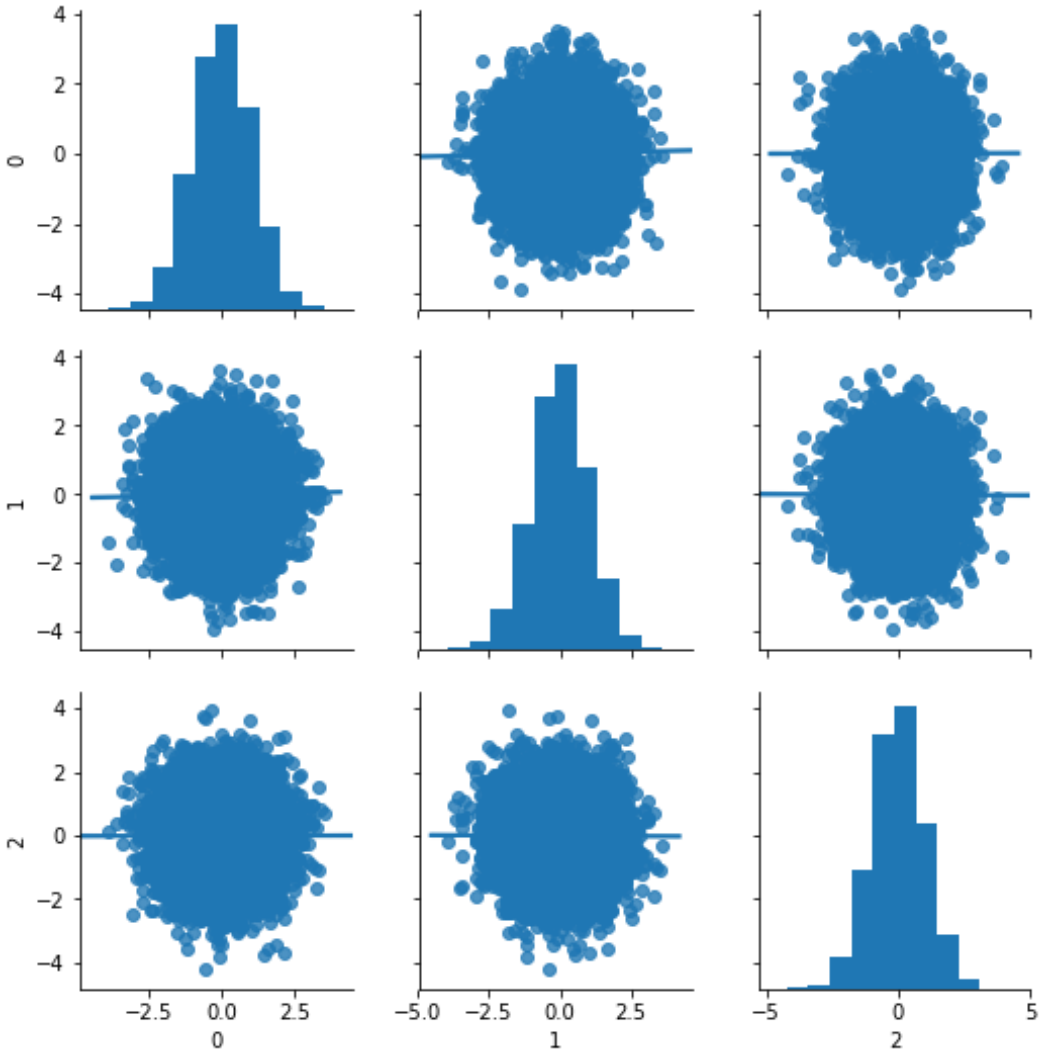
2. $Solve\ for\ \beta$

Monte-Carlo Simulation



Monte-Carlo Simulation.
Step 1

- Generate uncorrelated normal variables:



