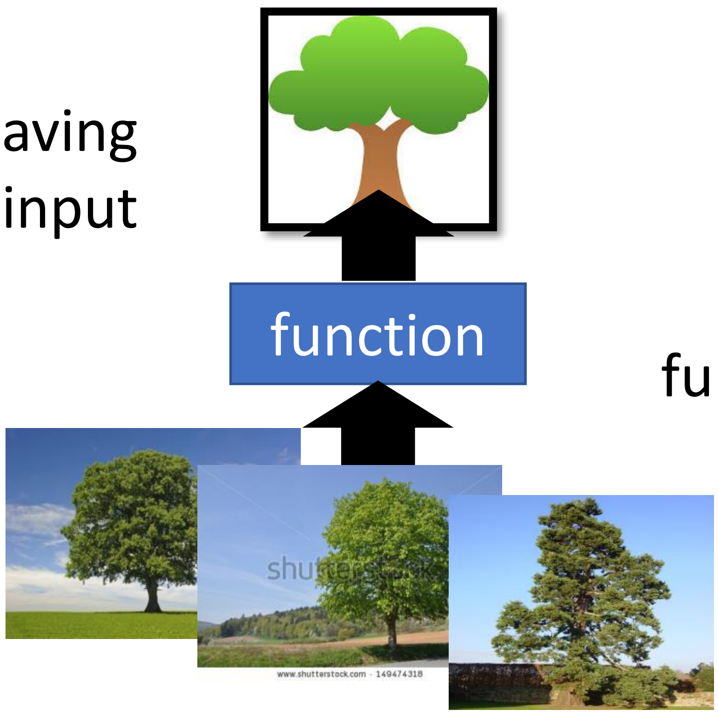
Unsupervised Learning:
Linear Dimension Reduction

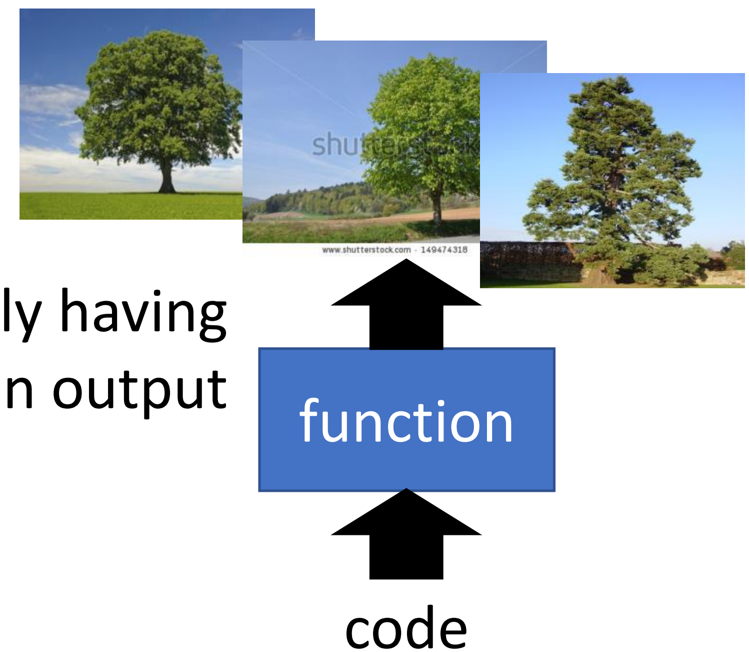
Unsupervised Learning

- Clustering & Dimension Reduction (化繁為簡)
- Generation (無中生有)

only having
function input



only having
function output



Clustering & Dimension Reduction in these slides

Clustering

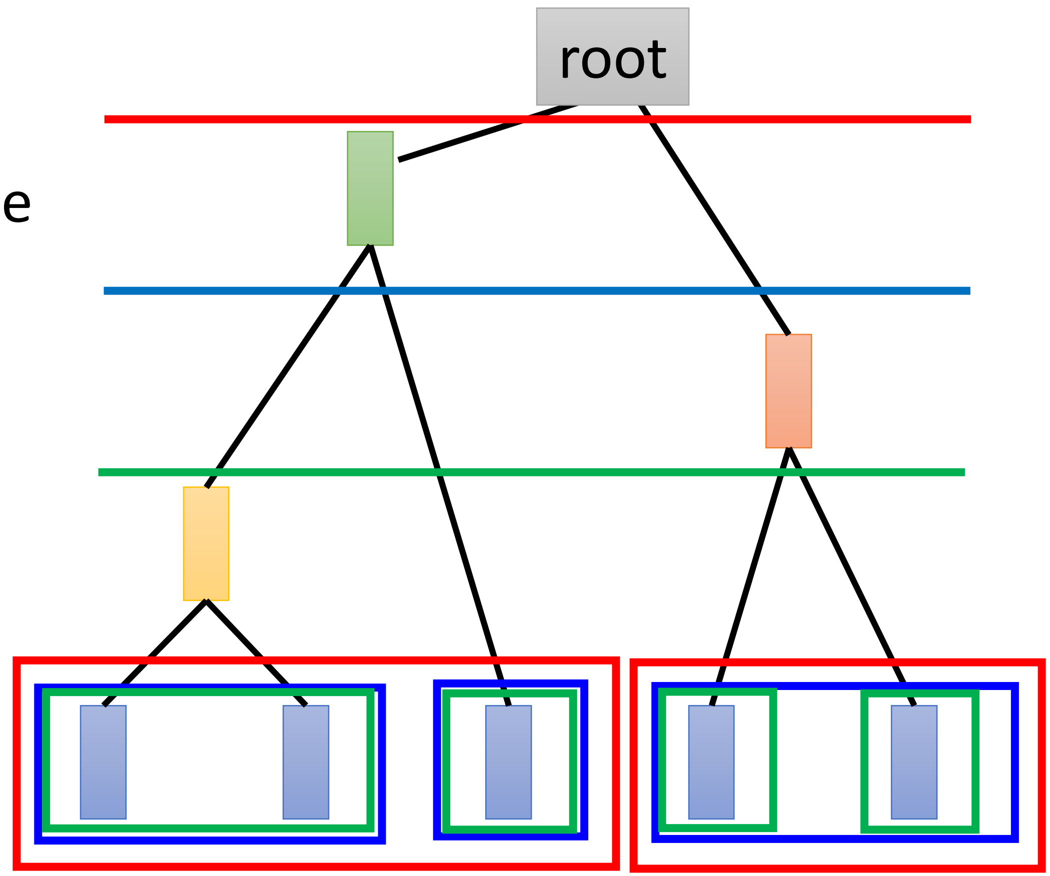
- K-means
  - Clustering $X = \{x^1, \cdots, x^n, \cdots, x^N\}$ into K clusters
  - Initialize cluster center c^i, i=1,2, ... K (K random x^n from X)
  - Repeat
    - For all x^n in X: $b_i^n \begin{cases} 1 & x^n \text{ is most "\textbf{\textit{close}}" to } c^i \\ 0 & \text{Otherwise} \end{cases}$
    - Updating all c^i: $c^i = \sum_{x^n} b_i^n x^n \Big/ \sum_{x^n} b_i^n$

Clustering

- Hierarchical Agglomerative Clustering (HAC)

