GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

LECTURE 13: DIMENSIONALITY REDUCTION

I. DIMENSIONALITY REDUCTION
II. PRINCIPAL COMPONENTS ANALYSIS
III. SINGULAR VALUE DECOMPOSITION
IV. OTHER METHODS

EXERCISE:
IV. DIMENSIONALITY REDUCTION IN SCIKIT-LEARN

I. DIMENSIONALITY REDUCTION

Q: What is dimensionality reduction?

Q: What is dimensionality reduction?

A: A set of techniques for reducing the size (in terms of features, records, and/or bytes) of the dataset under examination.

Q: What is dimensionality reduction?

A: A set of techniques for reducing the size (in terms of features, records, and/or bytes) of the dataset under examination.

In general, the idea is to regard the dataset as a matrix and to decompose the matrix into simpler, meaningful pieces.

Q: What is dimensionality reduction?

A: A set of techniques for reducing the size (in terms of features, records, and/or bytes) of the dataset under examination.

In general, the idea is to regard the dataset as a matrix and to decompose the matrix into simpler, meaningful pieces.

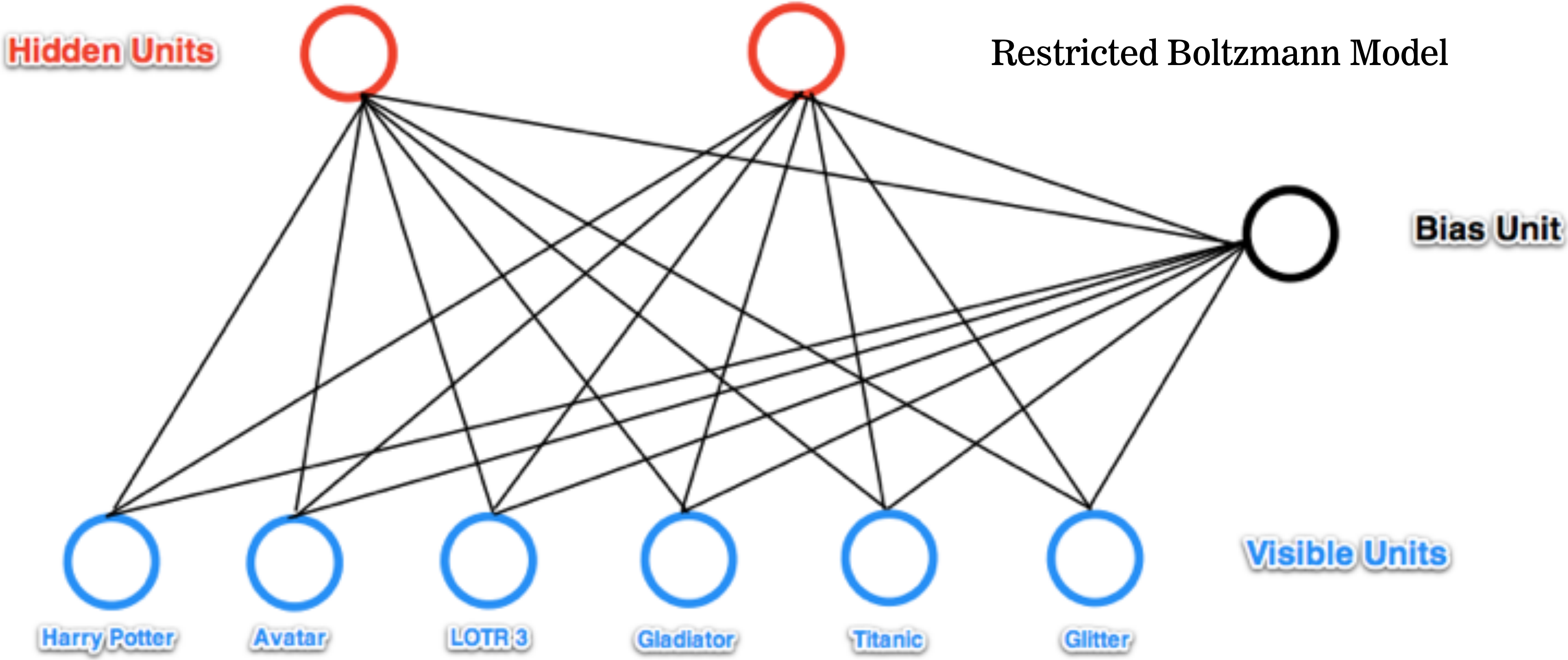
Dimensionality reduction is frequently performed as a pre-processing step before another learning algorithm is applied.