Monte-Carlo Simulation. Step 2

- Calculate covariance matrix

$$\text{Covariance Matrix} = \begin{bmatrix} 10.0 & -2.0 & 2.0 \\ -2.0 & 20.0 & 0.5 \\ 2.0 & 0.5 & 0.5 \end{bmatrix}$$

Monte-Carlo Simulation. Step 3

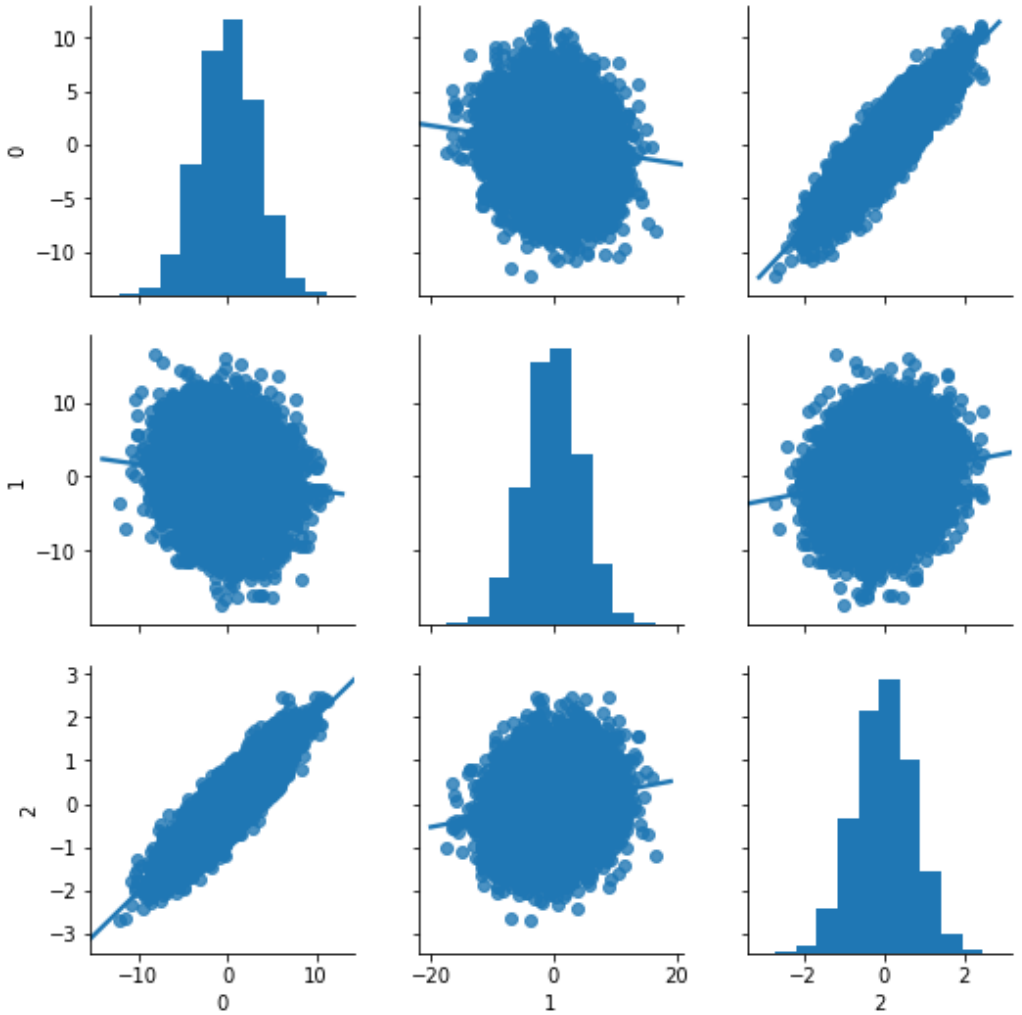
- Apply Cholesky Decomposition to covariance matrix

```
[10.   -2.    2. ]
[-2.   20.    0.5]
[ 2.    0.5   0.5]
```

```
[ 3.162  0.     0.   ]
[-0.632  4.427  0.   ]
[ 0.632  0.203  0.242]
```