Step 1: build a tree

Step 2: pick a threshold



Distributed Representation

- Clustering: an object must belong to one cluster

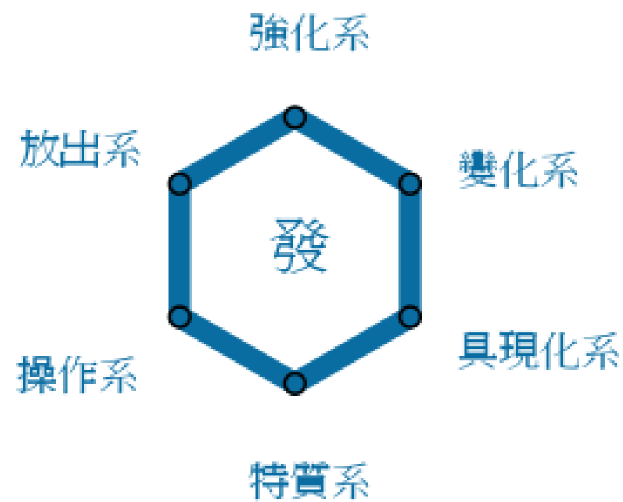
小傑是強化系

- Distributed representation

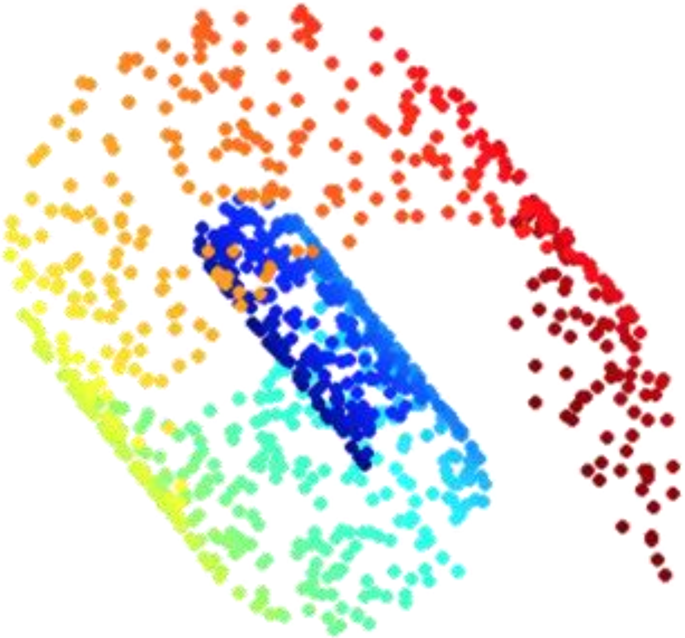
Dimension Reduction

小傑是

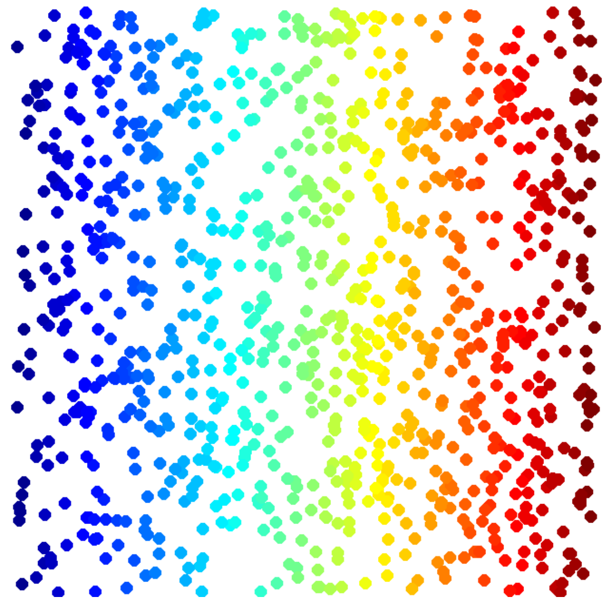
強化系	0.70
放出系	0.25
變化系	0.05
操作系	0.00
具現化系	0.00
特質系	0.00



Dimension Reduction



Looks like 3-D

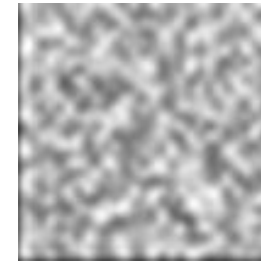


Actually, 2-D

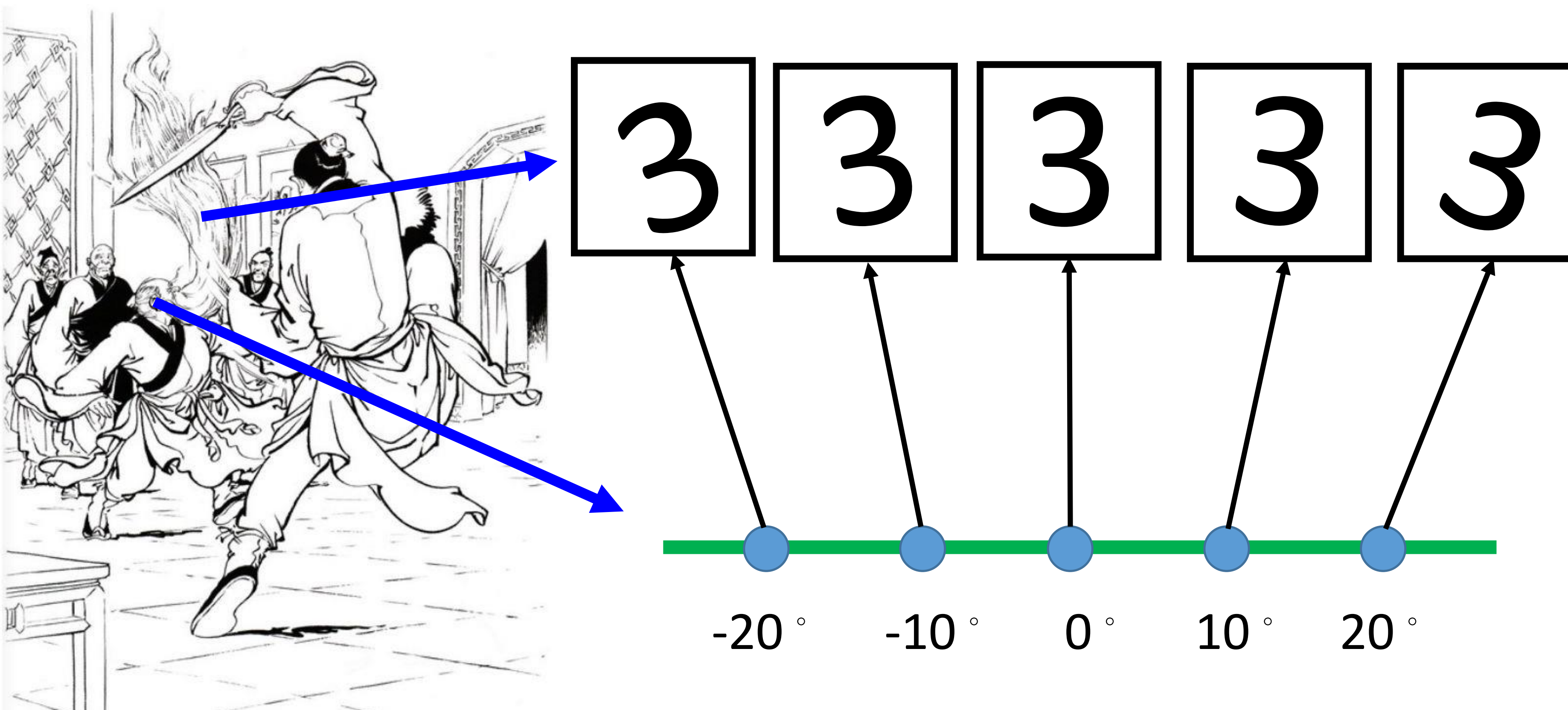
http://reuter.mit.edu/blue/images/research/manifold.png

http://archive.cnx.org/resources/51a9b2052ae167db310fda5600b89badea85eae5/isomapCNXtrue1.png

Dimension Reduction



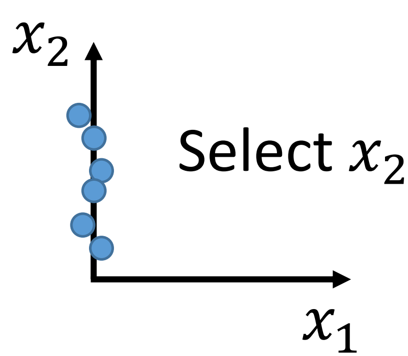
- In MNIST, a digit is 28 x 28 dims.
  - Most 28 x 28 dim vectors are not digits

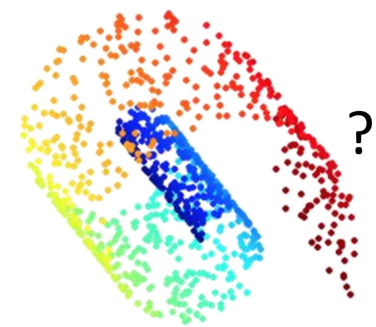


Dimension Reduction

The dimension of z would be smaller than x

- Feature selection





- Principle component analysis (PCA)
  [Bishop, Chapter 12]

$$z = Wx$$

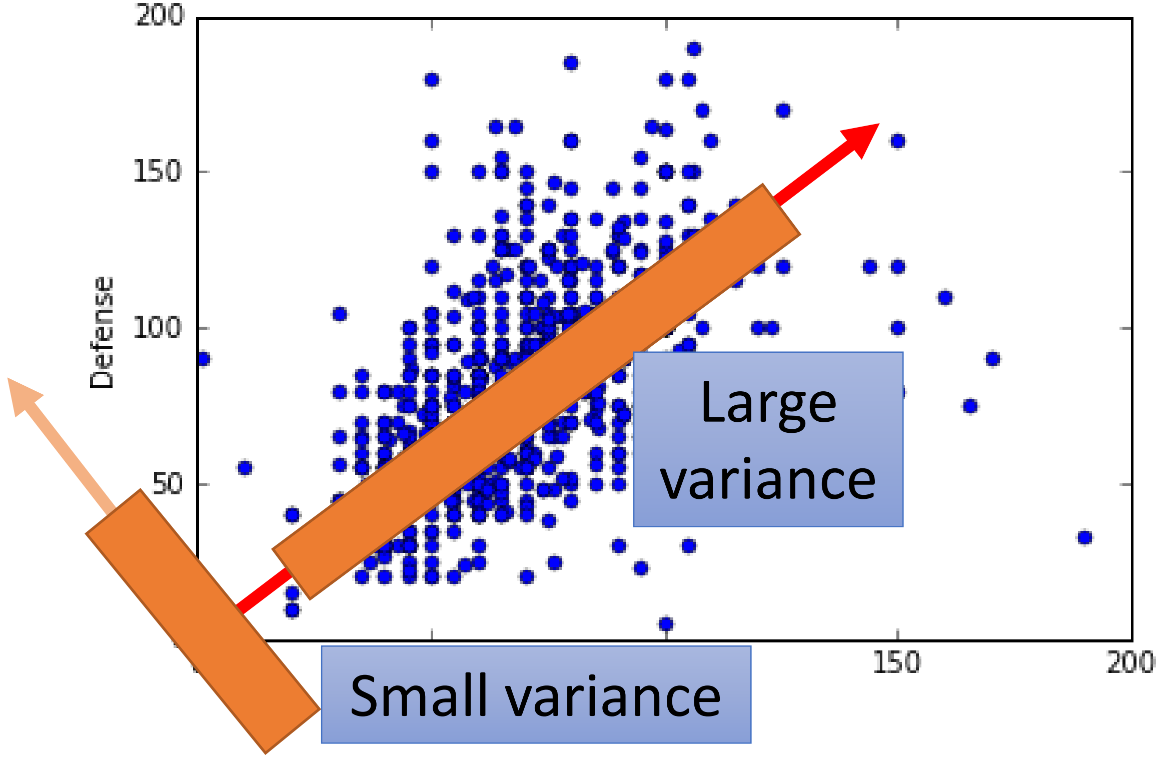
Principle Component Analysis (PCA)