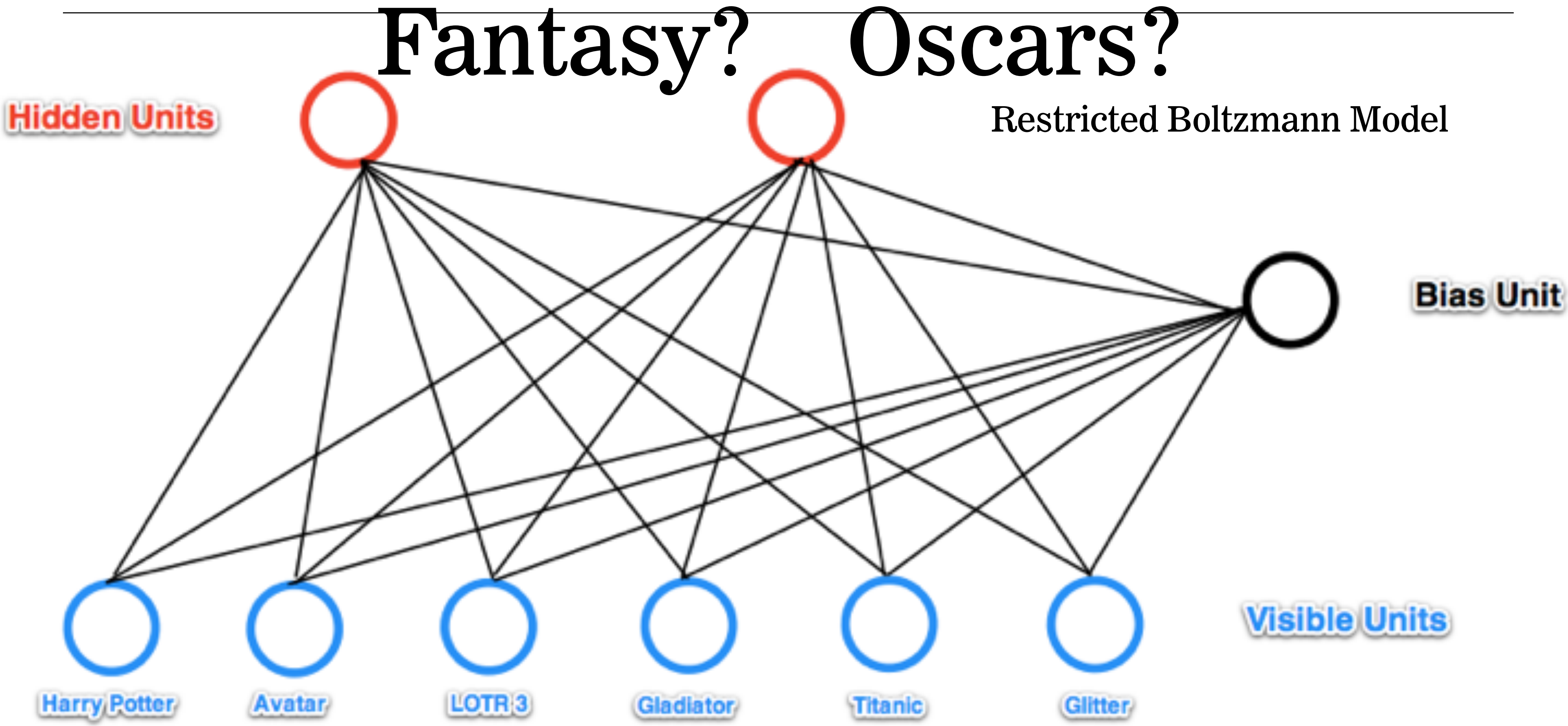
Q: What are the motivations for dimensionality reduction?

