Data Representation as Low Rank Matrix Factorization

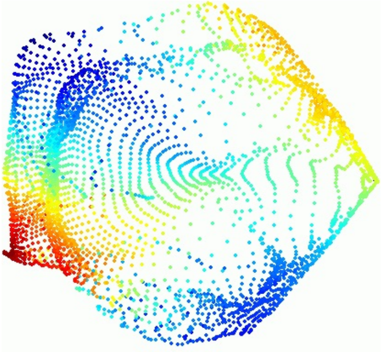
Ziv Epstein
ziv.epstein@pomona.edu

Pomona College

February 13, 2017

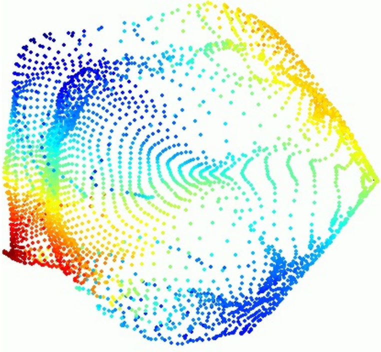
Points in $\mathbb{R}^m$

In many contexts in data science and linear algebra, we have lots of points in a high-dimensional space.



Denote these $x_i \in \mathbb{R}^m$ for $i = 1 \ldots n$.

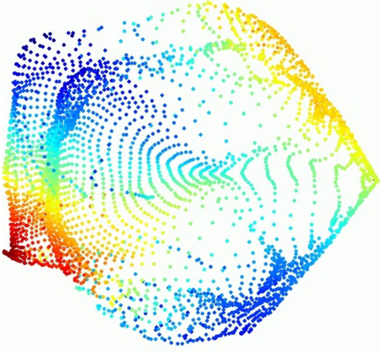
How do we cluster x_i?



Denote these $x_i \in \mathbb{R}^m$ for $i = 1 \ldots n$.

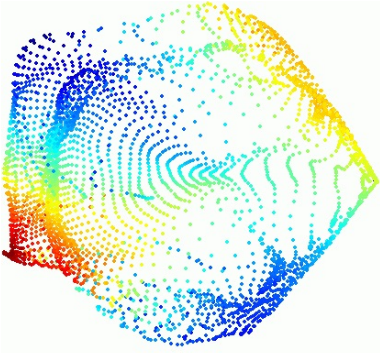
Points in $\mathbb{R}^m$

How do we cluster x_i? How do we perform dimensionality reduction?



Denote these $x_i \in \mathbb{R}^m$ for $i = 1 \ldots n$.

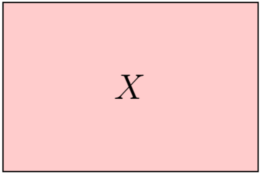
How do we cluster x_i? How do we perform dimensionality reduction? How do we visualize them?



Denote these $x_i \in \mathbb{R}^m$ for $i = 1 \ldots n$.

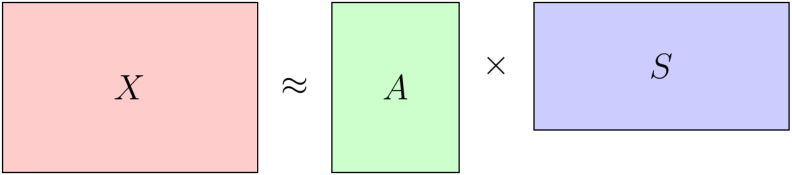
Representation and Factorization

We can *approximate* $X \in \mathbb{R}^{n \times m}$.



with $\mathrm{rank}(X) = \min(m, n)$.

We can *approximate* $X \in \mathbb{R}^{n \times m}$.



with $\text{rank}(AS) = k << min(m, n)$.

Low Dimensional Interpretation

Our original high-dimensional points are linear combinations of "basis elements'.'

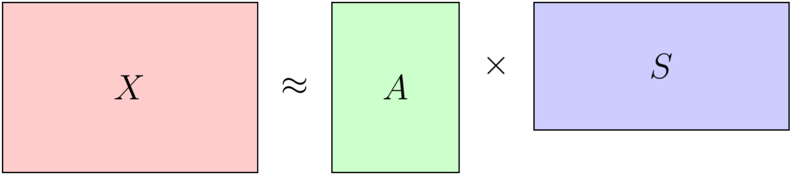
$$X_i = \sum_{j=1}^{k} A_{i,j} S_j$$

.

Low Dimensional Interpretation

Our original high-dimensional points are linear combinations of "basis elements'.'