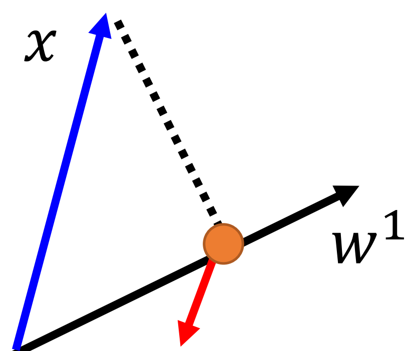
PCA

$$z = Wx$$

Reduce to 1-D:

$$z_1 = w^1 \cdot x$$





$$z_1 = w^1 \cdot x$$

Project all the data points x onto w^1, and obtain a set of z_1

We want the variance of z_1 as large as possible

$$Var(z_1) = \sum_{z_1} (z_1 - \bar{z_1})^2 \quad \|w^1\|_2 = 1$$

PCA

$$z = Wx$$

Reduce to 1-D:

$$z_1 = w^1 \cdot x$$

$$z_2 = w^2 \cdot x$$

$$W = \begin{bmatrix} (w^1)^T \\ (w^2)^T \\ \vdots \end{bmatrix}$$

Orthogonal matrix

Project all the data points x onto w^1, and obtain a set of z_1

We want the variance of z_1 as large as possible

$$Var(z_1) = \sum_{z_1} (z_1 - \bar{z_1})^2 \quad \|w^1\|_2 = 1$$

We want the variance of z_2 as large as possible

$$Var(z_2) = \sum_{z_2} (z_2 - \bar{z_2})^2 \quad \|w^2\|_2 = 1$$

$$w^1 \cdot w^2 = 0$$

Warning of Math

PCA

$$z_1 = w^1 \cdot x$$

$$\bar{z_1} = \frac{1}{N}\sum z_1 = \frac{1}{N}\sum w^1 \cdot x = w^1 \cdot \frac{1}{N}\sum x = w^1 \cdot \bar{x}$$

$$Var(z_1) = \frac{1}{N}\sum_{z_1}(z_1 - \bar{z_1})^2$$

$$= \frac{1}{N}\sum_{x}(w^1 \cdot x - w^1 \cdot \bar{x})^2$$

$$= \frac{1}{N}\sum\left(w^1 \cdot (x - \bar{x})\right)^2$$

$$= \frac{1}{N}\sum(w^1)^T(x - \bar{x})(x - \bar{x})^T w^1$$

$$= (w^1)^T \boxed{\frac{1}{N}\sum(x - \bar{x})(x - \bar{x})^T} w^1$$

$$= (w^1)^T Cov(x) w^1$$

$$S = Cov(x)$$

$$(a \cdot b)^2 = (a^T b)^2 = a^T b a^T b$$

$$= a^T b (a^T b)^T = a^T b b^T a$$

$$\boxed{\text{Find } w^1 \text{ maximizing } (w^1)^T S w^1, \quad \|w^1\|_2 = (w^1)^T w^1 = 1}$$

Find w^1 maximizing $(w^1)^T S w^1$ $\quad (w^1)^T w^1 = 1$

$S = Cov(x)$ Symmetric positive-semidefinite (non-negative eigenvalues)

Using Lagrange multiplier [Bishop, Appendix E]

$$g(w^1) = (w^1)^T S w^1 - \alpha\left((w^1)^T w^1 - 1\right)$$

$$\left.\begin{array}{l} \partial g(w^1)/\partial w_1^1 = 0 \\ \partial g(w^1)/\partial w_2^1 = 0 \\ \vdots \end{array}\right\}$$

$$S w^1 - \alpha w^1 = 0$$

$$S w^1 = \alpha w^1$$

w^1 : eigenvector

$$(w^1)^T S w^1 = \alpha (w^1)^T w^1$$

$$= \alpha$$

Choose the maximum one

w^1 is the eigenvector of the covariance matrix S

Corresponding to the largest eigenvalue λ_1

Find w^2 maximizing $(w^2)^T S w^2$ $\quad (w^2)^T w^2 = 1 \quad (w^2)^T w^1 = 0$

$$g(w^2) = (w^2)^T S w^2 - \alpha\left((w^2)^T w^2 - 1\right) - \beta\left((w^2)^T w^1 - 0\right)$$

$$\left.\begin{array}{l} \partial g(w^2)/\partial w_1^2 = 0 \\ \partial g(w^2)/\partial w_2^2 = 0 \\ \vdots \end{array}\right\} \quad S w^2 - \alpha w^2 - \beta w^1 = 0$$

$$\underline{\boxed{0}} - \alpha \underline{\boxed{0}} - \beta \underline{\boxed{1}} = 0$$

$$= \left((w^1)^T S w^2\right)^T = (w^2)^T S^T w^1$$

$$= (w^2)^T S w^1 = \lambda_1 (w^2)^T w^1 = 0$$

$$\boxed{S w^1 = \lambda_1 w^1}$$

$\beta = 0$: $\quad S w^2 - \alpha w^2 = 0 \qquad S w^2 = \alpha w^2$

w^2 is the eigenvector of the covariance matrix S

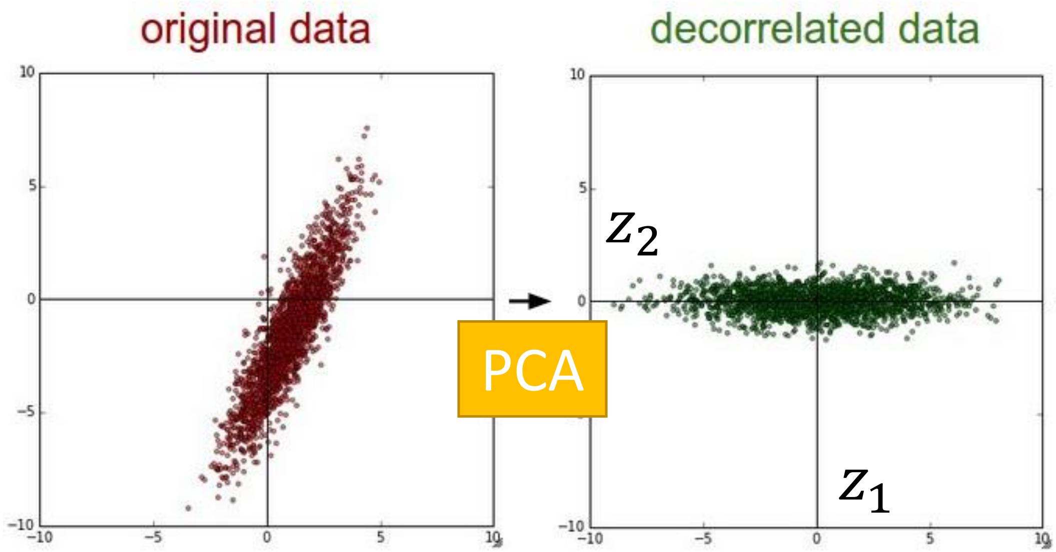
Corresponding to the 2nd largest eigenvalue λ_2

PCA - decorrelation

$$z = Wx$$

$$Cov(z) = D$$

Diagonal matrix



$$Cov(z) = \frac{1}{N}\sum(z-\bar{z})(z-\bar{z})^T = WSW^T \qquad S = Cov(x)$$

$$= WS\begin{bmatrix} w^1 & \cdots & w^K \end{bmatrix} = W\begin{bmatrix} Sw^1 & \cdots & Sw^K \end{bmatrix}$$

$$= W\begin{bmatrix} \lambda_1 w^1 & \cdots & \lambda_K w^K \end{bmatrix} = \begin{bmatrix} \lambda_1 Ww^1 & \cdots & \lambda_K Ww^K \end{bmatrix}$$

