# Lecture 11: Dimensionality Reduction and Subspace Methods

DD2421

Atsuto Maki

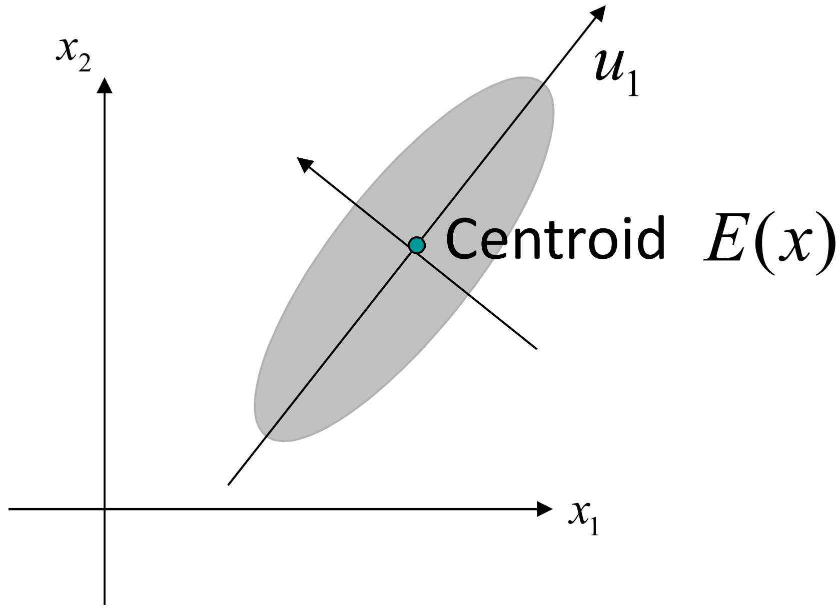
Autumn, 2018

# Our keywords today:

- Dimensionality reduction
  - Principal Component Analysis (PCA)
- Discriminant function
  - Similarity measures: angle, projection length
- Subspace Methods

# Principal Component Analysis (PCA)

## 1. Maximizing variance




Number of samples

Mean vector of $x$ : $E(x) = (1/r)\sum x$

Covariance matrix: $\Sigma = E((x - E(x))(x - E(x))^T)$

# 1. Maximum variance criterion

Reduce the effective number of variables

(only dealing with components with larger variances)

$$\boxed{E((x^T u_i - E(x^T u_i))^2)} \rightarrow \text{Maximize} \; (i = 1, \ldots, p)$$

$$= E((u_i^T (x - E(x)))^2)$$

$$= u_i^T \underbrace{E((x - E(x))(x - E(x))^T)}_{\text{Covariance matrix}} u_i = u_i^T \Sigma u_i$$

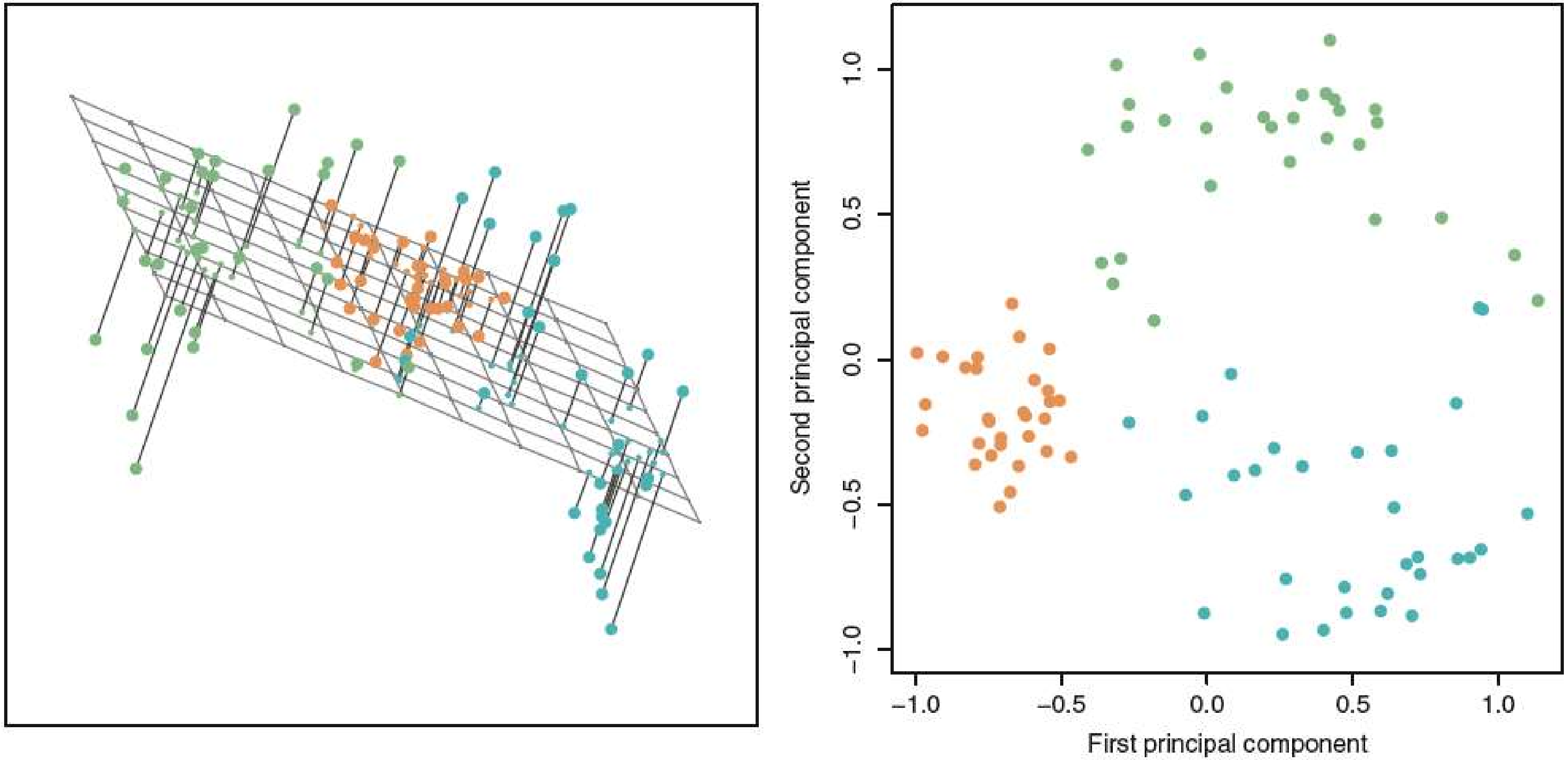
Condition:

$$u_i^T u_j = \delta_{ij}$$

$$\boxed{\max[\mathrm{tr}(U^T \Sigma U)]}$$

The transformation matrix $U$ consists of $p$ columns: the eigenvectors of the covariance matrix, $\Sigma$ (corresponding to its $p$ largest eigenvalues).

# Example 3-d to 2-d: Ninety observations simulated in 3-d

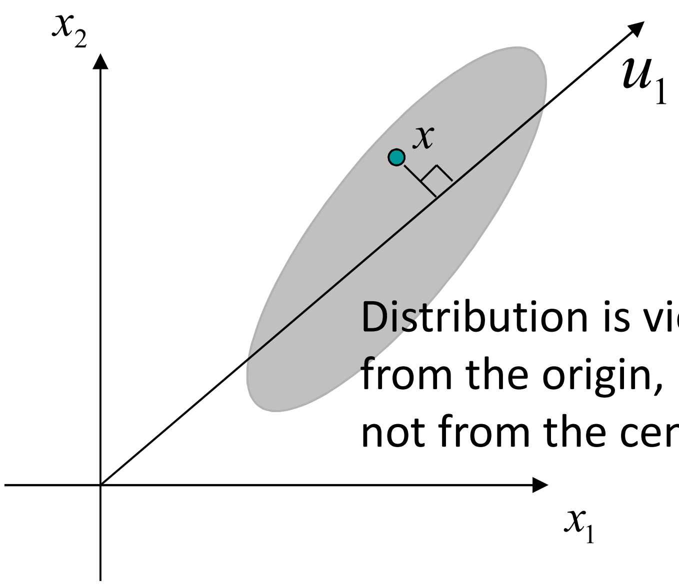



The first 2 principal component directions span the plane that best fits the data.
It minimizes the sum of squared distances from each point to the plane.