Monte-Carlo Simulation. Step 4

- Multiply the matrix of uncorrelated random variables and L matrix



Non-negative matrix factorization

$$V \simeq WH$$

$$W, H = \underset{W,H}{argmin} \|V - WH\|_F$$

$$W \geq 0, \; H \geq 0$$

The optimization methods mentioned earlier are not suitable for that problem

Problems with finding Non-negative matrix factorization

$$(W^*, H^*) = \underset{W \geq 0,\, H \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2 .$$

GD can’t be applied since there is limitation

Problems with finding Non-negative matrix factorization

$$(W^*, H^*) = \underset{W \geq 0,\, H \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2 .$$

GD can't be applied since there is limitation

Idea! $h_{kj} \leftarrow h_{kj} - \nu_{kj} \frac{\partial D_F}{\partial h_{kj}}, w_{ik} \leftarrow w_{ik} - \eta_{ik} \frac{\partial D_F}{\partial w_{ik}}. k = 1, \ldots, r, ; \quad i = 1, \ldots, m, ; \quad j = 1, \ldots, n$

Problems with finding Non-negative matrix factorization

$$(W^*, H^*) = \underset{W \geq 0,\, H \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2 .$$

GD can't be applied since there is limitation

Idea! $h_{kj} \leftarrow h_{kj} - \nu_{kj} \frac{\partial D_F}{\partial h_{kj}}, w_{ik} \leftarrow w_{ik} - \eta_{ik} \frac{\partial D_F}{\partial w_{ik}}. k = 1, \ldots, r, ; \quad i = 1, \ldots, m, ; \quad j = 1, \ldots, n$

Problem: The limitations are not fulfilled

Deriving Multiplicative Update Rules for NMF

$$\frac{\partial D_F}{\partial h_{kj}} = \sum_{i=1}^{m} w_{ik}\hat{x}_{ij} - \sum_{i=1}^{m} w_{ik}x_{ij},$$

$$D_F\left(X, \hat{X}\right) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left(x_{ij} - \hat{x}_{ij}\right)^2 \equiv \|X - WH\|_F^2$$

Deriving Multiplicative Update Rules for NMF

$$\frac{\partial D_F}{\partial h_{kj}} = \sum_{i=1}^{m} w_{ik}\hat{x}_{ij} - \sum_{i=1}^{m} w_{ik}x_{ij},$$

$$\nu_{kj} = \frac{h_{kj}}{\sum_{i=1}^{m} w_{ik}\hat{x}_{ij}}$$

$$D_F\left(X, \hat{X}\right) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left(x_{ij} - \hat{x}_{ij}\right)^2 \equiv \|X - WH\|_F^2$$

Deriving Multiplicative Update Rules for NMF

$$\frac{\partial D_F}{\partial h_{kj}} = \sum_{i=1}^{m} w_{ik}\hat{x}_{ij} - \sum_{i=1}^{m} w_{ik}x_{ij},$$

$$\nu_{kj} = \frac{h_{kj}}{\sum_{i=1}^{m} w_{ik}\hat{x}_{ij}}$$

$$h_{kj} \leftarrow h_{kj} - \frac{h_{kj}}{\sum_{i=1}^{m} w_{ik}\hat{x}_{ij}} \left(\sum_{i=1}^{m} w_{ik}\hat{x}_{ij} - \sum_{i=1}^{m} w_{ik}x_{ij} \right)$$

$$D_F\left(X, \hat{X}\right) = \sum_{i=1}^{m}\sum_{j=1}^{n} \left(x_{ij} - \hat{x}_{ij}\right)^2 \equiv \|X - WH\|_F^2$$