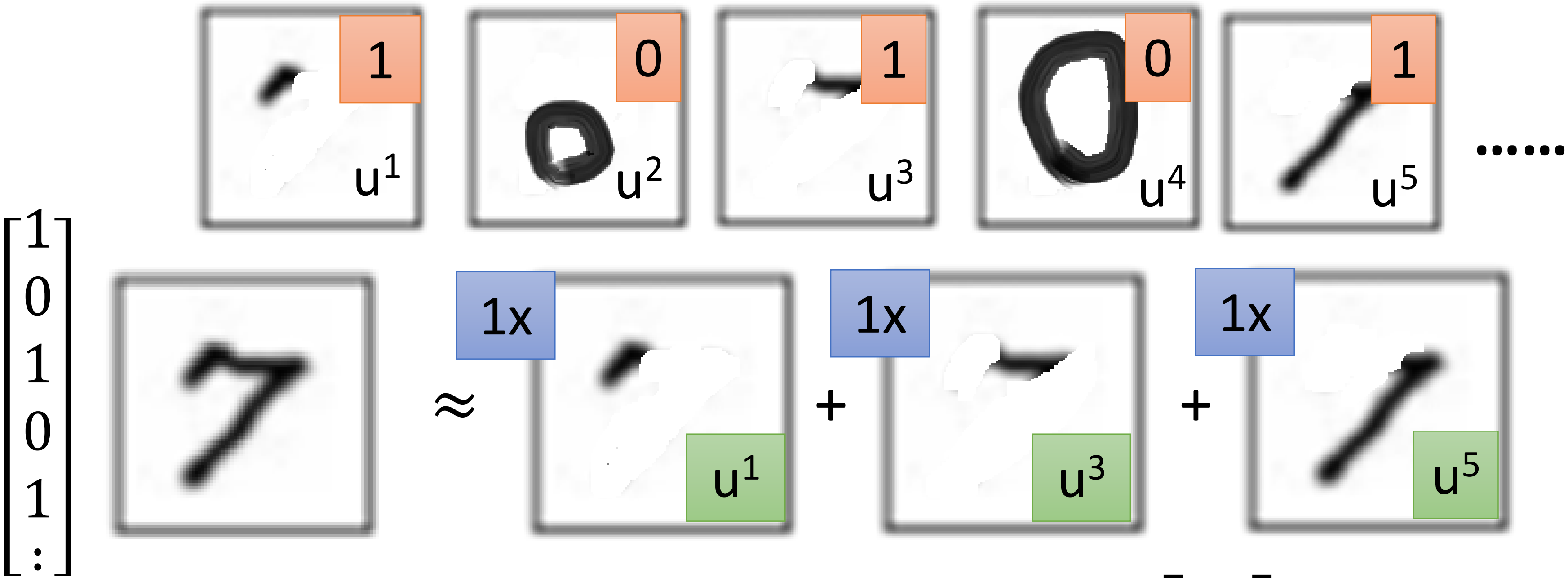
$$= \begin{bmatrix} \lambda_1 e_1 & \cdots & \lambda_K e_K \end{bmatrix} = D$$

Diagonal matrix

End of Warning

PCA – Another Point of View

Basic Component:



$$x \approx c_1 u^1 + c_2 u^2 + \cdots + c_K u^K + \bar{x}$$

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{bmatrix}$$ Represent a digit image

Pixels in a digit image

component

number of

effective when then components is less then pixels

PCA – Another Point of View

$$x - \bar{x} \approx c_1 u^1 + c_2 u^2 + \cdots + c_K u^K \;\; = \hat{x}$$

Reconstruction error:
$\| (x - \bar{x}) - \hat{x} \|_2$

Find $\{u^1, \ldots, u^K\}$ minimizing the error

$$L = \min_{\{u^1,\ldots,u^K\}} \sum \left\| (x - \bar{x}) - \underbrace{\left(\sum_{k=1}^{K} c_k u^k \right)}_{\hat{x}} \right\|_2$$

PCA: $\quad z = Wx$

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_K \end{bmatrix} = \begin{bmatrix} (w_1)^{\mathrm{T}} \\ (w_2)^{\mathrm{T}} \\ \vdots \\ (w_K)^{\mathrm{T}} \end{bmatrix} x$$

$\{w^1, w^2, \ldots w^K\}$ is the component $\{u^1, u^2, \ldots u^K\}$ minimizing L

Proof in [Bishop, Chapter 12.1.2]

$$x - \bar{x} \approx c_1 u^1 + c_2 u^2 + \cdots + c_K u^K \ = \hat{x}$$

Reconstruction error:
$\| (x - \bar{x}) - \hat{x} \|_2$

Find $\{u^1, \ldots, u^K\}$ minimizing the error

$$x^1 - \bar{x} \approx c_1^1 u^1 + c_2^1 u^2 + \cdots$$

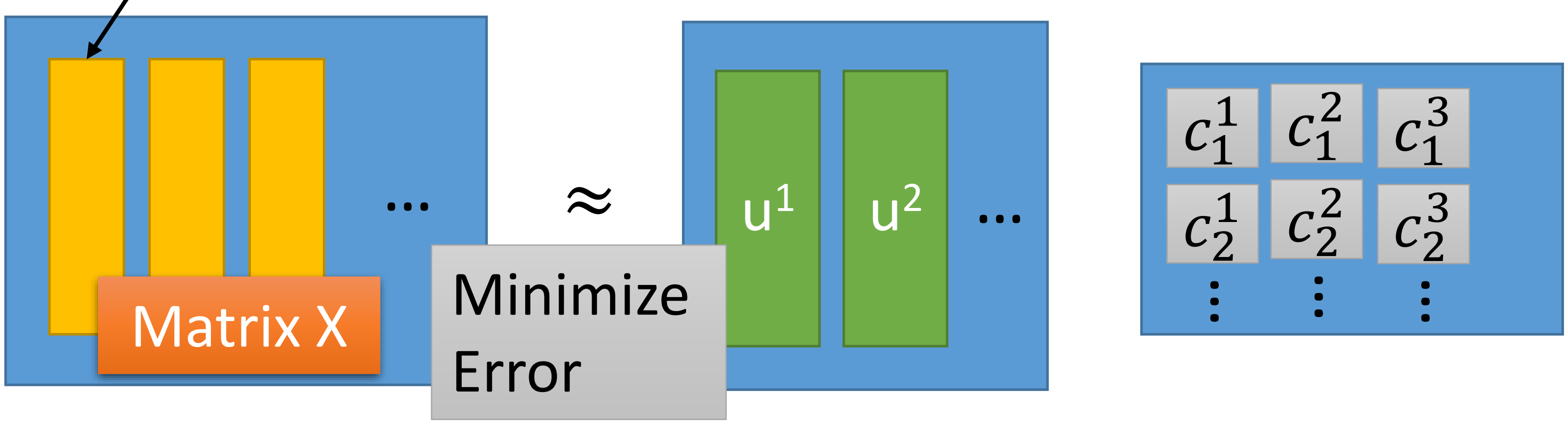
$$x^2 - \bar{x} \approx c_1^2 u^1 + c_2^2 u^2 + \cdots$$

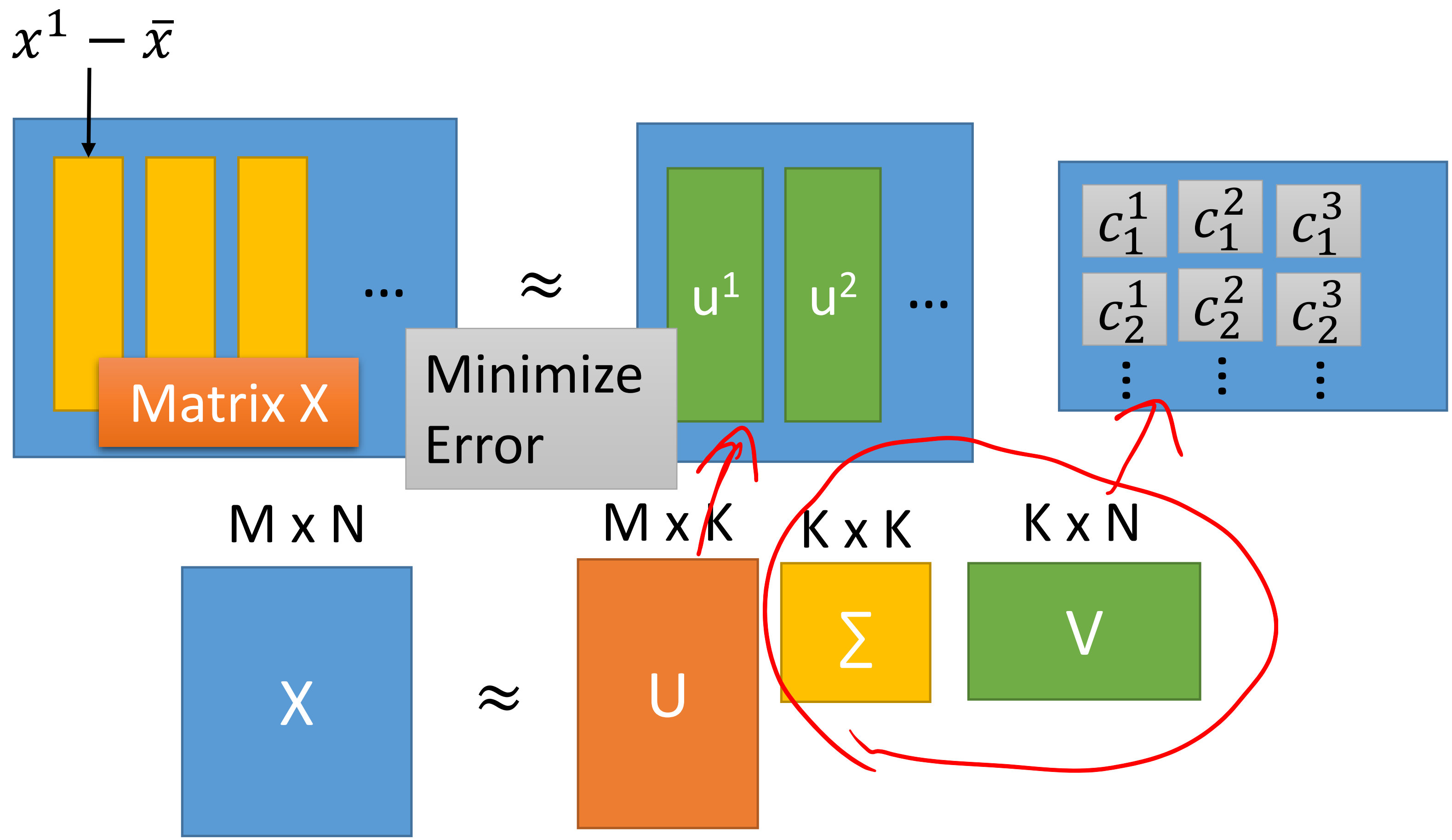
$$x^3 - \bar{x} \approx c_1^3 u^1 + c_2^3 u^2 + \cdots$$

$$\vdots$$

5xL

5x3 3x2





K columns of U: a set of orthonormal eigen vectors corresponding to the k largest eigenvalues of XX^T