Figure from
An Introduction to Statistical Learning (James et al.)

# Principal Component Analysis (PCA)

## 2. Min. approximation error

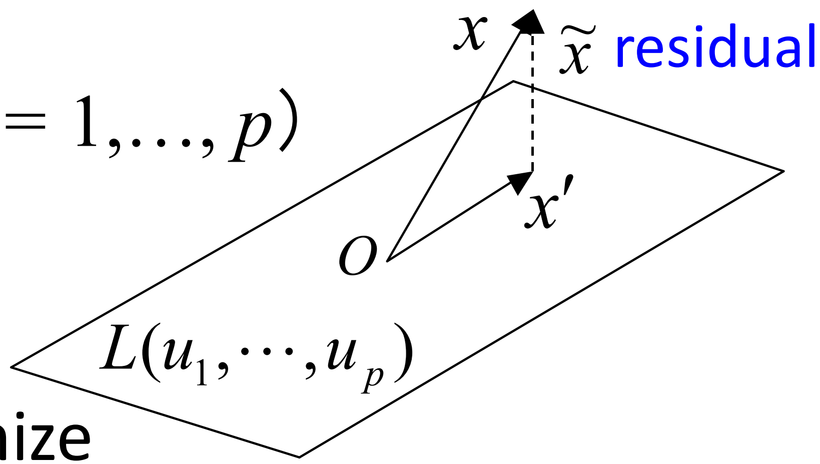



Distribution is viewed
from the origin,
not from the centroid

Autocorrelation matrix: $Q = E(xx^T)$

# 2. Minimum squared distance criterion

Averaged squared error between $x$ and its approximation to be minimized by a set $\{u_1, \cdots, u_p\}$



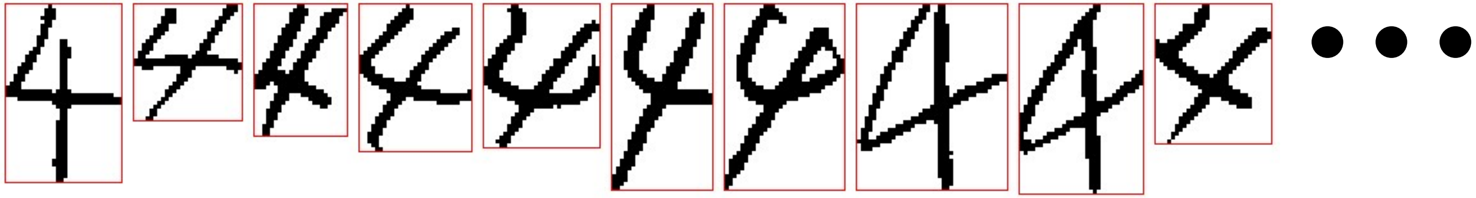

$E(\| x - x' \|^2)$ →minimize $(i = 1, \ldots, p)$

Approximated $x' = \sum_{i=1}^{p} (x^T u_i) u_i$

$\| x' \|^2 = \| x \|^2 - \| \widetilde{x} \|^2$ →maximize

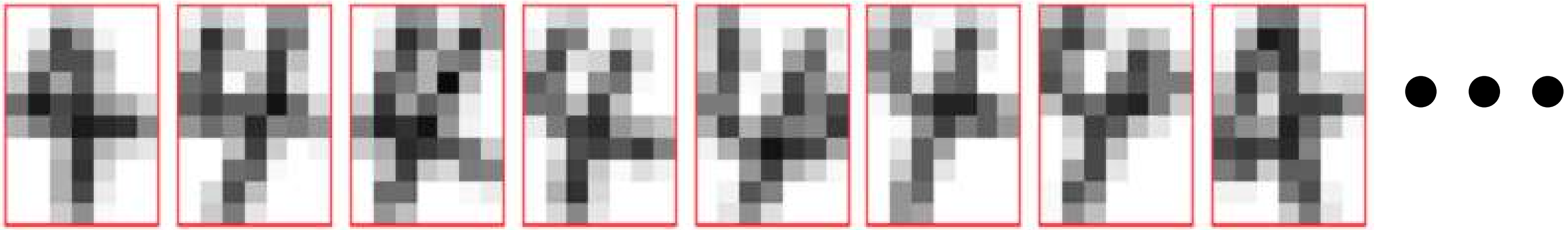
The basis consists of $p$ eigenvectors of the autocorrelation matrix, $Q$ (corresponding to its $p$ largest eigenvalues).