Deriving Multiplicative Update Rules for NMF

$$\frac{\partial D_F}{\partial h_{kj}} = \sum_{i=1}^{m} w_{ik}\hat{x}_{ij} - \sum_{i=1}^{m} w_{ik}x_{ij},$$

$$\nu_{kj} = \frac{h_{kj}}{\sum_{i=1}^{m} w_{ik}\hat{x}_{ij}}$$

$$h_{kj} \leftarrow h_{kj} - \frac{h_{kj}}{\sum_{i=1}^{m} w_{ik}\hat{x}_{ij}} \left(\sum_{i=1}^{m} w_{ik}\hat{x}_{ij} - \sum_{i=1}^{m} w_{ik}x_{ij} \right)$$

$$h_{kj} \leftarrow h_{kj} \frac{\sum_{i=1}^{m} w_{ik}x_{ij}}{\sum_{i=1}^{m} w_{ik}\hat{x}_{ij}}.$$

If the origin matrices are properly intialized, then they will remain positive

Other methods of optimization

ALS:

$$H \leftarrow \max\left(\underset{H}{\operatorname{argmin}} \|X - WH\|_F^2, 0\right) = \\ = \max\left(\left(W^T W\right)^{-1} W^T X, 0\right)$$

Other methods of optimization

ALS:

$$H \leftarrow \max\left(\underset{H}{\operatorname{argmin}} \|X - WH\|_F^2, 0\right) = \\ = \max\left(\left(W^T W\right)^{-1} W^T X, 0\right)$$

ANLS:

$$H \leftarrow \underset{H \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2$$

Other methods of optimization

ALS:

$$H \leftarrow \max\left(\underset{H}{\operatorname{argmin}} \|X - WH\|_F^2, 0\right) = \; = \max\left(\left(W^T W\right)^{-1} W^T X, 0\right)$$

ANLS:

$$H \leftarrow \underset{H \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2$$

HALS:

$$h_k \leftarrow \underset{h_k \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2 = \max\left(0, \frac{w_k^T X - \sum_{l \neq k} w_k^T w_l h_l}{w_k^T w_k}\right)$$

NMF
Loss functions

- L1-norm $d_1\left(x, \hat{x}\right) = \left|x - \hat{x}\right|$

NMF
Loss functions

- L1-norm $d_1(x, \hat{x}) = |x - \hat{x}|$

- Frobenius norm $d_F(x, \hat{x}) = (x - \hat{x})^2$

NMF
Loss functions

- L1-norm $d_1\left(x, \hat{x}\right) = \left|x - \hat{x}\right|$
- Frobenius norm $d_F\left(x, \hat{x}\right) = \left(x - \hat{x}\right)^2$
- Kullback–Leibler divergence $d_{KL}\left(x, \hat{x}\right) = x \ln \frac{x}{\hat{x}} - x + \hat{x}$