This is the solution of PCA

SVD: http://speech.ee.ntu.edu.tw/~tlkagk/courses/LA_2016/Lecture/SVD.pdf

PCA looks like a neural network with one hidden layer (linear activation function)

Autoencoder

If $\{w^1, w^2, \dots w^K\}$ is the component $\{u^1, u^2, \dots u^K\}$

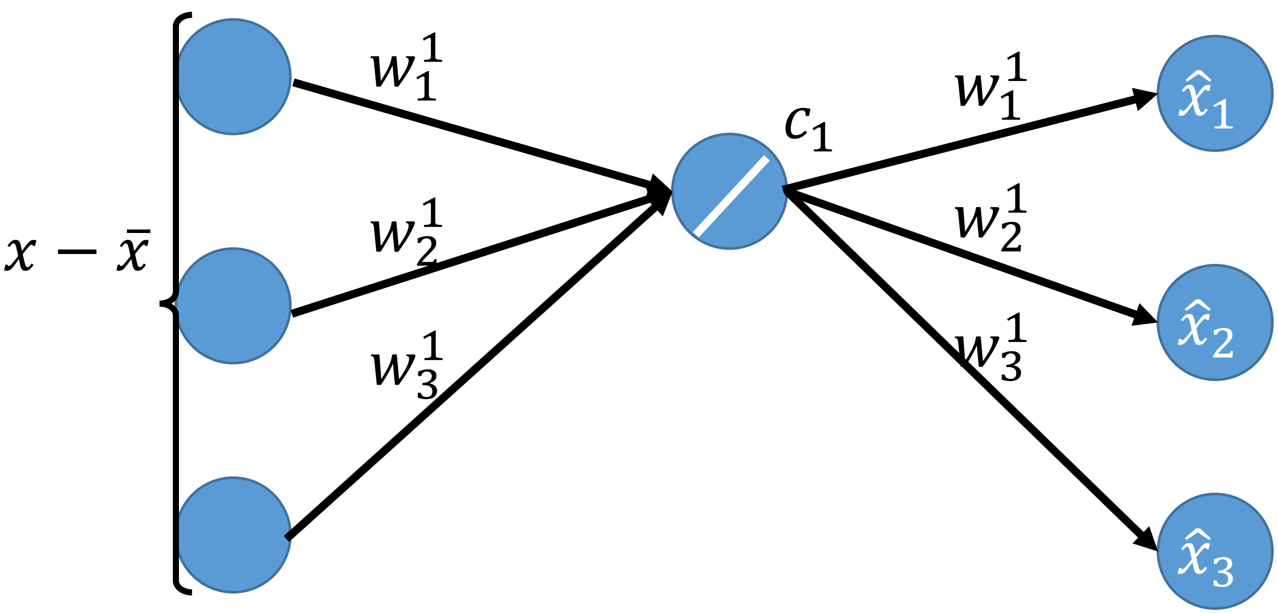
$$\hat{x} = \sum_{k=1}^{K} c_k w^k \Leftrightarrow x - \bar{x}$$

To minimize reconstruction error:

$$c_k = (x - \bar{x}) \cdot w^k$$

orthonormal

$K = 2$:



PCA looks like a neural network with one hidden layer (linear activation function)

Autoencoder

If $\{w^1, w^2, \dots w^K\}$ is the component $\{u^1, u^2, \dots u^K\}$

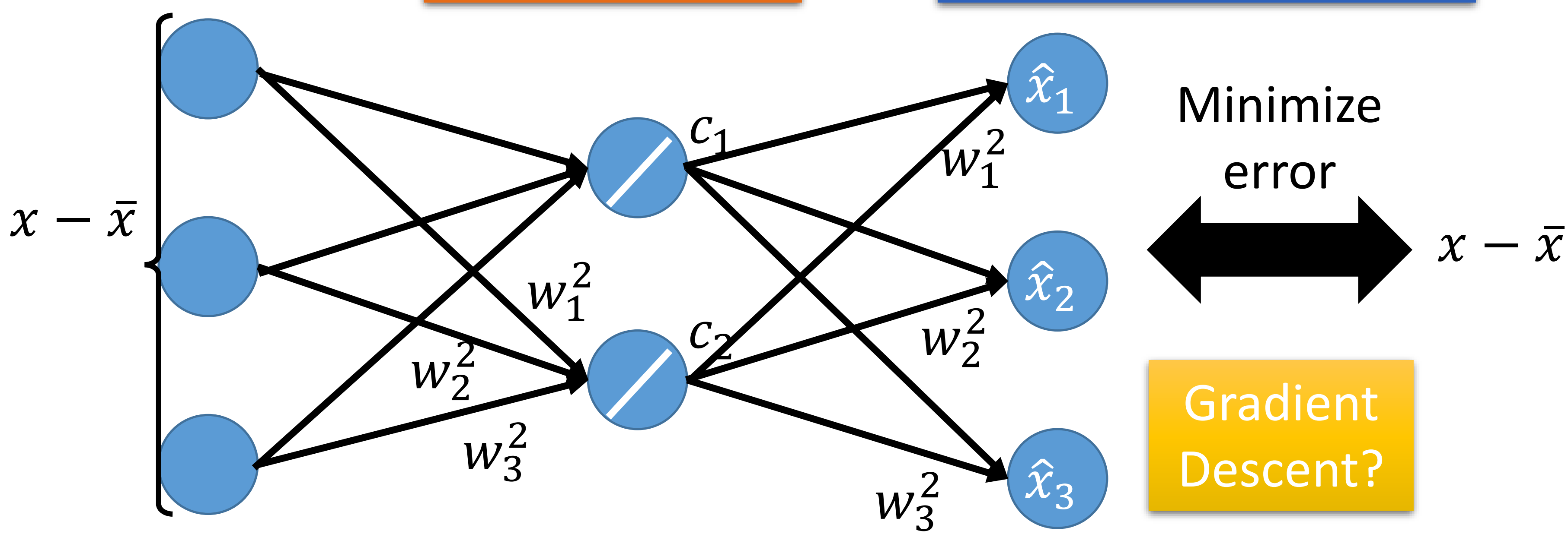
$$\hat{x} = \sum_{k=1}^{K} c_k w^k \Leftrightarrow x - \bar{x}$$

To minimize reconstruction error:

$$c_k = (x - \bar{x}) \cdot w^k$$

$K = 2$:

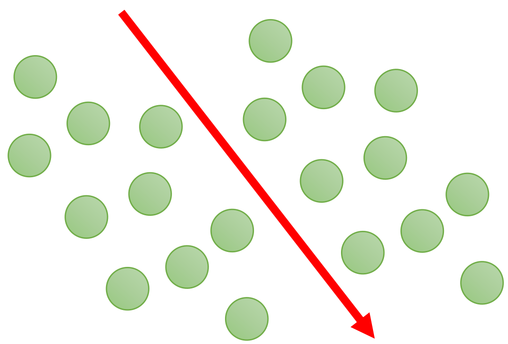
It can be deep. → Deep Autoencoder



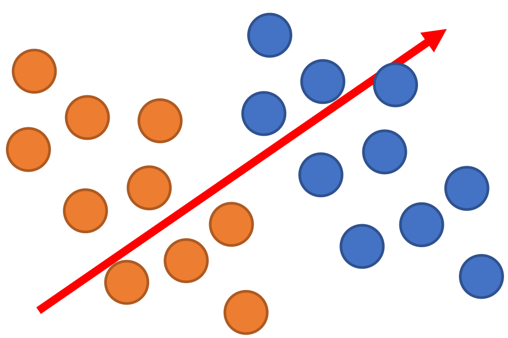
Gradient Descent?

Weakness of PCA

- Unsupervised

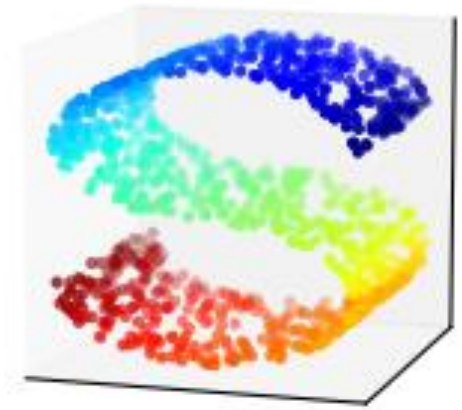


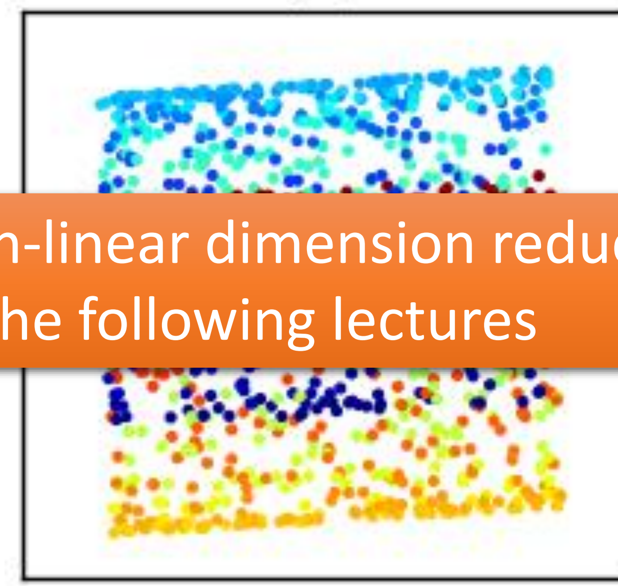
PCA



LDA

- Linear





Non-linear dimension reduction in the following lectures

http://www.astroml.org/book_figures/chapter7/fig_S_manifold_PCA.html

PCA - Pokémon

- Inspired from: https://www.kaggle.com/strakul5/d/abcsds/pokemon/principal-component-analysis-of-pokemon-data
- 800 Pokemons, 6 features for each (HP, Atk, Def, Sp Atk, Sp Def, Speed)
- How many principle components? $\dfrac{\lambda_i}{\lambda_1+\lambda_2+\lambda_3+\lambda_4+\lambda_5+\lambda_6}$