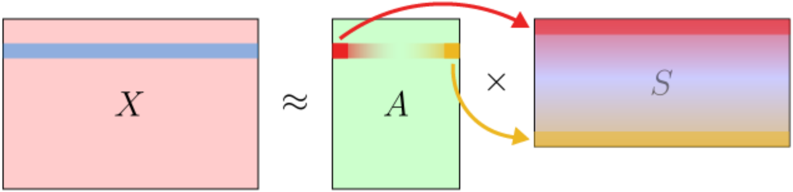
$$X_i = \sum_{j=1}^{k} A_{i,j} S_j$$



Low Dimensional Interpretation

Our original high-dimensional points are linear combinations of “basis elements’.’

$$X_i = \sum_{j=1}^{k} A_{i,j} S_j$$

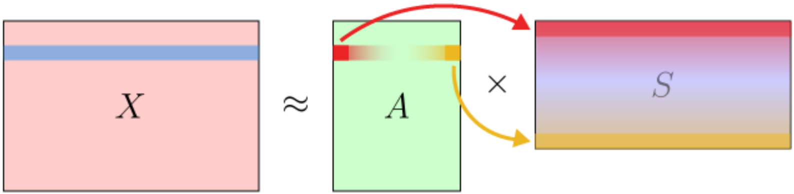


.

Low Dimensional Interpretation

Our original high-dimensional points are linear combinations of "basis elements'.'

$$X_i = \sum_{j=1}^{k} A_{i,j} S_j$$



Find factorization by solving optimization problem

$$\min_{A,S} ||X - AS||$$

.

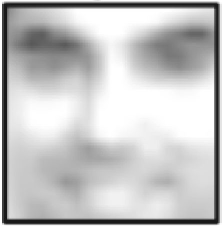
Example (Lee and Seung 1999)

Example (Lee and Seung 1999)

A 19x19 pixel greyscale image of a face is a data point ($x_i \in \mathbb{R}^{361}$ with values between 0 and 256)

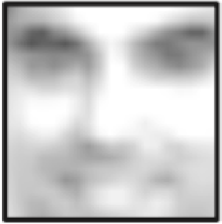
Example (Lee and Seung 1999)

A 19x19 pixel greyscale image of a face is a data point ($x_i \in \mathbb{R}^{361}$ with values between 0 and 256)



Example (Lee and Seung 1999)

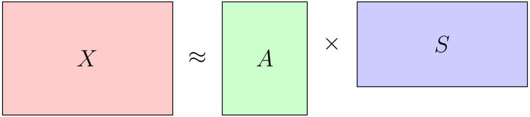
A 19x19 pixel greyscale image of a face is a data point ($x_i \in \mathbb{R}^{361}$ with values between 0 and 256)



Take a database of 2,429 faces to get $X \in \mathbb{R}^{2429 \times 361}$

Example (Lee and Seung 1999)

Before: Find any A and S such that $||X - AS||$ is minimized.



Example (Lee and Seung 1999)

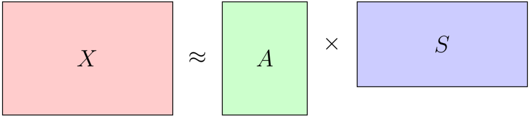
Before: Find any A and S such that $||X - AS||$ is minimized.

$$X \approx A \times S$$

Now: We add some constraints

Example (Lee and Seung 1999)

Before: Find any A and S such that $||X - AS||$ is minimized.

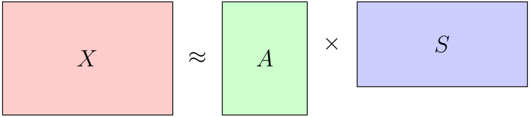


Now: We add some constraints

1. Columns of A to be orthonormal; Rows of S to be orthogonal (PCA)

Example (Lee and Seung 1999)

Before: Find any A and S such that $||X - AS||$ is minimized.

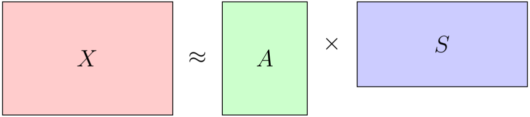


Now: We add some constraints

1. Columns of A to be orthonormal; Rows of S to be orthogonal (PCA)
2. Row of A sums to 1, which one entry equal to one and the rest equal to zero (VQ)

Example (Lee and Seung 1999)

Before: Find any A and S such that $||X - AS||$ is minimized.

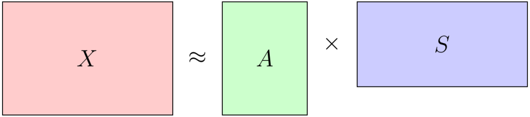


Now: We add some constraints

1. Columns of A to be orthonormal; Rows of S to be orthogonal (PCA)
2. Row of A sums to 1, which one entry equal to one and the rest equal to zero (VQ)
3. All entries of A and S are non-negative (NNMF)

Example (Lee and Seung 1999)

Before: Find any A and S such that $||X - AS||$ is minimized.