NMF
Loss functions

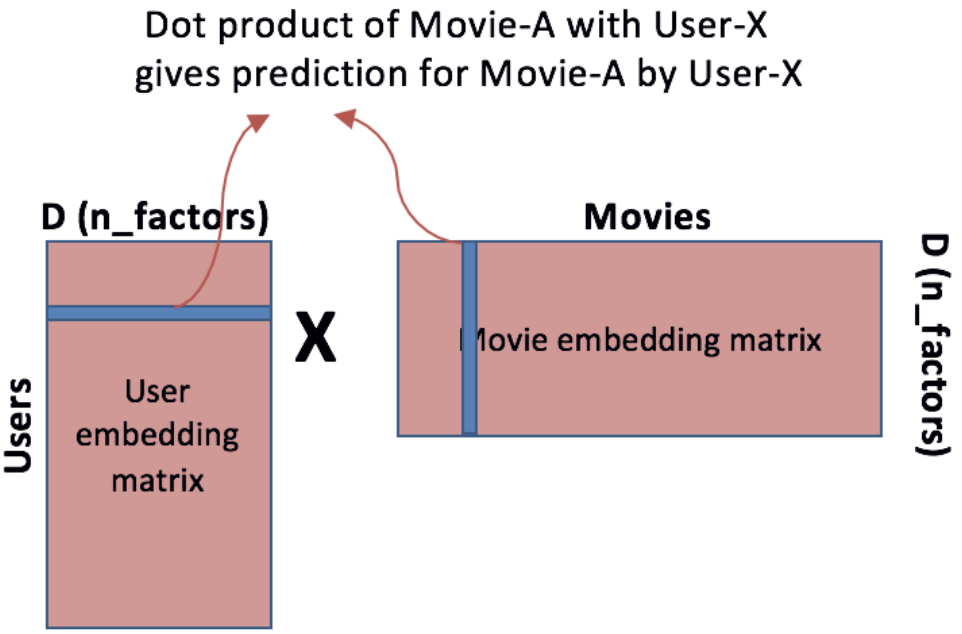
- L1-norm $d_1\left(x, \hat{x}\right) = \left|x - \hat{x}\right|$
- Frobenius norm $d_F\left(x, \hat{x}\right) = \left(x - \hat{x}\right)^2$
- Kullback–Leibler divergence $d_{KL}\left(x, \hat{x}\right) = x \ln \frac{x}{\hat{x}} - x + \hat{x}$
- Hellinger distance $d_H\left(x, \hat{x}\right) = \left(\sqrt{x} - \sqrt{\hat{x}}\right)^2$

NMF
Recommender systems

Movies

Users	*(Sparse matrix)*	Waking Life	Boyhood	Before Sunset
	Jesse	4.5		4.0
	Celine		3.5	5.0