	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6
ratio	0.45	0.18	0.13	0.12	0.07	0.04

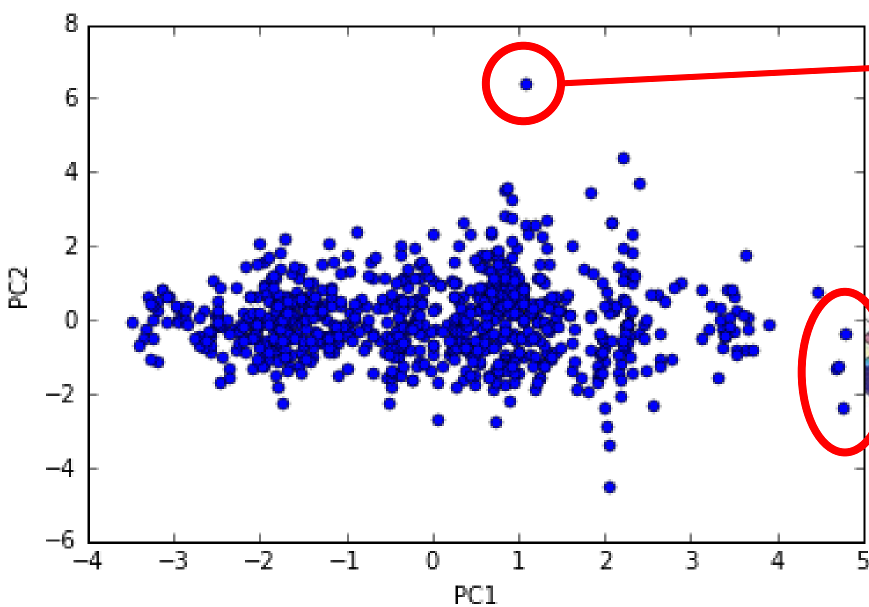
Using 4 components is good enough

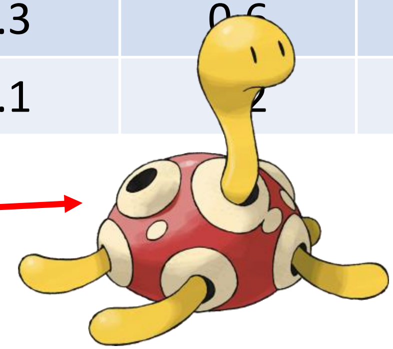
PCA - Pokémon

	HP	Atk	Def	Sp Atk	Sp Def	Speed
PC1	0.4	0.4	0.4	0.5	0.4	0.3
PC2	0.1	0.0	0.6	-0.3	0.2	-0.7
PC3	-0.5	-0.6	0.1	0.3	0.6	[illegible]
PC4	0.7	-0.4	-0.4	0.1	[illegible]	-0.3

強度

防禦(犧牲速度)



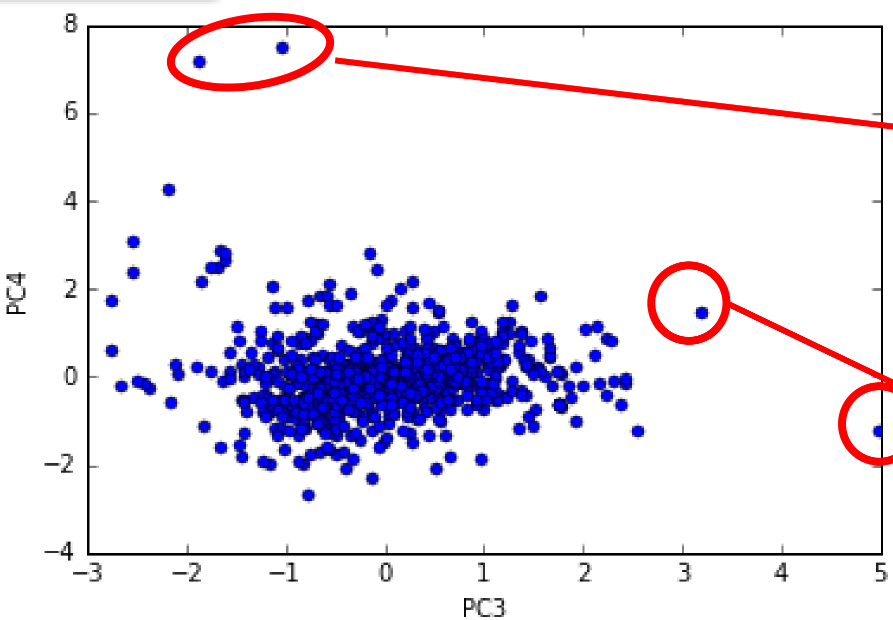


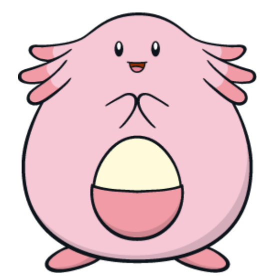
PCA - Pokémon

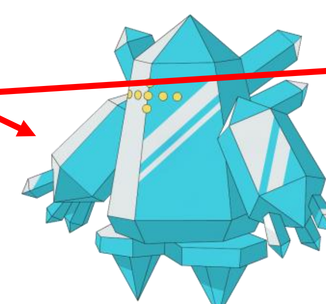
	HP	Atk	Def	Sp Atk	Sp Def	Speed
PC1	0.4	0.4	0.4	0.5	0.4	0.3
PC2	0.1	0.0	0.6	-0.3	0.2	-0.7
PC3	-0.5	-0.6	0.1	0.3	0.6	
	0.7	-0.4	-0.4	0.1	0.2	

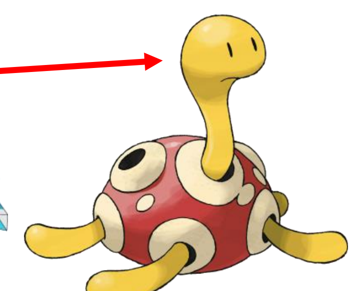
特殊防禦(犧牲攻擊和生命)