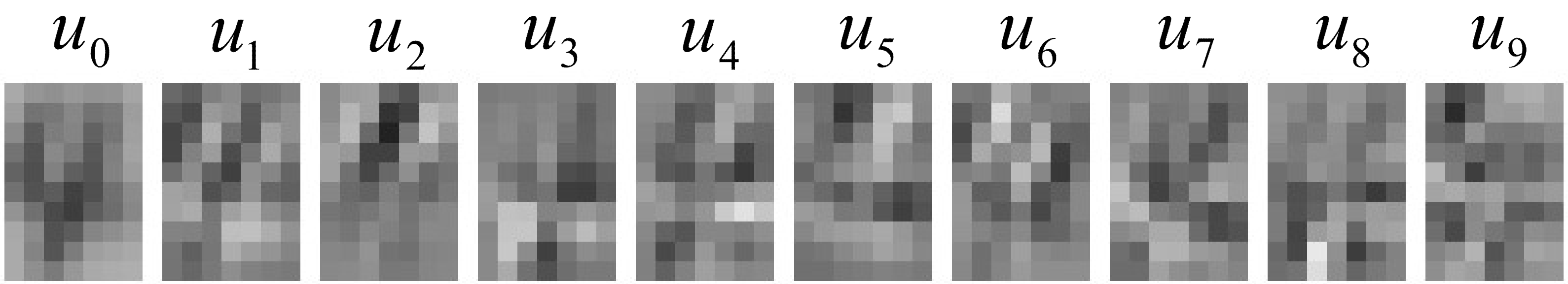
# PCA example 1: Hand-written digits



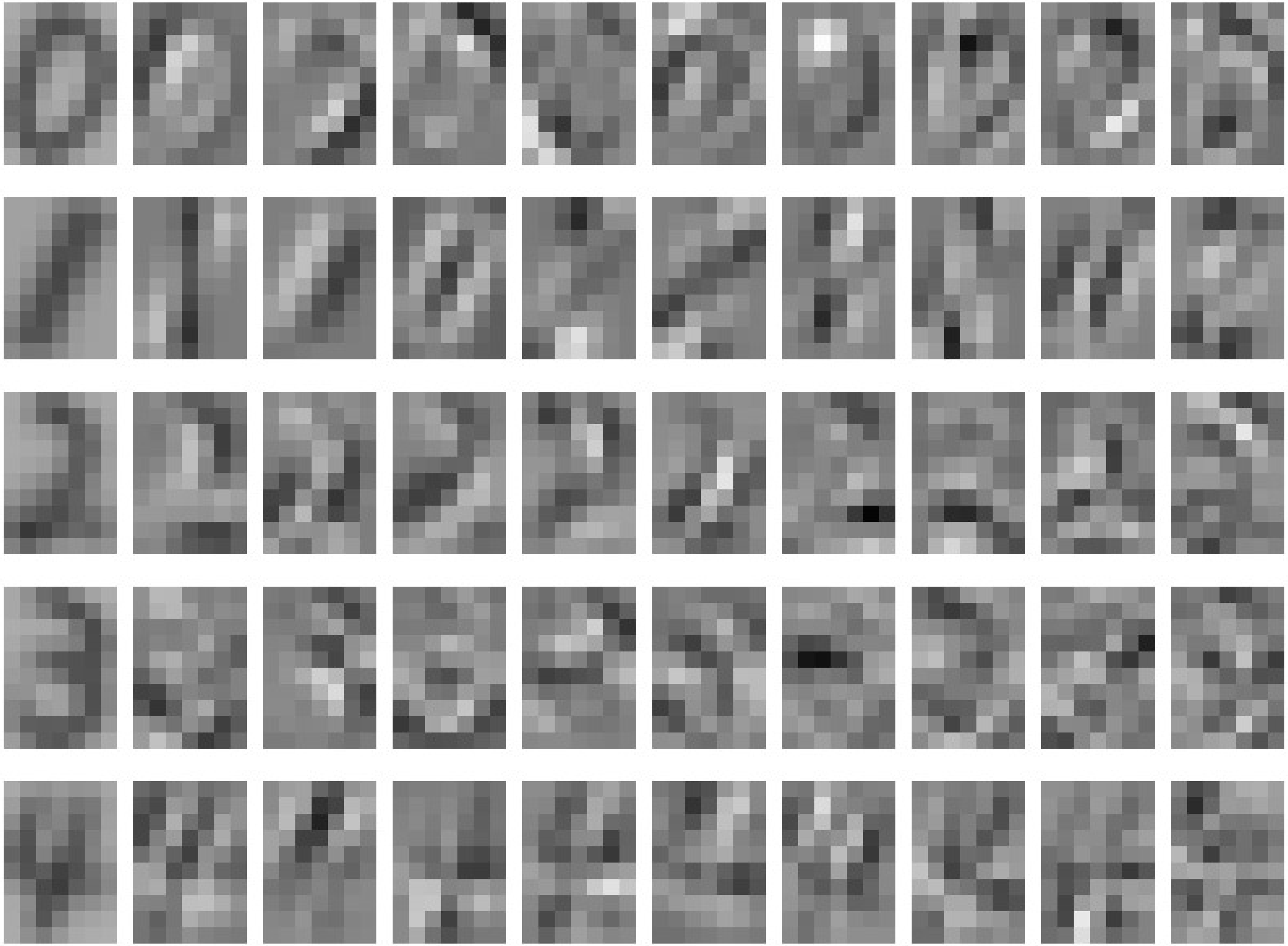
Feature extraction

Pattern vectors: normalized & blurred patterns





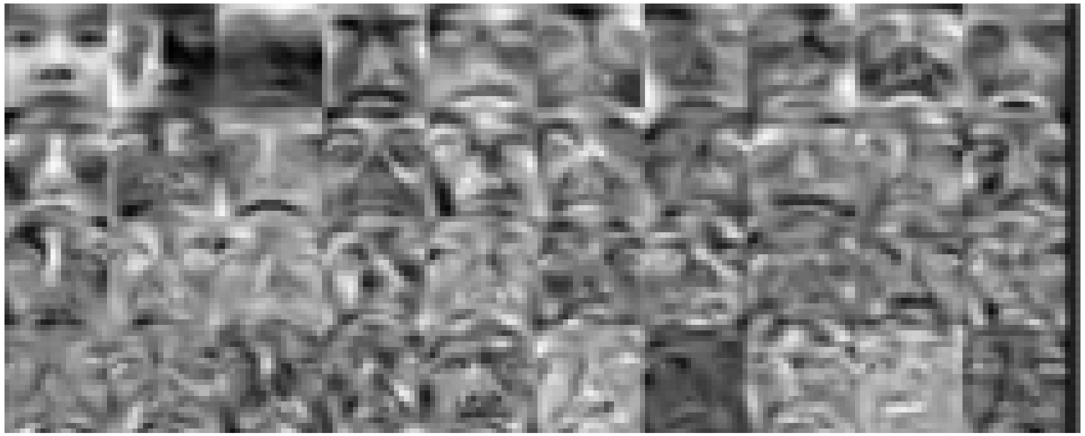
(figure credit: Y. Kurosawa)



Numeral Characters(0 - 4) from 79660 patterns (figure credit: Y. Kurosawa)

# Example 2: Human face classificaiton

Basis vectors of a person: someone's *dictionary*



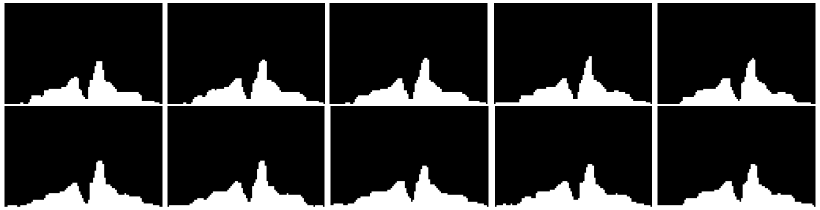
…

(Eigenvectors from a large collection of his/her face)

(figure credit: K. Fukui)

# Example 3: Ship classification (profiles)

Profile vectors

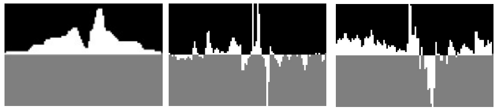


Principal Component Analysis (PCA)