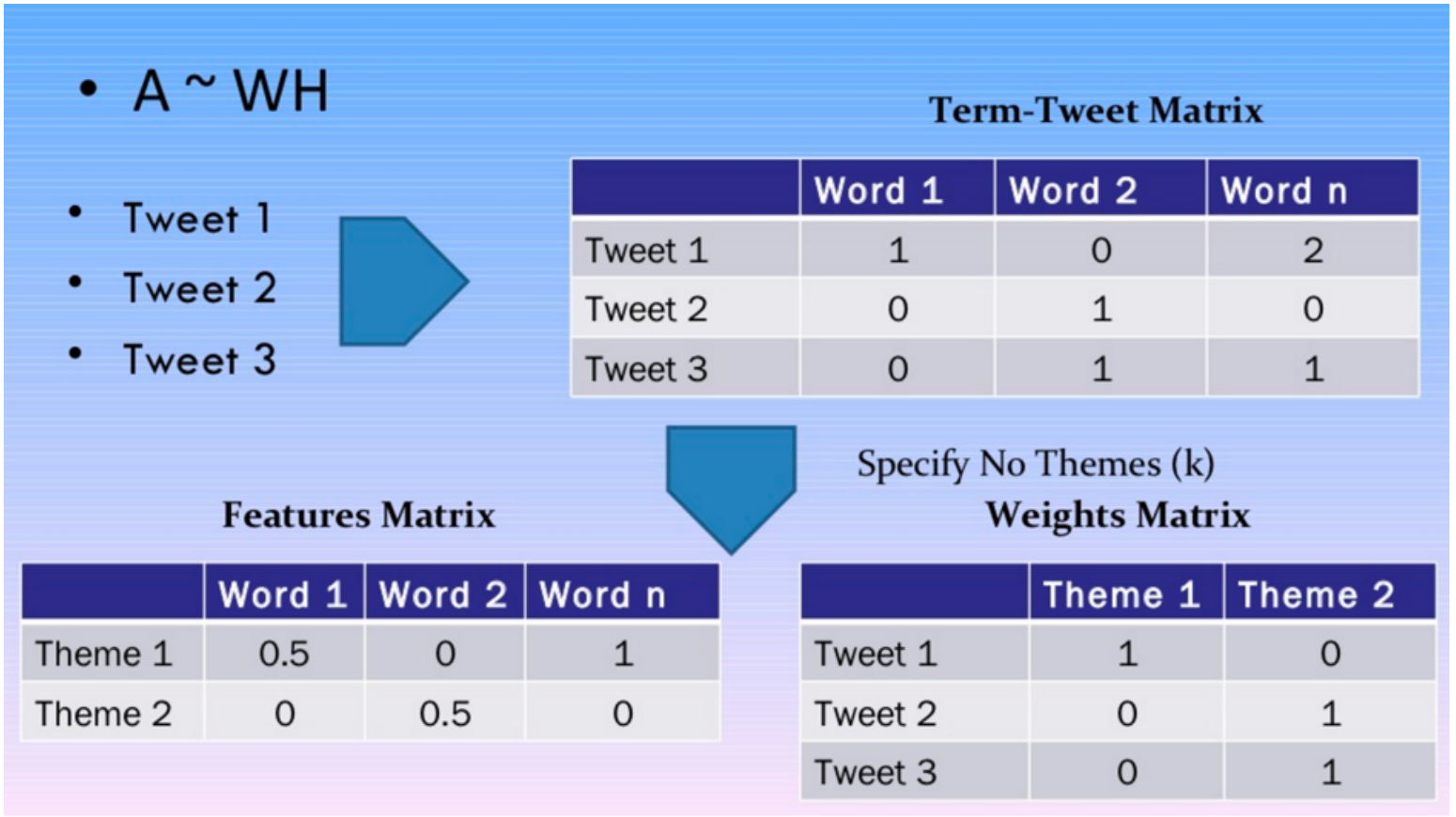
≈



The key advantage of the approach is the interpretability of results

NMF
Topic modeling

- Bag of words text presentation
- Tf-idf text presentation
- NMF



NMF
Topic modeling results

NMF Topics:

Topic 0: people don think like know time right good did say

Topic 1: windows file use dos files window using program problem card

Topic 2: god jesus bible christ faith believe christian christians church sin

Topic 3: drive scsi drives hard disk ide controller floppy cd mac

Topic 4: game team year games season players play hockey win player

Topic 5: key chip encryption clipper keys government escrow public use algorithm

Topic 6: thanks does know mail advance hi anybody info looking help

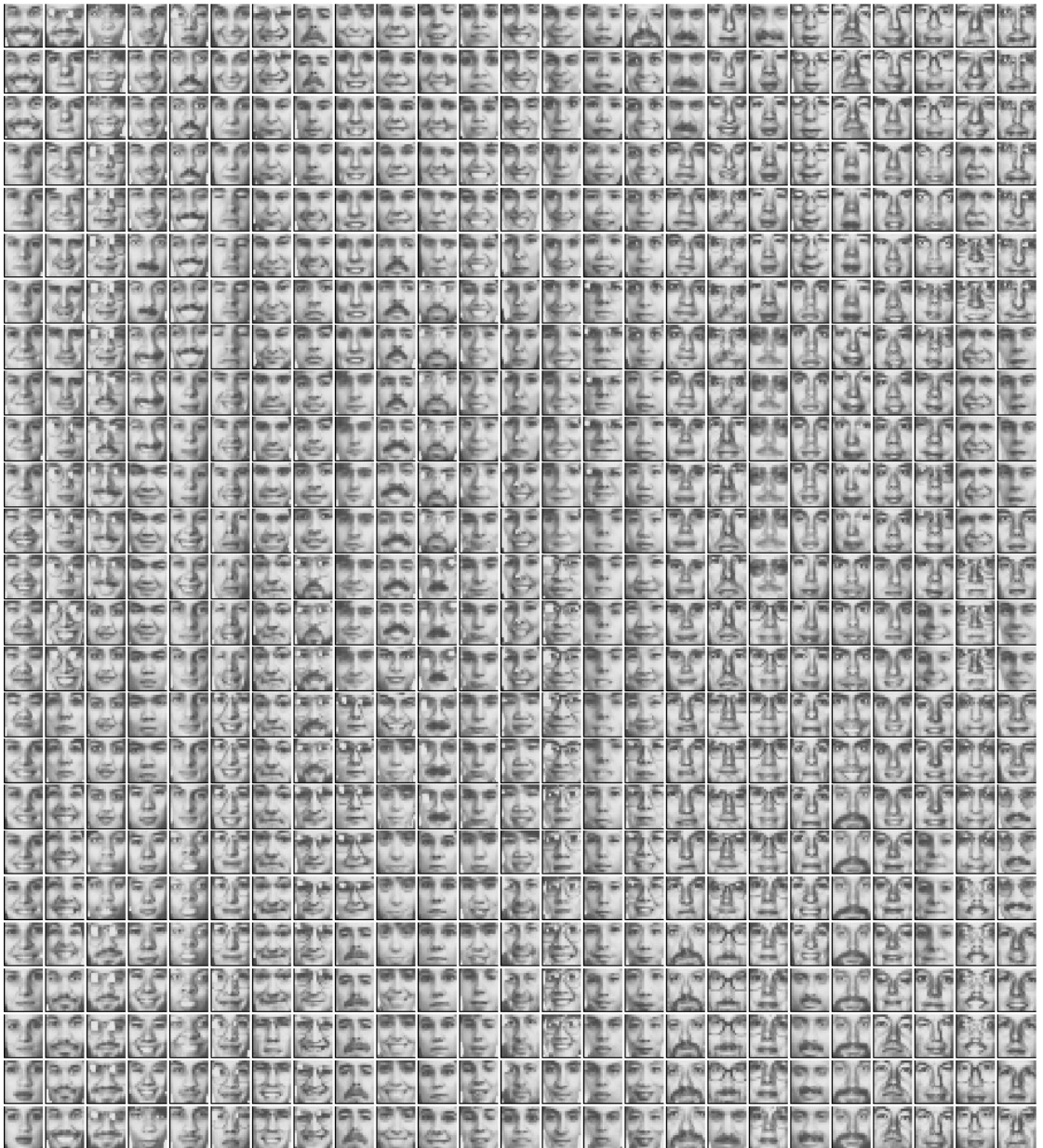
Topic 7: car new 00 sale price 10 offer condition shipping 20

