EEEN4/60151: Machine Learning & Opti…

Hujun Yin

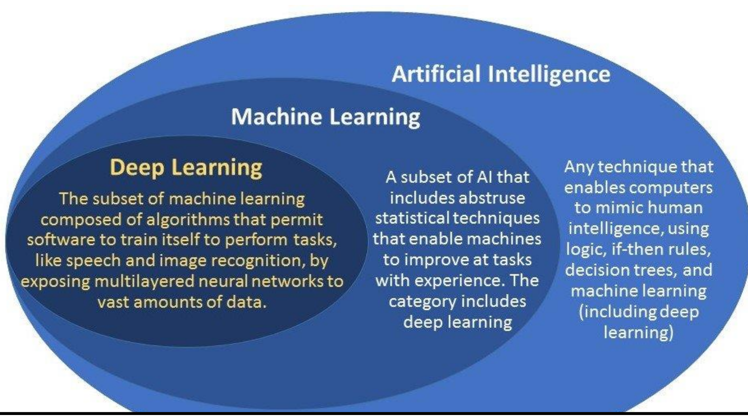
1. Introduction
2. Fundamentals
3. Clustering and mixture models
4. Decision tree learning
5. Classification, Bayes theory, SVM
6. Neural networks
7. Introduction to deep learning

1

EEEN4/60151: Machine Learning & Opti…

Hujun Yin

Part 2: Fundamentals



2

EEEN4/60151: Machine Learning & Opti...

Hujun Yin

Part 2: Fundamentals

- Random variables, probabilities, random processes
- Gradient descent & stochastic gradient descent
- Least-squares method
- Principal component analysis

3

2.0 Variables & Random Variables

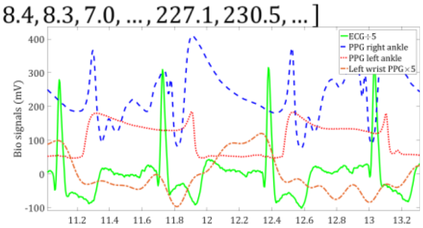
- **Variables** (variables, vectors, matrices)

Clinic trial of effectiveness of a treatment

Center	Status	Response / Treatment	Poor	Moderate	Excellent
1	1	Active	3	20	5
1	1	Placebo	11	14	8
1	2	Active	3	14	12
1	2	Placebo	6	13	5
2	1	Active	12	12	0
2	1	Placebo	11	10	0
2	2	Active	3	9	4
2	2	Placebo	6	9	3

A face image (146x216 pixels)

$$ECG_5 = [7.3, 8.4, 8.3, 7.0, \dots, 227.1, 230.5, \dots]$$

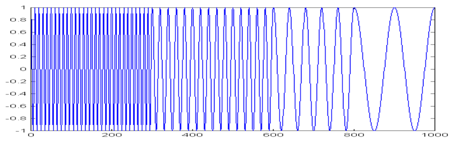


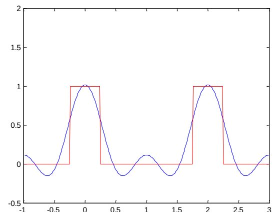
$$A = \begin{bmatrix} 201 & 198 \dots & 167 \\ 200 & 199 \dots & 170 \\ \dots & \dots \dots & \dots \end{bmatrix}$$

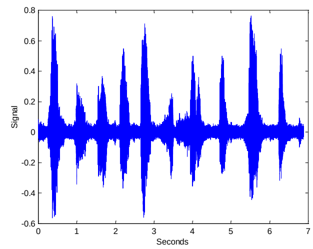
4

2.1 Random Variables & Probabilities

- **Random variables** (in contrast to constant/deterministic)







5

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

A random variable X is a variable with its value depending on its probability $P(x)$ or $Prob(X=x)$, *i.e. the probability of X having value of x.*
Probabilities are always nonnegative.

continuous	$\int P(x)dx = 1$	$P(x) \rightarrow 0$ while $P(x_1<x<x_2)$ usually $\neq 0$
discrete	$\sum_{k=1}^{N} P(x_k) = 1$	$P(x_k)$ usually $\neq 0$ while $x_k \in \{x_1, x_2, \ldots x_N\}$

6

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

Distribution/density

	cumulative distribution function (cdf):	*probability distribution function* (pdf):
continuous	$F(x_0) = \int_{-\infty}^{x_0} P(x)dx$	$p(x_0) = \left.\frac{dF(x)}{dx}\right\|_{x=x_0}$
discrete	$\sum_{k=1}^{N} P(x_k) = 1$	$p(x_k)$ or $P(x_k)$