Q: What are the motivations for dimensionality reduction?

The number of features in our dataset can be difficult to manage, or even misleading (eg, if the relationships are actually simpler than they appear).

Restricted Boltzmann Model




Fantasy?
Oscars?
Hidden Units
Restricted Boltzmann Model
Bias Unit
Visible Units
Harry Potter
Avatar
LOTR 3
Gladiator
Titanic
Glitter

Q: What is the goal of dimensionality reduction?

Q: What is the goal of dimensionality reduction?

- *reduce computational expense*
- *reduce susceptibility to overfitting*
- *reduce noise in the dataset*
- *enhance our intuition*

Q: How is dimensionality reduction performed?

Q: How is dimensionality reduction performed?

A: There are two approaches: feature selection and feature extraction.

Q: How is dimensionality reduction performed?

A: There are two approaches: feature selection and feature extraction.

feature selection – *selecting a subset of features using an external criterion (filter) or the learning algo accuracy itself (wrapper)*

feature extraction – *mapping the features to a lower dimensional space*

Feature selection is important, but typically when people say dimensionality reduction, they are referring to feature extraction.

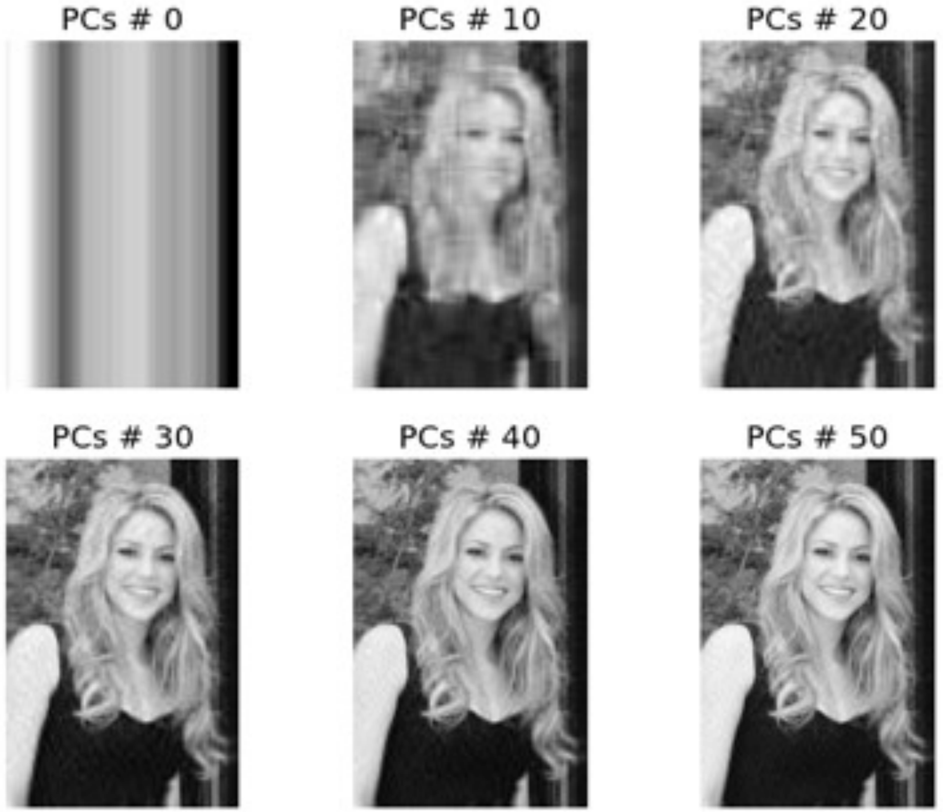
The goal of feature extraction is to create a new set of coordinates that simplify the representation of the data.