Now: We add some constraints

1. Columns of A to be orthonormal; Rows of S to be orthogonal (PCA)
2. Row of A sums to 1, which one entry equal to one and the rest equal to zero (VQ)
3. All entries of A and S are non-negative (NNMF)

For each, find A and S subject to constraints that minimize

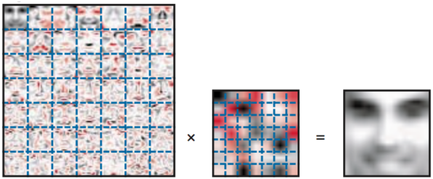
$$||X - AS||$$

Example (Lee and Seung 1999)

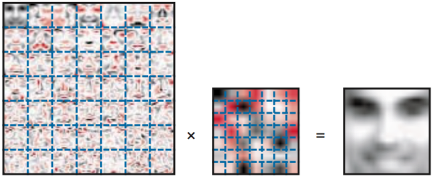
Example (Lee and Seung 1999)

1: PCA

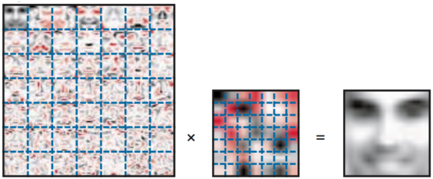
Example (Lee and Seung 1999)

1: PCA 

Example (Lee and Seung 1999)

1: PCA  "eigenfaces"

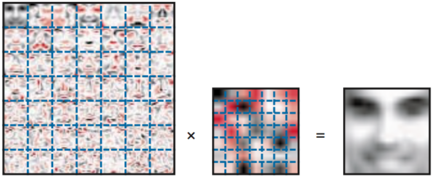
Example (Lee and Seung 1999)

1: PCA  “eigenfaces”

2: VQ

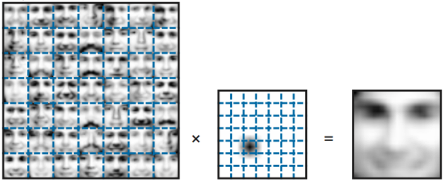
Example (Lee and Seung 1999)

1: PCA



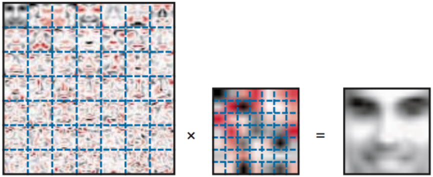
“eigenfaces”

2: VQ



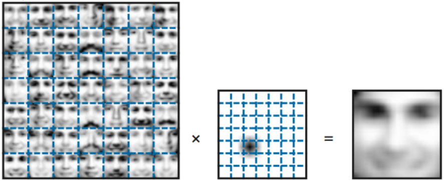
Example (Lee and Seung 1999)

1: PCA



“eigenfaces”

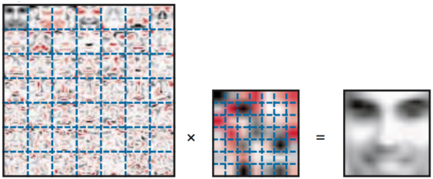
2: VQ



“protofaces”

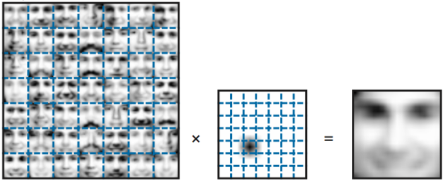
Example (Lee and Seung 1999)

1: PCA



“eigenfaces”

2: VQ

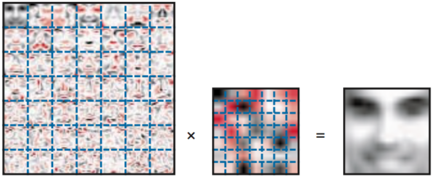


“protofaces”

3: NMF

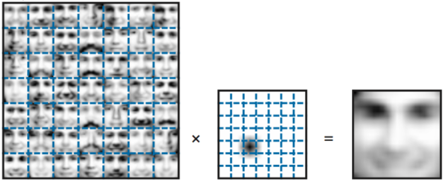
Example (Lee and Seung 1999)

1: PCA



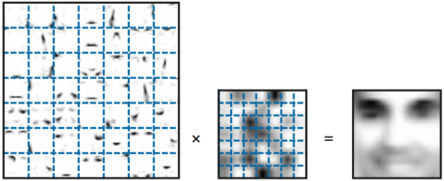
“eigenfaces”

2: VQ

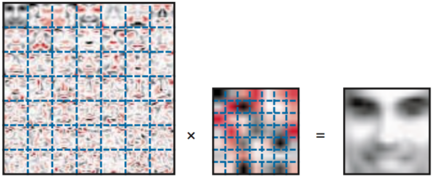


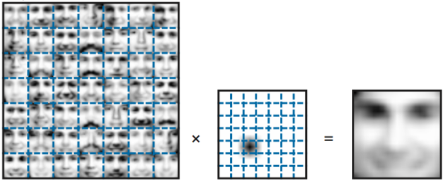
“protofaces”

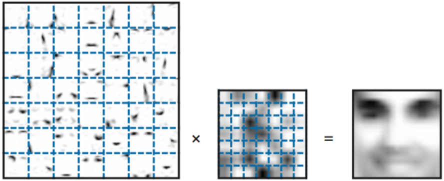
3: NMF



Example (Lee and Seung 1999)

1: PCA  "eigenfaces"

2: VQ  "protofaces"

3: NMF  parts-based

NNMF learns an additive, parts based model

NNMF learns an additive, parts based model

NNMF learns an additive, parts based model

which is what our brain does when recognizing images!