7

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

Expectation/statistical mean

continuous $\quad E\{X\} = \mu_x = \int xp(x)dx$

discrete $\quad E\{X\} = \mu_x = \sum_{k=1}^{N} x_k P(x_k)$

8

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

  Variance

continuous $$E\{(X-\mu_x)^2\} = \sigma_x^2 = \int (x-\mu_x)^2 p(x)dx$$

discrete $$\sigma_x^2 = \sum_{k=1}^{N} (x_k-\mu_x)^2 P(x_k)$$

9

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

  Covariance of two variables *X* and *Y*

continuous
$$Cov(X,Y) = E\{(X-\mu_x)(Y-\mu_y)\}$$
$$= \iint (x-\mu_x)(y-\mu_y)p(x,y)dxdy$$
$$= E\{XY\} - E\{X\}E\{Y\}$$

discrete $$Cov(X,Y) = \frac{1}{N}\sum_{k=1}^{N} (x_k-\mu_x)(y_k-\mu_y)P_k$$

10

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

  k-th moment/*k*-th central moment

$$E\{X^k\} = \mu_k = \int x^k p(x)dx$$

$$E\{(X-\mu_x)^k\} = \int (x-\mu_x)^k p(x)dx$$

Skewness $\frac{\mu_3}{\sigma^3}$

Kurtosis (normalised) $\frac{\mu_4}{\sigma^4} - 3$

11

2.1 Random Variables & Probabilities

- **Random variables**
  continuous or discrete

  Examples:

  stock prices

  room temperatures

  ……

  number of visitors to …

  number of winning tickets

  outcome of coin tossing

In practice, there are often a constant or deterministic part and a random part in an acquired data value.

12

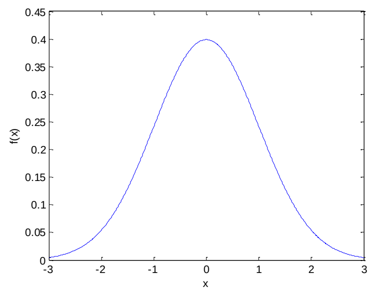
2.1 Random Variables & Probabilities

- **Probability & distribution examples**
  continuous or discrete

Gaussian/normal distribution

$$X \sim \mathcal{N}(\mu_x, \sigma_x)$$

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu_x)^2}{2\sigma^2}}$$



Normal distribution (pdf) with μ=0 and σ=1.

13

2.1 Random Variables & Probabilities

- **Probability & distribution examples**
  continuous or discrete

Bernoulli distribution: *X* discrete: 1 (success) or 0 (failure)
p: *P*(*X*=1); q=(1-p)= *P*(*X*=0)

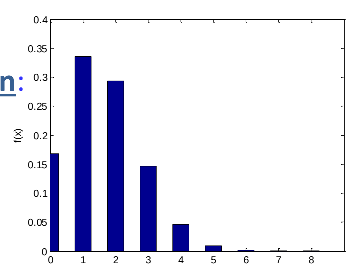
$$p(x) = p^x(1-p)^{1-x}$$

$\mu = p$ and $\sigma^2 = pq$.

Bernoulli trials & binomial distribution:
a sequence of Bernoulli trials.
X=number of successes in n Bernoulli trials (0, 1, … n)

$$p(x) = \binom{n}{x} p^x (1-p)^{n-x}$$

$\mu = np$ and $\sigma^2 = npq$.



Binomial distribution with *p*=0.2 and *n*=8.

14

2.1 Random Variables & Probabilities

- **Joint Probability/Distribution**
  continuous or discrete

  Joint probability of *X* and *Y*

$$P((X,Y) \in A) = \iint_A p(x,y)dxdy$$

joint density, and $\iint p(x,y)dxdy = 1$

Marginal densities/distributions:

$$p_x(x) = \int p(x,y)dy$$

$$p_y(y) = \int p(x,y)dx$$

Exercise: Assume X and Y are two independent random variables and

$$X \sim \mathcal{N}(\mu_x, \sigma_x)$$

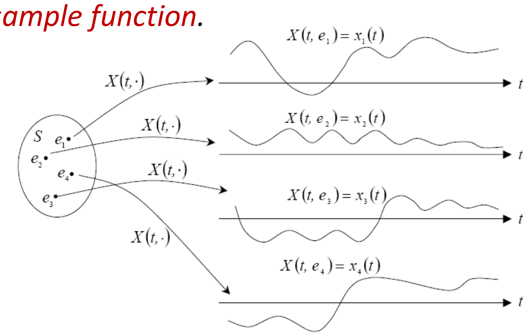
$$Y \sim \mathcal{N}(\mu_y, \sigma_y)$$

What is their joint density?

2.1 Random Variables & Probabilities

- **Random processes** (as contrast to deterministic functions)

A random process $X(t)$ or $X[n]$ is a time varying random variable, or a collection/ensemble (over time) of random variables. At any time instant, t_i or n_i, the random process $x(t)$ or $x[n]$ is a random variable and gives a realisation. Each collection is called a collection or sample function.