Topic 8: just like don thought ll got oh tell mean fine

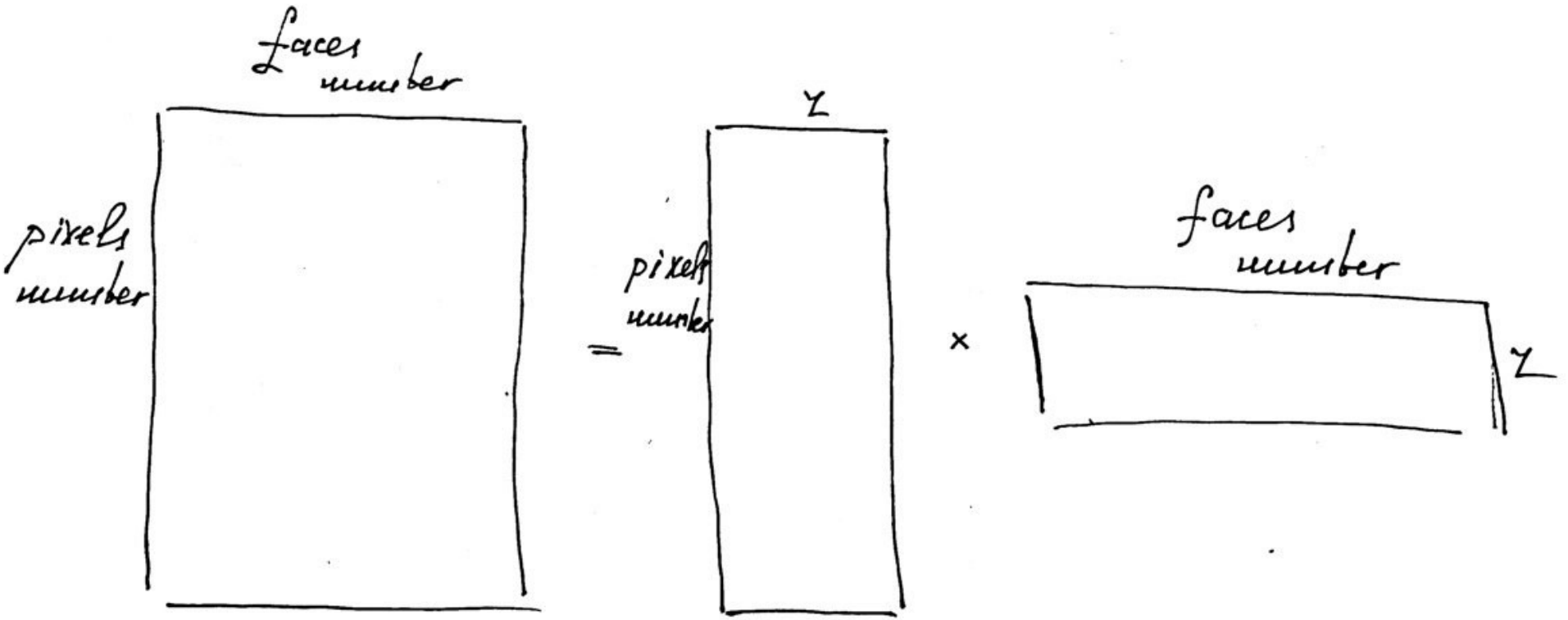
Topic 9: edu soon cs university com email internet article ftp send

20 Newsgoups dataset

NMF
Decomposing face images

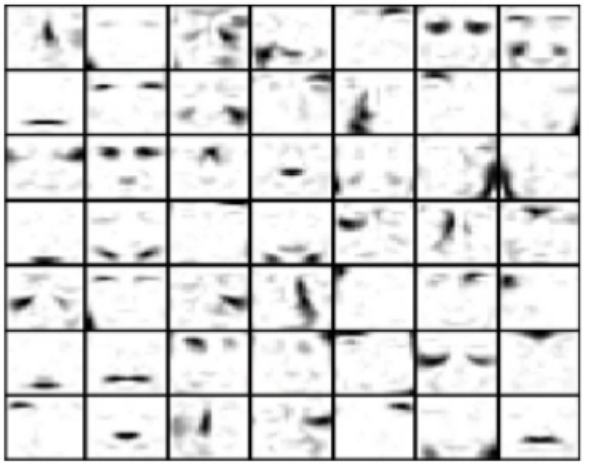


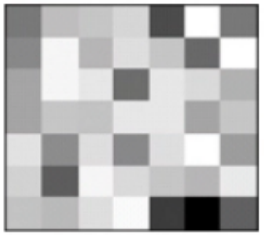
NMF
Decomposing face images



NMF
Decomposing face images

$$\underbrace{X(:,j)}_{j\text{th facial image}} \approx \sum_{k=1}^{r} \underbrace{W(:,k)}_{\text{facial features}} \underbrace{H(k,j)}_{\text{importance of features in } j\text{th image}} = \underbrace{WH(:,j)}_{\text{approximation of } j\text{th image}} .$$





Контрольные вопросы

- Как связаны между собой SVD и PCA? Как связаны собственные значения матрицы ковариации и сингулярные значения исходной матрицы?
- Укажите асимптотику нахождения разложения Холецкого
- Что такое Non-negative matrix factorization и как его получить? Опишите смысл применения NMF на примере разложения матриц из изображений лиц: что из себя представляют столбец матрицы признаков?