生命力強







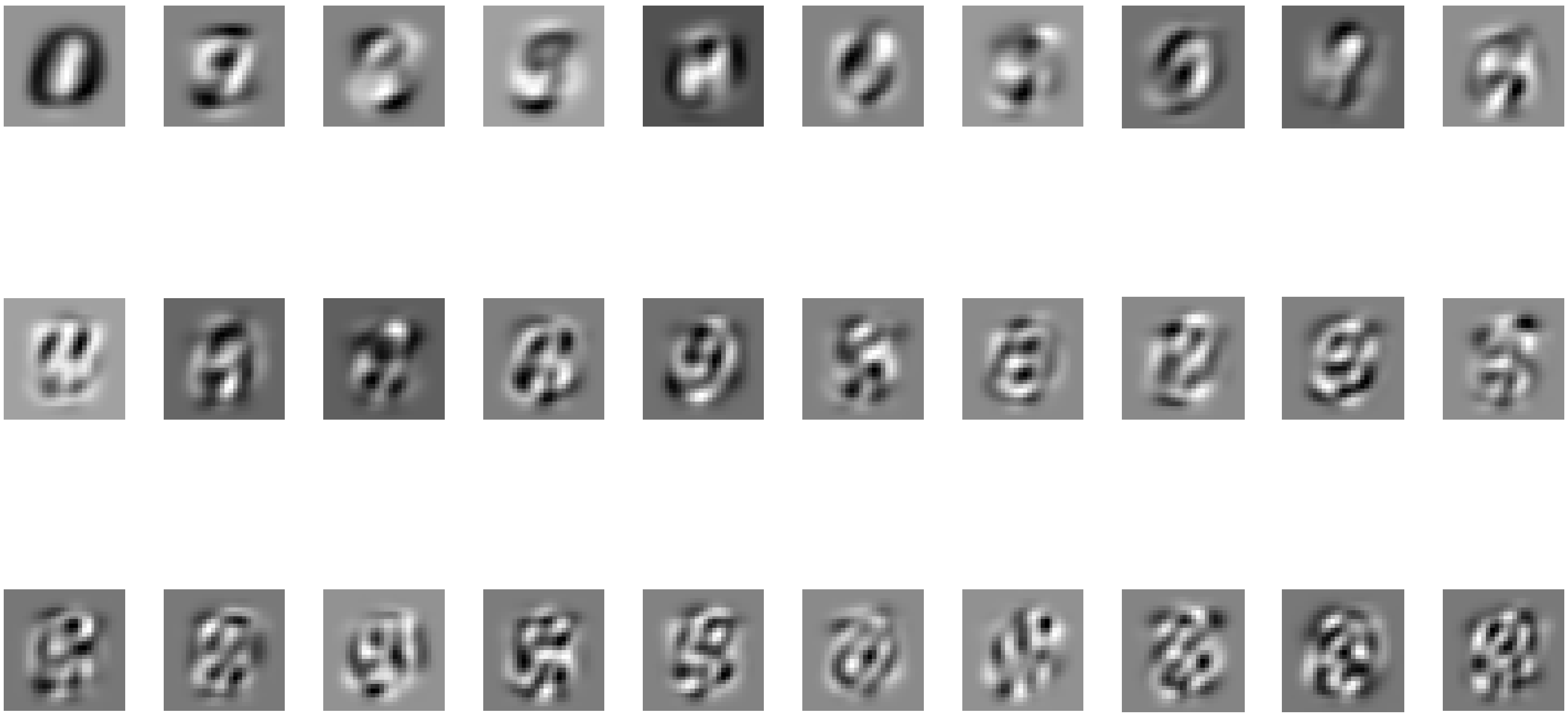


PCA - MNIST

$$= a_1 w^1 + a_2 w^2 + \cdots$$

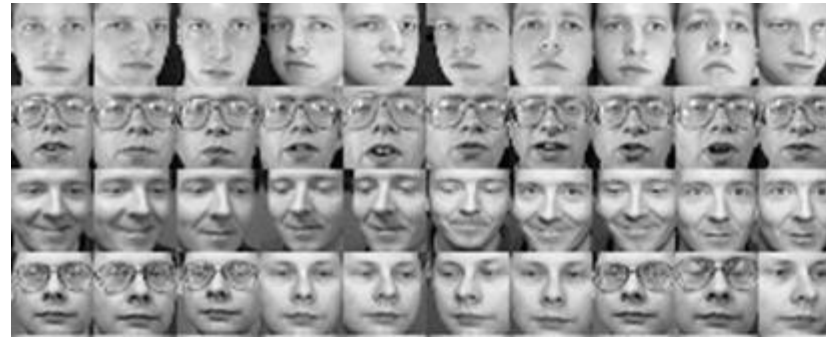
images

30 components:



Eigen-digits

PCA - Face



30 components:

http://www.cs.unc.edu/~lazebnik/research/spring08/assignment3.html

Eigen-face

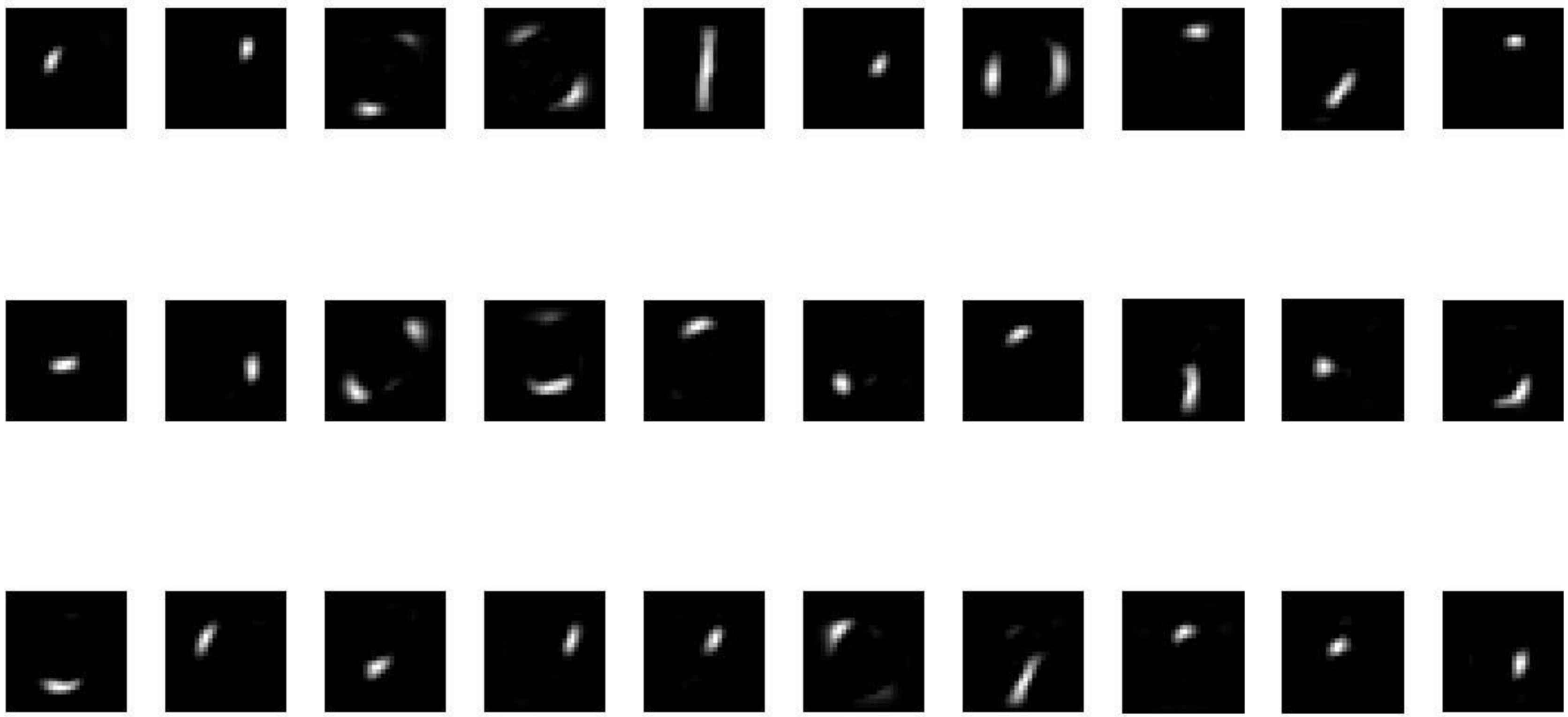
What happens to PCA?

$$= a_1 w^1 + a_2 w^2 + \cdots$$

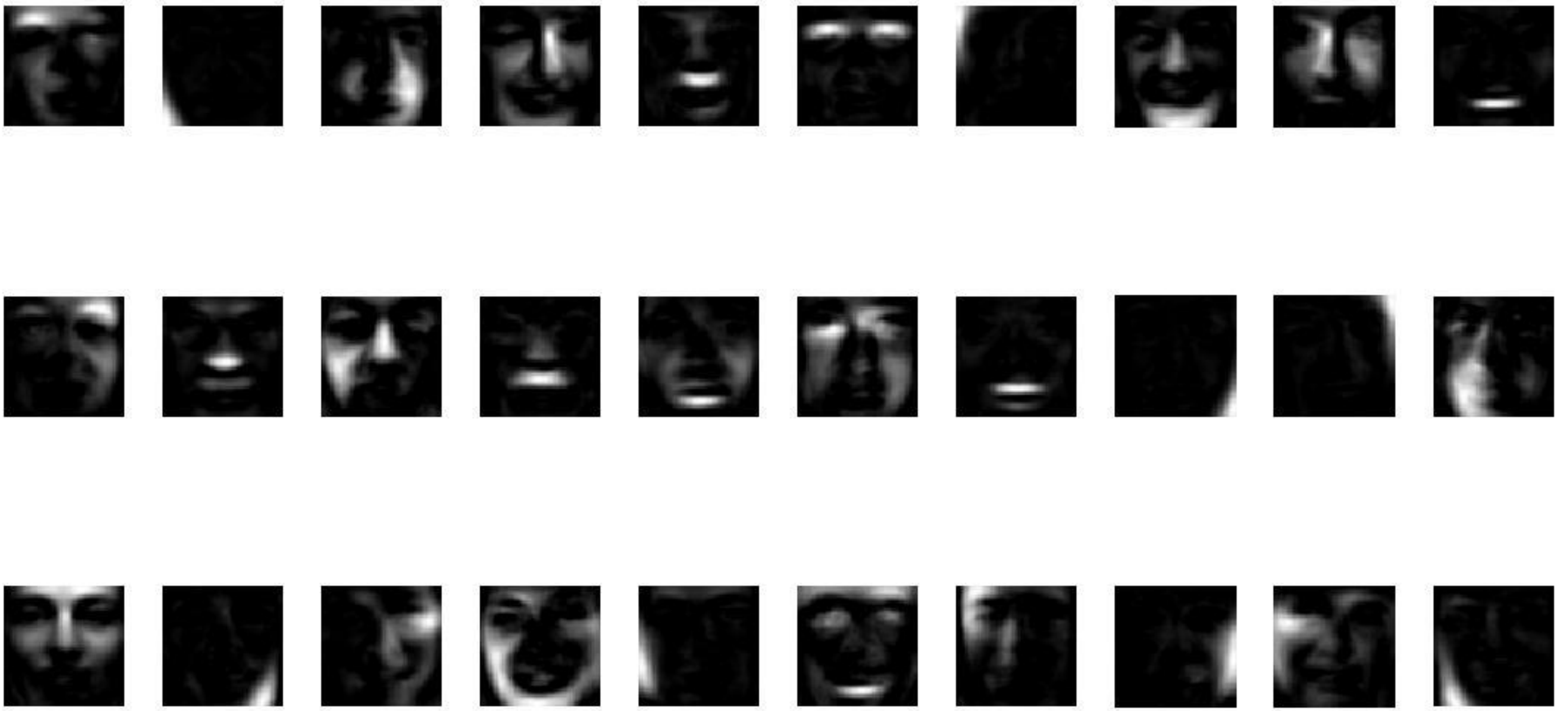
Can be any real number

- PCA involves adding up and subtracting some components (images)
  - Then the components may not be “parts of digits”
- Non-negative matrix factorization (NMF)
  - Forcing a_1, a_2 …… be non-negative
    - additive combination
  - Forcing w^1, w^2…… be non-negative
    - More like “parts of digits”
- Ref: Daniel D. Lee and H. Sebastian Seung. "Algorithms for non-negative matrix factorization." *Advances in neural information processing systems*. 2001.