↓

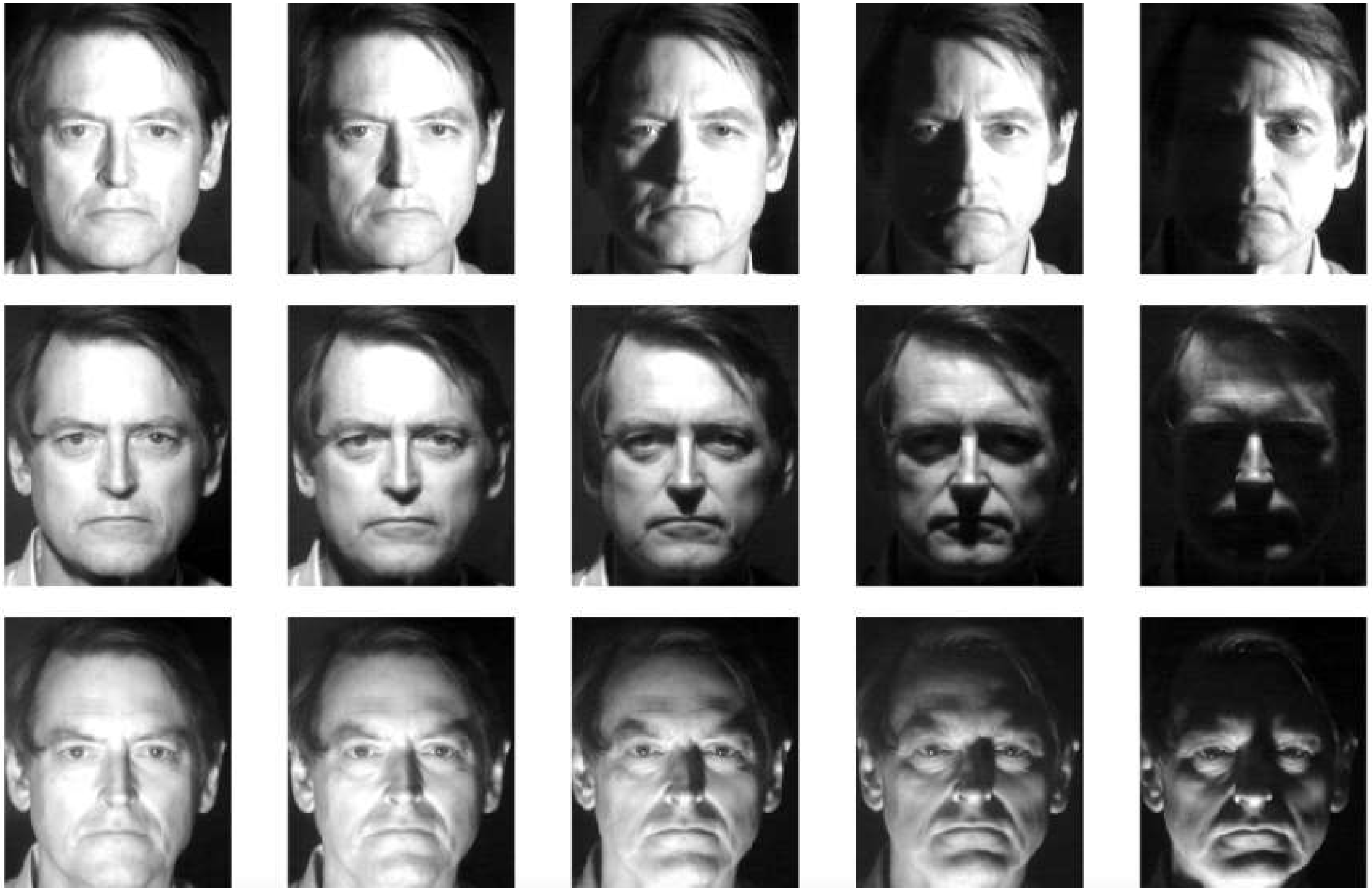
Eigenvectors

Eigenvectors for the greatest eigenvalues



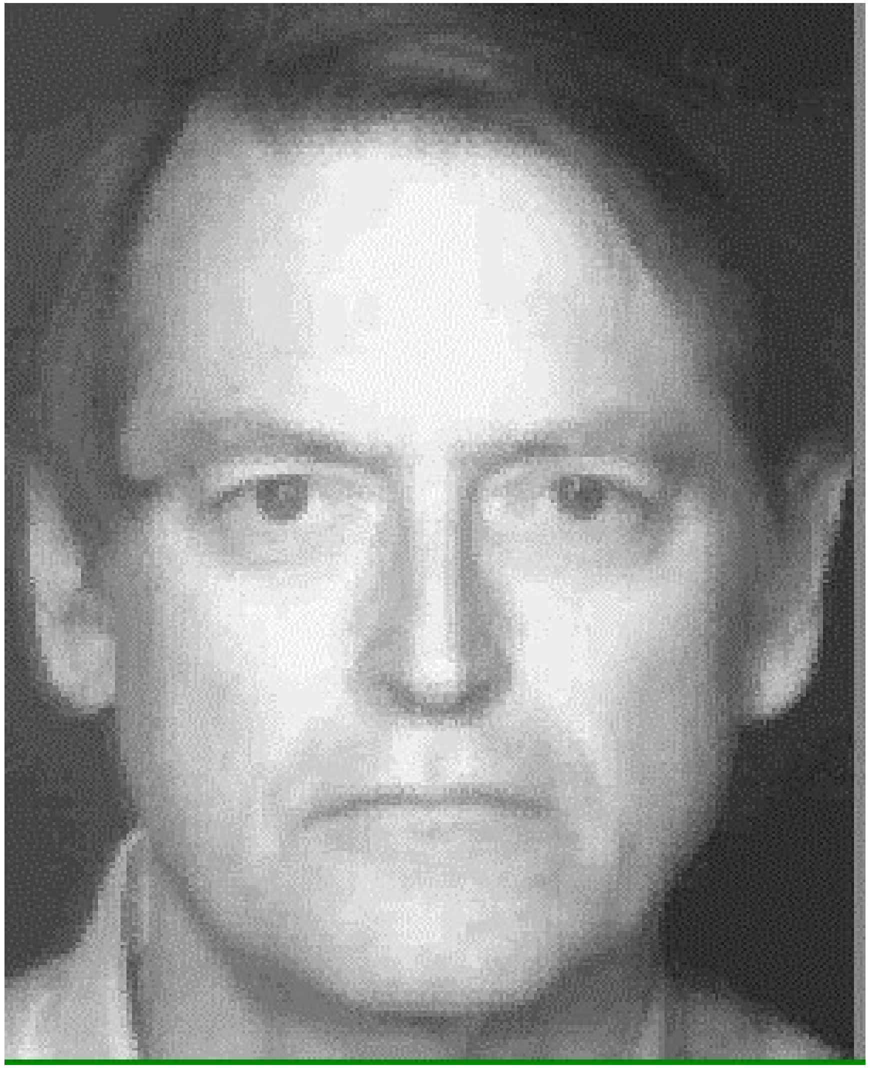
…

# What is the set of images of an object under all possible lighting conditions?



(Harvard Database)

# Synthesized by bases: How many?



(Belhumeur and Kriegman 1996)

# Concept of subspace

Subspace $L$ is a collection of $n$-d vectors:

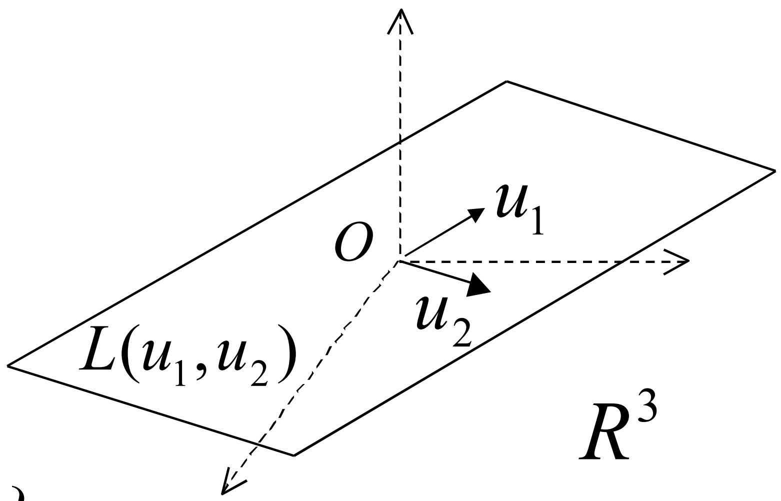
spanned by a basis, a set of linearly independent vectors

$$L(b_1, \cdots, b_p) = \{z \mid z = \sum_{i=1}^{p} \xi_i b_i\} \qquad (\xi_i \in R, b_i \in R^n)$$

Dimension of a subspace:

the number of base vectors

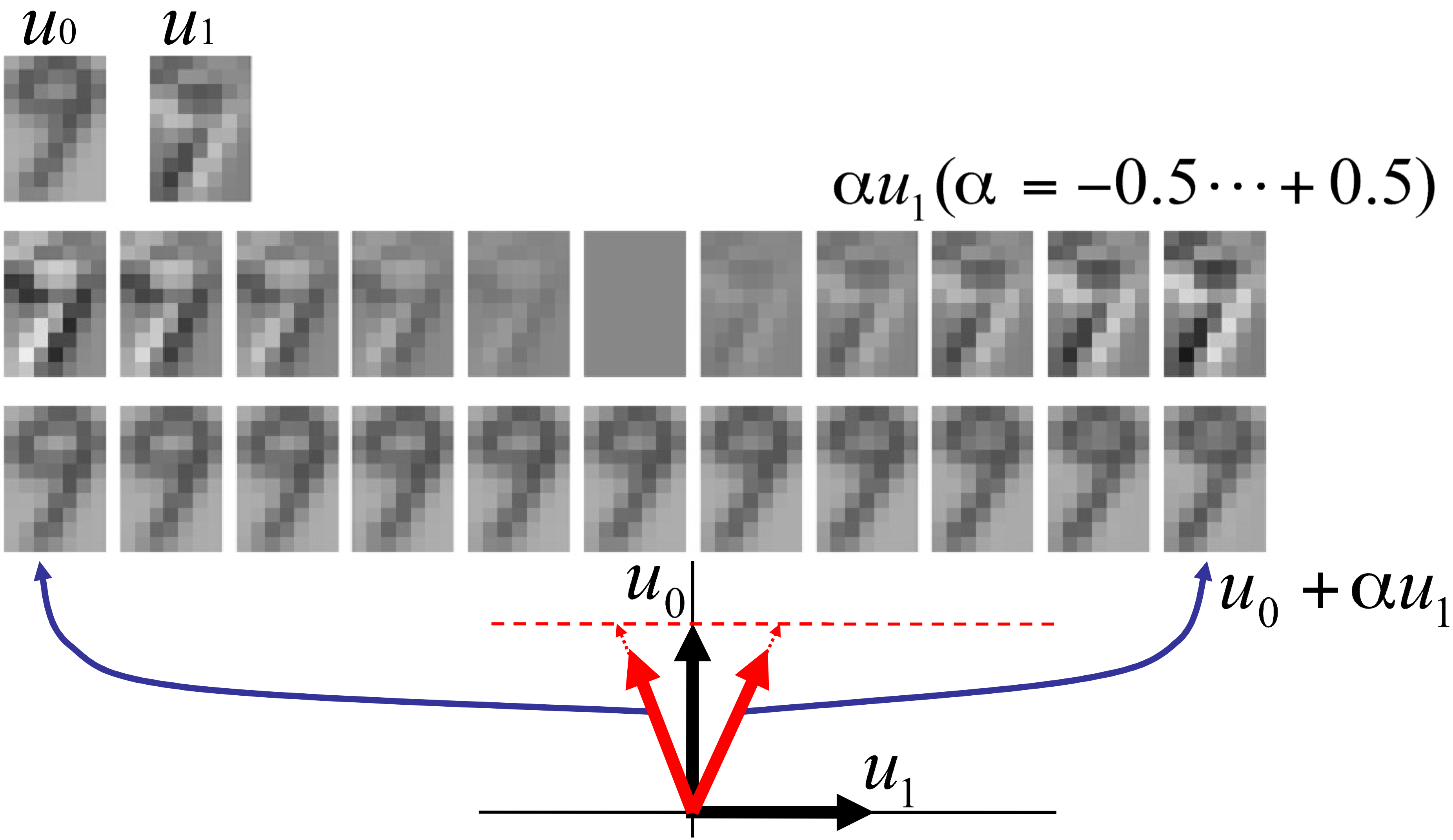
$$p = \dim(L) \qquad << n$$




Conveniently represented

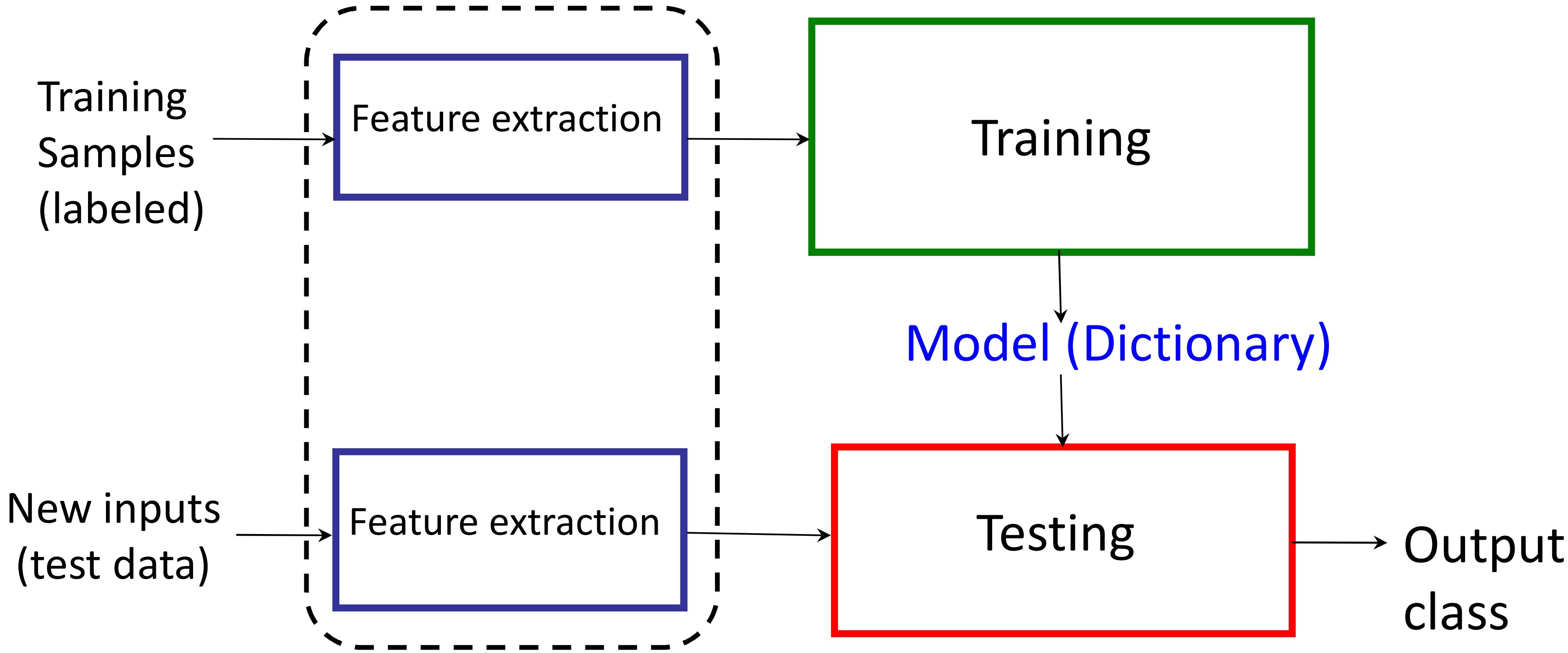
by orthonormal basis $\{u_1, \cdots, u_p\}$

- Variations of “9” covered by a 2-d subspace




(figure credit: Y. Kurosawa)

# Background: Schematic of classification



# Training phase

- Given: Limited number of labeled data

  (samples whose classes are known)
- The dimensionality often too high for limited number of samples

One approach to this is to find redundant variables and discard them, i.e. *dimensionality reduction*

(without losing essential information)

Information compression: to extract the class characteristics and throw away the rest!

# Testing phase

- Various ways to measure the distance
  - Euclidean / Mahalanobis distance
  - Angle between vectors
  - Projection length on subspaces

    …

- Classification methods
  - Discriminant function
  - Subspace method

    …

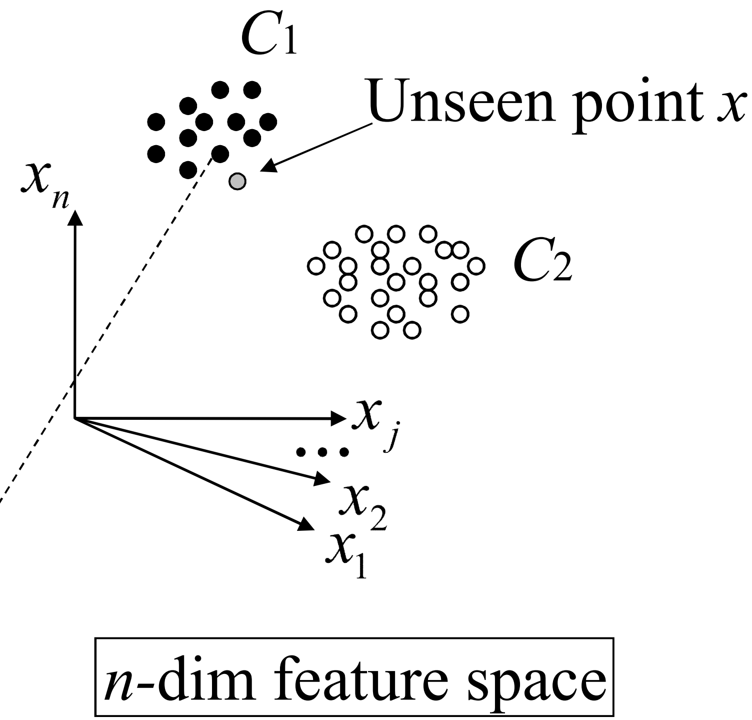
# Nearest Neighbor methods (revisiting)

- Binary classification

  - $N_1$ samples of class $C_1$
  - $N_2$ samples of class $C_2$

  - Unseen data $x$

  → Compute distances to $N_1 + N_2$ samples



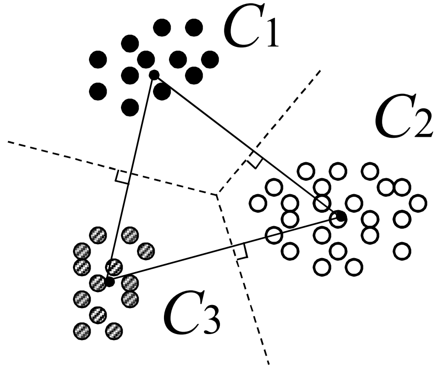

- Find the nearest neighbour

→ classify $x$ to the same class

# Discriminant function

- Need to remember all the samples?
  - In $k$-NN we simply used all the training data
  - Still cover only a small portion of possible patterns