Q: What are some applications of dimensionality reduction?

Q: What are some applications of dimensionality reduction?

- *topic models (document clustering)*
- *image recognition/computer vision*
- *recommender systems*

II. PRINCIPAL COMPONENT ANALYSIS



source: http://glowingpython.blogspot.it/2011/07/pca-and-image-compression-with-numpy.html

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

This procedure produces a new basis, each of whose components retain as much variance from the original data as possible.

Principal component analysis is a dimension reduction technique that can be used on a matrix of any dimensions.

This procedure produces a new basis, each of whose components retain as much variance from the original data as possible.

The PCA of a matrix A boils down to the **eigenvalue decomposition** *of the* **covariance matrix** *of A.*

The covariance matrix C of a matrix A is always square:

$$C = \begin{bmatrix} \mathrm{E}[(X_1-\mu_1)(X_1-\mu_1)] & \mathrm{E}[(X_1-\mu_1)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_1-\mu_1)(X_n-\mu_n)] \\ \mathrm{E}[(X_2-\mu_2)(X_1-\mu_1)] & \mathrm{E}[(X_2-\mu_2)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_2-\mu_2)(X_n-\mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}[(X_n-\mu_n)(X_1-\mu_1)] & \mathrm{E}[(X_n-\mu_n)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_n-\mu_n)(X_n-\mu_n)] \end{bmatrix}.$$

off-diagonal elements C_{ij} give the covariance between X_i, X_j $(i \neq j)$

diagonal elements C_{ii} give the variance of X_i

The eigenvalue decomposition of a square matrix A is given by:

$$A = Q \Lambda Q^{-1}$$

The eigenvalue decomposition of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The columns of Q are the **eigenvectors** *of A, and the values in Λ are the associated* **eigenvalues** *of A.*

The eigenvalue decomposition of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The columns of Q are the **eigenvectors** *of A, and the values in Λ are the associated* **eigenvalues** *of A.*

For an eigenvector v of A and its eigenvalue λ, we have the important relation:

$$Av = \lambda v$$

The eigenvalue decomposition of a square matrix A is given by:

$$A = Q\Lambda Q^{-1}$$

The columns of Q are the **eigenvectors** *of A, and the valu*

the associated **eigenvalues** *of A.*

NOTE

This relationship *defines* what it means to be an eigenvector of A.

For an eigenvector v of A and its eigenvalue λ, we have the important relation:

$$Av = \lambda v$$

The eigenvectors form a basis of the vector space on which A acts (eg, they are orthogonal).

The eigenvectors form a basis of the vector space on which A acts (eg, they are orthogonal).

Furthermore the basis elements are ordered by their eigenvalues (from largest to smallest), and these eigenvalues represent the amount of variance explained by each basis element.

III. SINGULAR VALUE DECOMPOSITION

Consider a matrix A with n rows and d features.

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

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

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

Consider a matrix A *with* n *rows and* d *features.*

The **singular value decomposition** *of* A *is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

st. U, V *are* **orthogonal** *matrices and* Σ *is a* **diagonal** *matrix.*

Consider a matrix A with n rows and d features.

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U} \; \underset{(n \times d)}{\Sigma} \; \underset{(d \times d)}{V^T}$$

st. U, V are **orthogonal** *matrices and Σ is a* **diagonal** *matrix.*

→ $UU^T = I_n, \; VV^T = I_d$ → $\Sigma_{ij} = 0 \;\; (i \neq j)$

The **singular value decomposition** *of A is given by:*

$$\underset{(n \times d)}{A} = \underset{(n \times n)}{U}\ \underset{(n \times d)}{\Sigma}\ \underset{(d \times d)}{V^T}$$

The nonzero entries of Σ are the **singular values** *of A. These are real, nonnegative, and rank-ordered (decreasing from left to right).*

The **singular value decomposition** *of A is given by:*

$$A = U \, \Sigma \, V^T$$

(n x d) (n x n) (n x d) (d x d)

NOTE

The number of singular values is equal to the *rank* of A.