NMF on MNIST



NMF on Face



Matrix Factorization

Matrix Factorization

Number in table: number of figures a person has

A
B
C
D
E

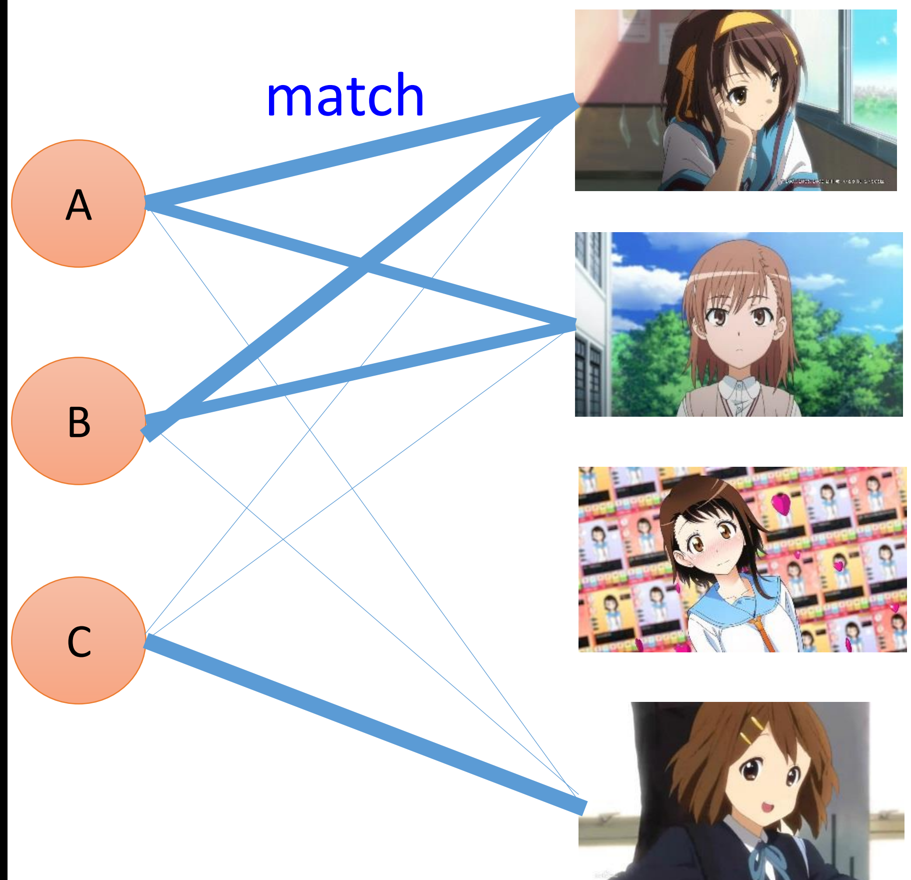
There are some common *factors* behind otakus and characters.

http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/

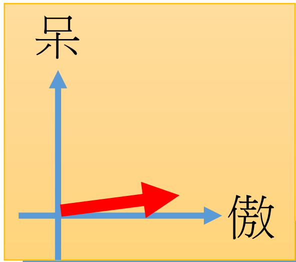
Matrix Factorization

The factors are latent.

No one cares ……



Not directly observable

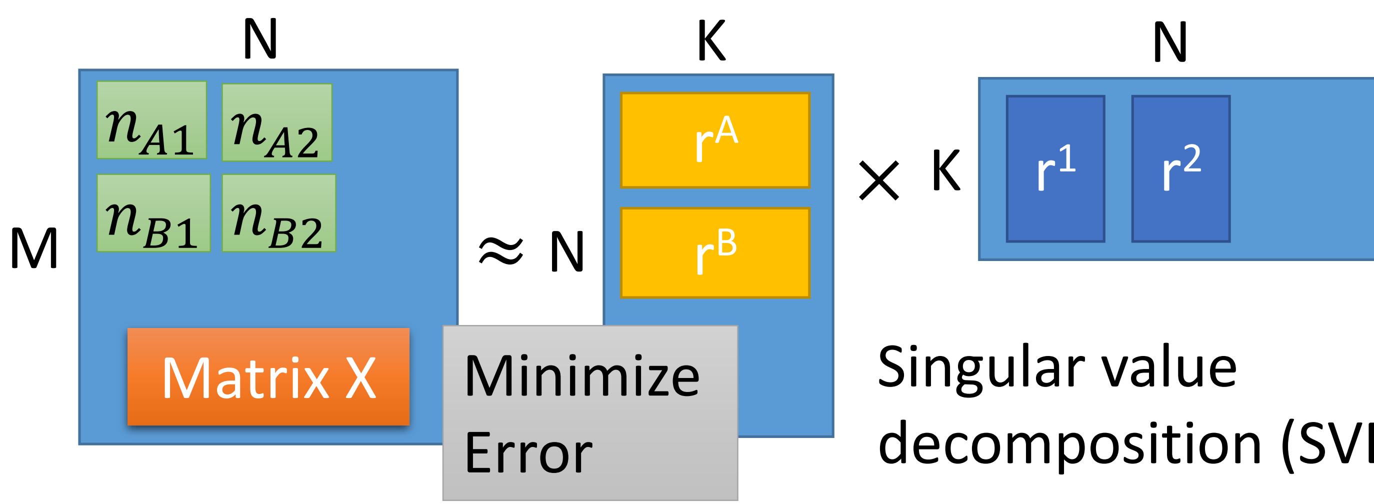


		r^1	r^2	r^3	r^4
r^A	A	5	3	0	1
r^B	B	4	3	0	1
r^C	C	1			5
r^D	D	1	0	4	4
r^E	E	0	1	5	4

Matrix X

No. of Otaku = M No. of characters = N No. of latent factor = K

$$r^A \cdot r^1 \approx 5$$
$$r^B \cdot r^1 \approx 4$$
$$r^C \cdot r^1 \approx 1$$
$$\vdots$$



Singular value decomposition (SVD)

	r^j	r^1	r^2	r^3	r^4
r^i					
r^A	A	5 n_{A1}	3	?	1
r^B	B	4	3	?	1
r^C	C	1	1	?	5
r^D	D	1	?	4	4
r^E	E	?	1	5	4

$$r^A \cdot r^1 \approx 5$$

$$r^B \cdot r^1 \approx 4$$

$$r^C \cdot r^1 \approx 1$$

$$\vdots$$

Minimizing

Only considering the defined value

$$L = \boxed{\sum_{(i,j)}} \left(r^i \cdot r^j - n_{ij} \right)^2$$

Find r^i and r^j by gradient descent

		r^1 	r^2 	r^3 	r^4 
r^A	A	5 n_{A1}	3	-0.4	1
r^B	B	4	3	-0.3	1
r^C	C	1	1	2.2	5
r^D	D	1	0.6	4	4
r^E	E	0.1	1	5	4

Assume the dimensions of r are all 2 (there are two factors)

A	0.2	2.1
B	0.2	1.8
C	1.3	0.7
D	1.9	0.2
E	2.2	0.0

1 (春日)	0.0	2.2
2 (炮姐)	0.1	1.5
3 (姐寺)	1.9	-0.3
4 (小唯)	2.2	0.5

More about Matrix Factorization

- Considering the induvial characteristics

$$r^A \cdot r^1 \approx 5 \quad \Rightarrow \quad r^A \cdot r^1 + b_A + b_1 \approx 5$$

b_A: otakus A likes to buy figures

b_1: how popular character 1 is

Minimizing $$L = \sum_{(i,j)} \left(r^i \cdot r^j + b_i + b_j - n_{ij} \right)^2$$

Find r^i, r^j, b_i, b_j by gradient descent (can add regularization)

- Ref: Matrix Factorization Techniques For Recommender Systems

Matrix Factorization for Topic analysis

character→document, otakus→word

- Latent semantic analysis (LSA)

	Doc 1	Doc 2	Doc 3	Doc 4
投資	5	3	0	1
股票	4	0	0	1
總統	1	1	0	5
選舉	1	0	0	4
立委	0	1	5	4

Number in Table:

Term frequency (weighted by inverse document frequency)

Latent factors are topics (財經、政治 ……)

- Probability latent semantic analysis (PLSA)
  - Thomas Hofmann, Probabilistic Latent Semantic Indexing, SIGIR, 1999
- latent Dirichlet allocation (LDA)
  - Blei, David M.; Ng, Andrew Y.; Jordan, Michael I (January 2003). Lafferty, John, ed. "Latent Dirichlet Allocation". Journal of Machine Learning Research. 3 (4–5): pp. 993–1022.

More Related Approaches Not Introduced

- Multidimensional Scaling (MDS) [Alpaydin, Chapter 6.7]
  - Only need distance between objects
- Probabilistic PCA [Bishop, Chapter 12.2]
- Kernel PCA [Bishop, Chapter 12.3]
  - non-linear version of PCA
- Canonical Correlation Analysis (CCA) [Alpaydin, Chapter 6.9]
- Independent Component Analysis (ICA)
  - Ref: http://cis.legacy.ics.tkk.fi/aapo/papers/IJCNN99_tutorialweb/
- Linear Discriminant Analysis (LDA) [Alpaydin, Chapter 6.8]
  - Supervised

Acknowledgement

- 感謝 彭冲 同學發現引用資料的錯誤
- 感謝 Hsiang-Chih Cheng 同學發現投影片上的錯誤