- Define a class by a few representative patterns
  - e.g. the centroid of class distribution




Extreme case: one vector per class

# Formulation: one prototype per class

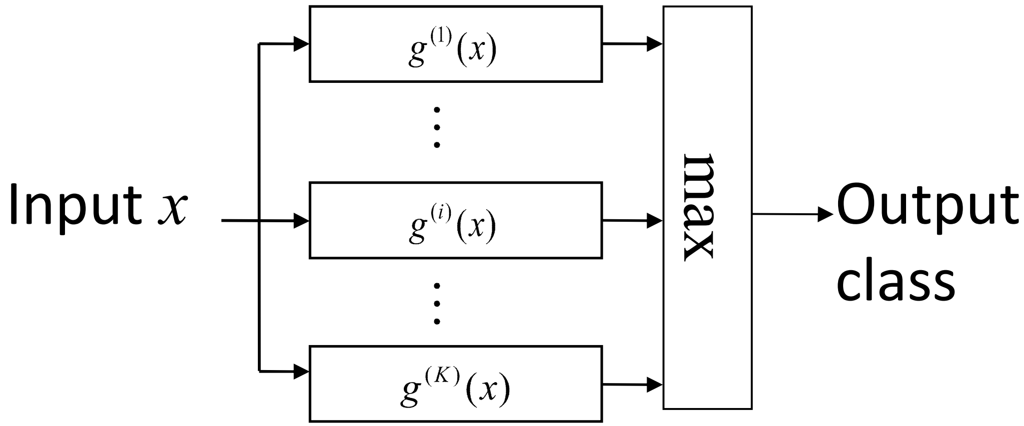
- $K$ classes: $C^{(1)}, \cdots, C^{(K)}$
- $K$ prototypes: $a^{(1)}, \cdots, a^{(K)}$

Consider Euclidean distances between the new input $x$ and the prototypes: $\| x - a^{(i)} \|^2 = \| x \|^2 - 2a^{(i)^T} x + \| a^{(i)} \|^2$

→ Choose the class that minimises the distance.

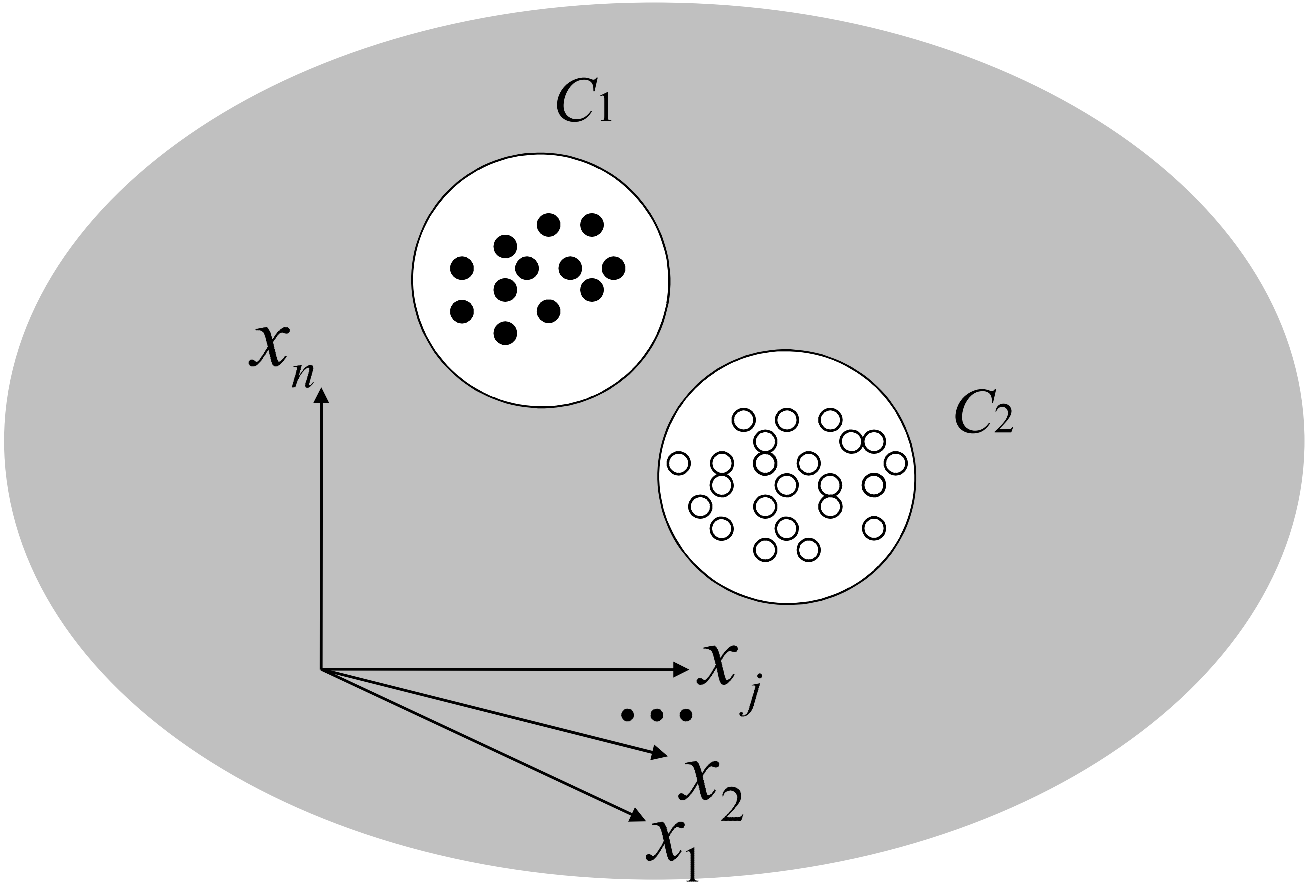
<u>Discriminant function</u>

$$g^{(i)}(x) \equiv a^{(i)^T} x - \frac{1}{2} \| a^{(i)} \|^2$$



# Setting the "don't know" category

- Reject if the distance is above the threshold



# Direction cosine as similarity

Think of the new input and the prototype as vectors.

Compute cosine between the input vector $x$ and vector $a^{(i)}$

$$g^{(i)}(x) = \frac{(x^T a^{(i)})}{\| x \| \, \| a^{(i)} \|} = \cos A$$

"Simple similarity"

$0 \leq \cos^2 A \leq 1$ (The closer it is to 1, the more likely to be in $C^{(i)}$)

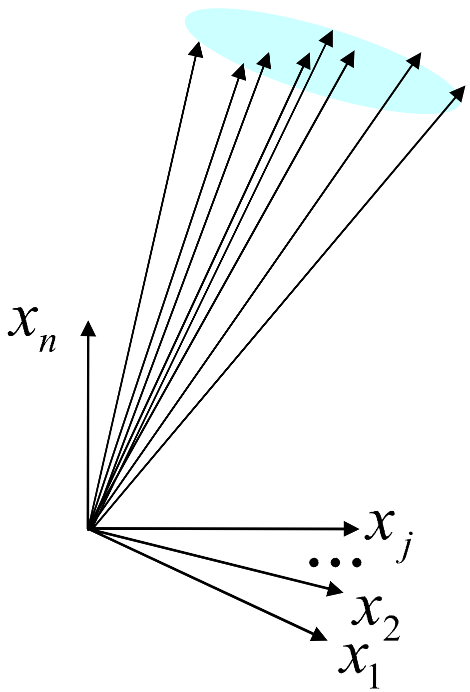
Now let's extend the class representative to

a set of basis vectors $\longrightarrow$ spans a subspace

# Subspace Methods

- Exploit localization of pattern distributions

Samples in the same class such as a digit (or face images of a person) are similar to each other.
They are <u>localized</u> in a *subspace* spanned by <u>a set of basis</u> $u_i$ .