The rank of a matrix measures its *non-degeneracy*.

The nonzero entries of Σ are the **singular values** *of A. These are real, nonnegative, and rank-ordered (decreasing from left to right).*

Q: How do you interpret the SVD?

Q: How do you interpret the SVD?

A: Recall that given a set of n points in d-dimensional space (eg, a matrix A), we want to find the best $k < d$ dimensional subspace to represent the data.

Q: How do you interpret the SVD?

A: Recall that given a set of n points in d-dimensional space (eg, a matrix A), we want to find the best $k < d$ dimensional subspace to represent the data.

For k = 1, this subspace is a line passing through the origin.

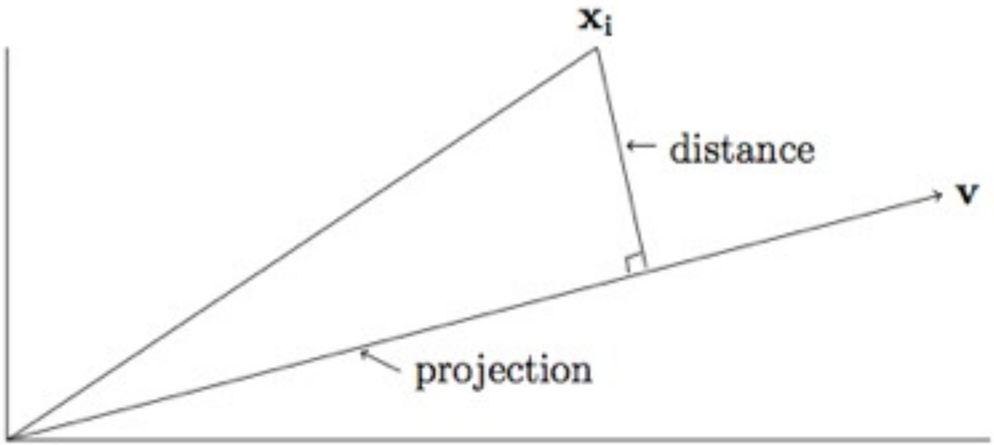


Figure 4.1: The projection of the point $\mathbf{x}_i$ onto the line through the origin in the direction of $\mathbf{v}$

source: http://www.cs.princeton.edu/courses/archive/spring12/cos598C/svdchapter.pdf

$$
\begin{array}{cc}
 & \begin{array}{ccccc}\text{data} & \text{inf.} & \text{retrieval} & \text{brain} & \text{lung}\end{array} \\
\begin{array}{c}\text{CS}\\ \\ \\ \\ \text{MD}\\ \\ \end{array} &
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 \\
2 & 2 & 2 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
5 & 5 & 5 & 0 & 0 \\
0 & 0 & 0 & 2 & 2 \\
0 & 0 & 0 & 3 & 3 \\
0 & 0 & 0 & 1 & 1
\end{bmatrix}
\end{array}
=
\underbrace{\begin{bmatrix}
0.18 & 0 \\
0.36 & 0 \\
0.18 & 0 \\
0.90 & 0 \\
0 & 0.53 \\
0 & 0.80 \\
0 & 0.27
\end{bmatrix}}_{\text{doc-to-concept similarity matrix}}
\text{x}
\underbrace{\begin{bmatrix}
9.64 & 0 \\
0 & 5.29
\end{bmatrix}}_{\text{concepts strengths}}
\text{x}
\underbrace{\begin{bmatrix}
0.58 & 0.58 & 0.58 & 0 & 0 \\
0 & 0 & 0 & 0.71 & 0.71
\end{bmatrix}}_{\text{term-to-concept similarity matrix}}
$$

In any case, the key difficulties with dimensionality reduction are time/space complexity, randomness (eg different results for different runs), and selecting the number of dimensions in the lower-dim subspace.