$u_i$ : reference vectors
(orthonormal basis)

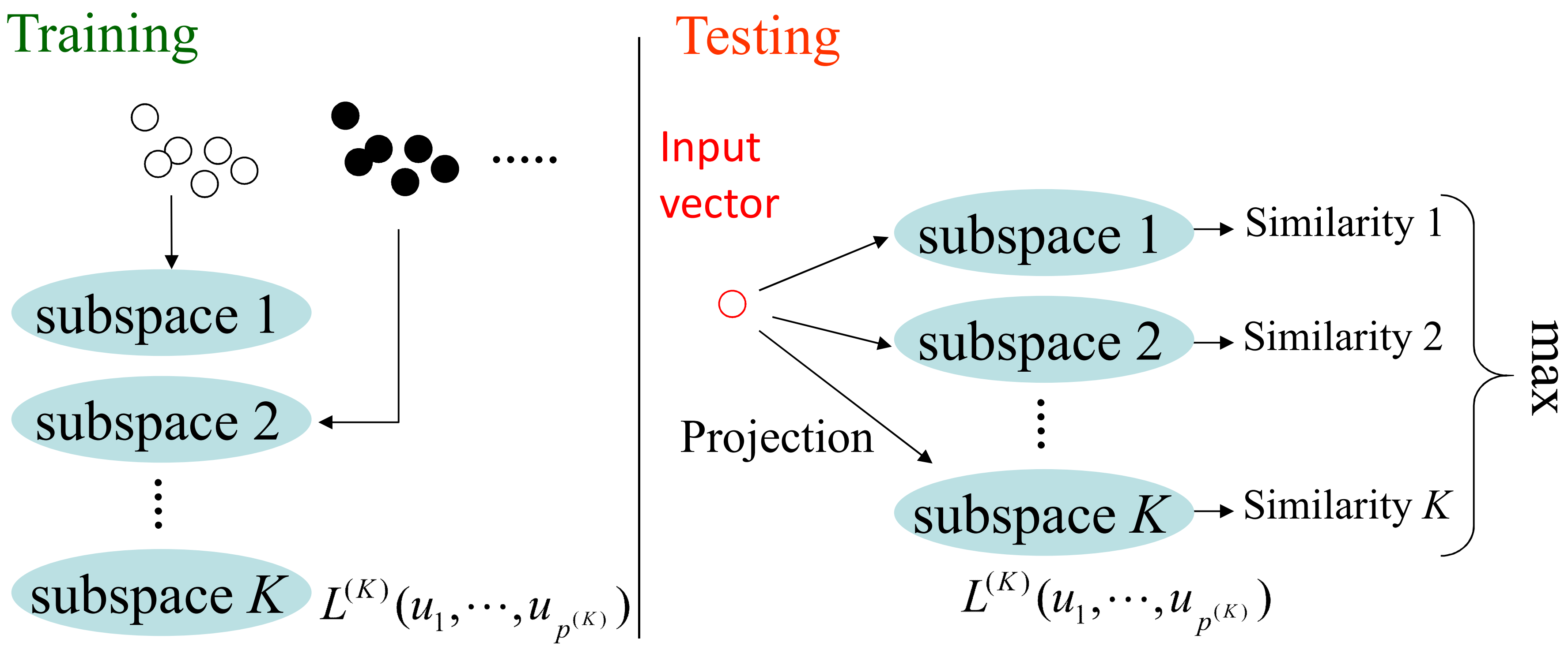
a.k.a. CLAFIC
CLAss-Featuring Information Compression

# Framework of Subspace Method

1. Training: for each class, compute a low-dimensional subspace that represents the distribution in the class.

$$\omega^{(1)}, \cdots, \omega^{(K)}$$

2. Testing: determine the class of new unknown input by comparing which subspace best approximates the input.



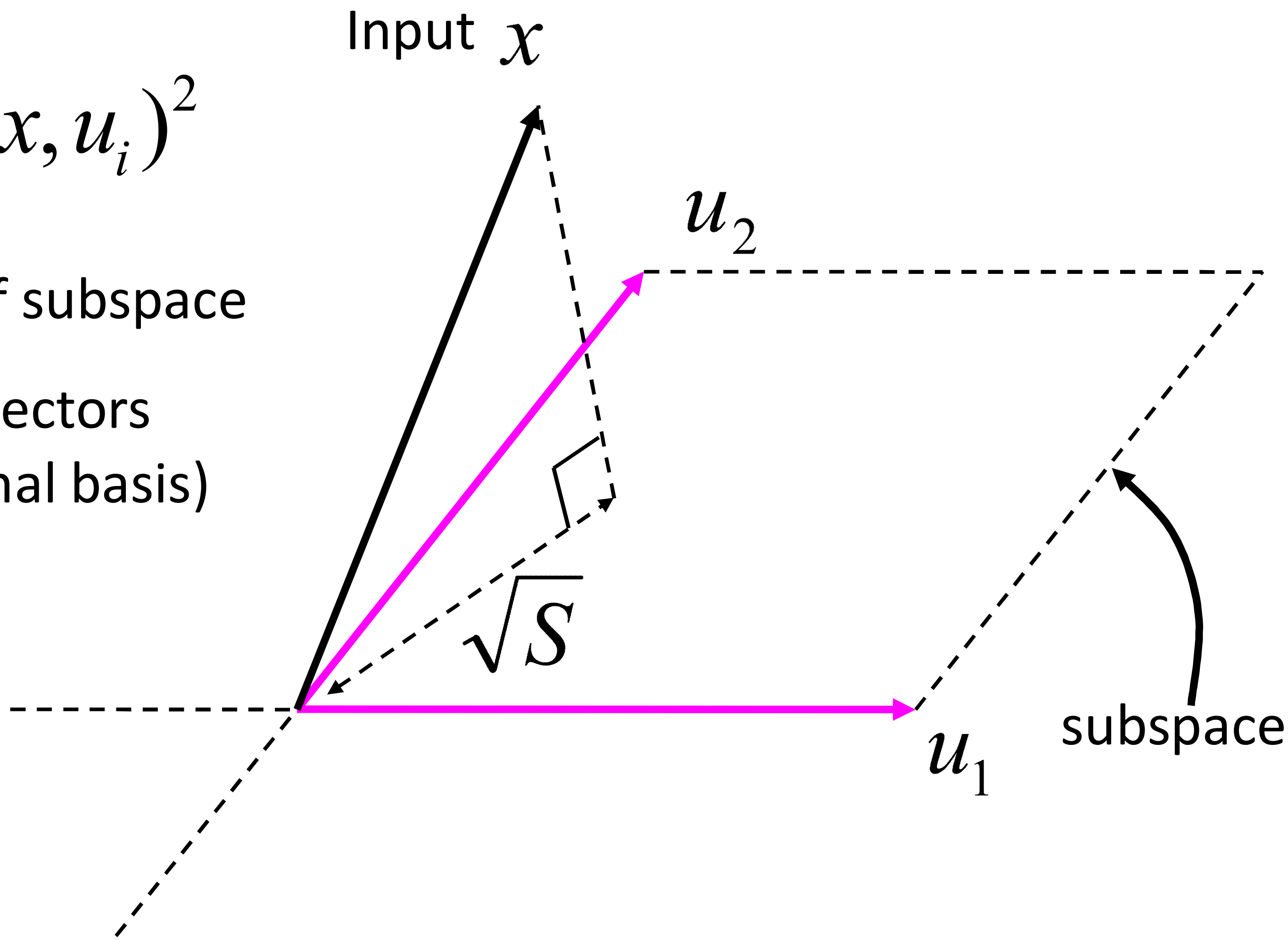
# Similarity in Subspace Method

## Projection length to the subspace

$$S = \sum_{i=1}^{p} (x, u_i)^2$$

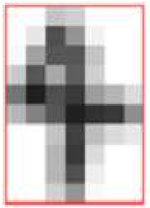
$p$: dimension of subspace

$u_i$ : reference vectors
(orthonormal basis)



# Similarity in Subspace Method (example)

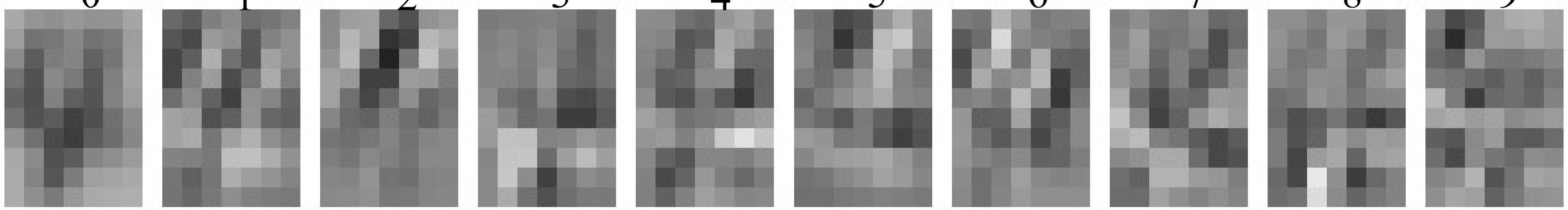
## Projection length to the subspace



$$S = \sum_{i=0}^{p-1} (x, u_i)^2$$

$p$: the dimensionality of class subspace
(can be determined for each class, how?)

$u_0$ $u_1$ $u_2$ $u_3$ $u_4$ $u_5$ $u_6$ $u_7$ $u_8$ $u_9$



# Dimensionality of a class subspace

Eigenvalues of autocorrelation matrix *Q*: $\lambda_1 \geq \ldots \lambda_j \ldots \geq \lambda_p \geq 0$

The number of dimensions to be used for each class:

- Too low → low capability to represent the class
- Too high→ issue of overlapping across classes

• Cumulative contributions

$$a(p^{(i)}) = \frac{\sum_{j=1}^{P^{(i)}} \lambda_j}{\sum_{j=1}^{p} \lambda_j}$$

Choose a dimension $p^{(i)}$ for each class $\omega^{(i)}$

$a(p^{(i)}) \leq \kappa \leq a(p^{(i)}+1)$（$\kappa$: common value）

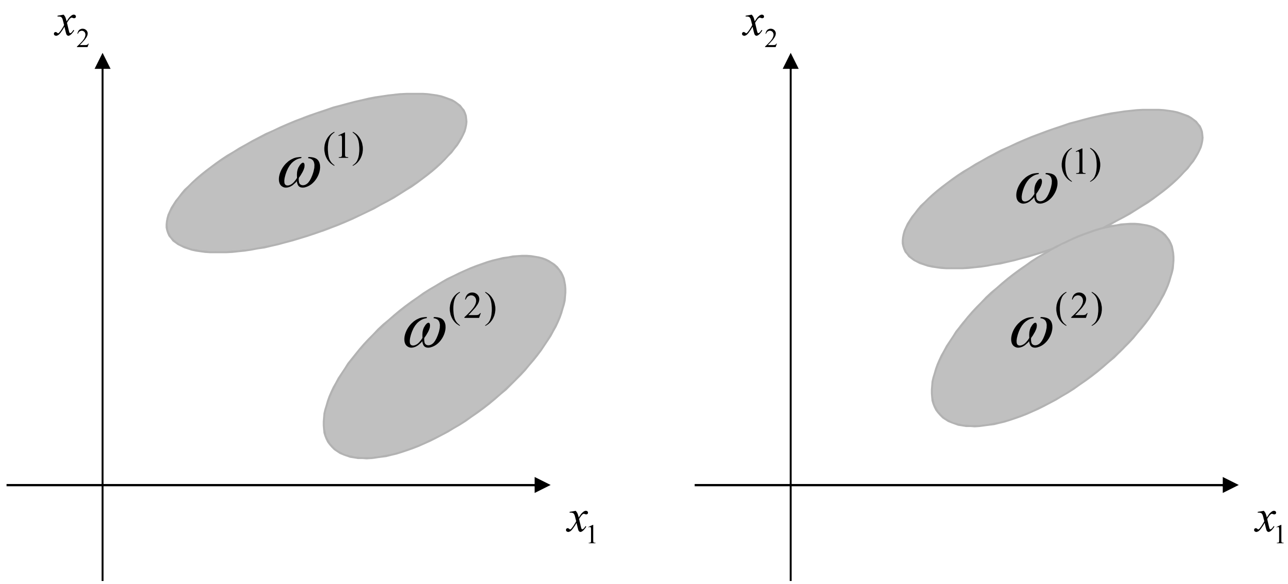
The projection length to the subspace is made uniform.

Experiments still needed to find a good dimensionality

# What is a good dimension (direction) for classification, given labels?

Ideal distributions of input pattern vectors:

- Patterns from an identical class be close
- Patterns from different classes be apart




→ Overlapping distributions harmful for classification

# Ratio of between-classes variance to within-class variance

Within-class variance

Average in class $\omega^{(i)}$

$$\sigma_W^2 = \frac{1}{r}\sum_{i=1}^{K}\sum_{x\in\omega^{(i)}}(x-E^{(i)}(x))^T(x-E^{(i)}(x))$$

Total # of samples

Average overall

Between-class variance

$$\sigma_B^2 = \frac{1}{r}\sum_{i=1}^{K} r^{(i)}(E^{(i)}(x)-E(x))^T(E^{(i)}(x)-E(x))$$

Number of samples in class $\omega^{(i)}$

## Within-class var. between-class var. ratio

$$J_\sigma = \frac{\sigma_B^2}{\sigma_W^2}$$

$\frac{\text{Between-class variance}}{\text{Within-class var in ave}}$

In short: distance between classes normalized by distance within class

→ the larger the better!

# Fisher's method

Find a subspace most suitable to classification

（discriminant analysis）

Given pattern distributions in 2 classes

⇒ Optimal axis direction where $J$ is maximized

Scatter matrix represents variation within class

$$S_i \equiv \sum_{x \in \omega^{(i)}} (x - E^{(i)}(x))(x - E^{(i)}(x))^T$$

Within-class: $S_W \equiv S_1 + S_2$

Between-classes: $S_B \equiv \sum_{i=1,2} r^{(i)} (E^{(i)}(x) - E(x))(E^{(i)}(x) - E(x))^T$

$$\ldots = \frac{r^{(1)} r^{(2)}}{r} (E^{(1)}(x) - E^{(2)}(x))(E^{(1)}(x) - E^{(2)}(x))^T$$

From $n$-d feature space to $1$-d space by Matrix $A$

$A$ is an $n$ x $1$ matrix → $n$-dim vector $a$ in practice

→ The pattern will become a scalar by $y = A^T x$

Scatter matrix in the space after the transformation:

$$\hat{S}_i \equiv \sum_{x \in \omega^{(i)}} (y - E^{(i)}(y))(y - E^{(i)}(y))^T$$

$$= \sum_{y \in \omega^{(i)}} A^T (x - E^{(i)}(x))(x - E^{(i)}(x))^T A = A^T S_i A$$

Within-class: $\hat{S}_W \equiv \hat{S}_1 + \hat{S}_2 = A^T S_1 A + A^T S_2 A = A^T S_W A$

Between-class: $\hat{S}_B \equiv \sum_{i=1,2} r^{(i)} (E^{(i)}(y) - E(y))^2$

Scalar

$$\ldots = \frac{r^{(1)} r^{(2)}}{r} A^T (E^{(1)}(x) - E^{(2)}(x))^2 A = A^T S_B A$$

## Fisher's criterion:

$$J_S(A) \equiv \frac{\hat{S}_B}{\hat{S}_W} = \frac{A^T S_B A}{A^T S_W A}$$

Maximizing the ratio of between-classes variance to within-class variance

Lagrange multiplier

$$J(a) \equiv a^T S_B a - \lambda(a^T S_W a - I)$$ →Maximize

Condition: $\hat{S}_W = I$

$$S_B a = \lambda S_W a$$

$$\Leftrightarrow S_W^{-1} S_B a = \lambda a$$

$$\Leftrightarrow \max\{J_S(a)\} = \lambda_1$$ The greatest eigenvalue of $S_W^{-1} S_B$

→ The eigenvector for the greatest eigenvalue of $S_W^{-1} S_B$ gives $A$ that maximises Fisher's criterion