2.2 Gradient Descent & Stochastic Gradient Descent

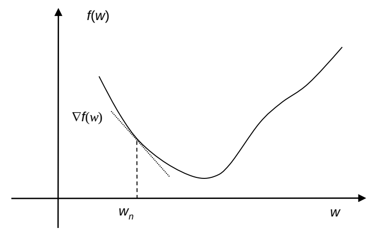
- **Gradient descent**

Optimisation problem: Find solution w to a function *f*(w, *n*)
So that *f*(w, *n*) is minimum (or maximum).

•*Gradient descent method:*

$$w_{n+1} = w_n - \alpha \nabla f(w_n)$$

Step size: $\alpha > 0$



17

2.2 Gradient Descent & Stochastic Gradient Descent

- **Vector gradient**

*If parameter **w** is a vector of p elements:*

$$\mathbf{w} = [w[0], w[1], \ldots, w[p-1]]^T$$

$$\nabla f(\mathbf{w}) \equiv \frac{\partial f(\mathbf{w})}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial f(\mathbf{w})}{\partial w[0]} \\ \frac{\partial f(\mathbf{w})}{\partial w[1]} \\ \vdots \\ \frac{\partial f(\mathbf{w})}{\partial w[p-1]} \end{bmatrix}$$

<u>Example</u>: consider the function of an inner product:

$$f(\mathbf{w}) = \sum_{k=0}^{p-1} a_k w[k] = \mathbf{a}^T \mathbf{w} \qquad \nabla f(\mathbf{w}) \equiv \frac{\partial f(\mathbf{w})}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial f(\mathbf{w})}{\partial w[0]} \\ \frac{\partial f(\mathbf{w})}{\partial w[1]} \\ \vdots \\ \frac{\partial f(\mathbf{w})}{\partial w[p-1]} \end{bmatrix} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{p-1} \end{bmatrix} = \mathbf{a}$$

18

2.2 Gradient Descent & Stochastic Gradient Descent

- **Matrix gradient**

*If parameter **W** is a matrix of pxp elements:*

$$\mathbf{W} = \begin{bmatrix} w[0,0] & \cdots & w[0,p-1] \\ \vdots & \cdots & \vdots \\ w[p-1,0] & \cdots & w[p-1,p-1] \end{bmatrix}$$

$$\nabla f(\mathbf{W}) \equiv \frac{\partial f(\mathbf{W})}{\partial \mathbf{W}} = \begin{bmatrix} \frac{\partial f(\mathbf{w})}{\partial w[0,0]} & \cdots & \frac{\partial f(\mathbf{w})}{\partial w[0,p-1]} \\ \vdots & \cdots & \vdots \\ \frac{\partial f(\mathbf{w})}{\partial w[p-1,0]} & \cdots & \frac{\partial f(\mathbf{w})}{\partial w[p-1,p-1]} \end{bmatrix}$$

19

2.2 Gradient Descent & Stochastic Gradient Descent

- **2nd-order vector gradient (Hessian matrix)**

*If parameter **w** is a vector of p elements:*

$$\mathbf{w} = \left[w[0], w[1], \ldots, w[p-1]\right]^T$$

$$\nabla^2 f(\mathbf{w}) \equiv \frac{\partial^2 f(\mathbf{w})}{\partial \mathbf{w}^2} = \begin{bmatrix} \frac{\partial^2 f(\mathbf{w})}{\partial w[0]^2} & \cdots & \frac{\partial^2 f(\mathbf{w})}{\partial w[0]w[p-1]} \\ \vdots & \cdots & \vdots \\ \frac{\partial^2 f(\mathbf{w})}{\partial w[p-1]w[0]} & \cdots & \frac{\partial^2 f(\mathbf{w})}{\partial w[p-1]w[p-1]} \end{bmatrix}$$

Example: consider a quadratic function:

$$f(\mathbf{w}) = \mathbf{w}^T \mathbf{A} \mathbf{w} = \sum_{i=0}^{p-1} \sum_{j=0}^{p-1} w[i] w[j] a_{ij}$$

$$\nabla^2 f(\mathbf{w}) \equiv \frac{\partial^2 \mathbf{w}^T \mathbf{A} \mathbf{w}}{\partial \mathbf{w}^2} = \begin{bmatrix} 2a_{0,0} & \cdots & a_{0,p-1} + a_{p-1,0} \\ \vdots & \cdots & \vdots \\ a_{p-1,0} + a_{0,p-1} & \cdots & 2a_{p-1,p-1} \end{bmatrix}$$

20

2.2 Gradient Descent & Stochastic Gradient Descent

- **2nd-order optimisation**

Newton method:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \frac{1}{\nabla^2 f(\mathbf{w}_n)} \nabla f(\mathbf{w}_n) = \mathbf{w}_n - \left[\nabla^2 f(\mathbf{w}_n)\right]^{-1} \nabla f(\mathbf{w}_n)$$

Acting as the optimal step size

21

2.2 Gradient Descent & Stochastic Gradient Descent

- **Stochastic gradient descent**

If the function to be optimised is a mean function: $E\{f(\mathbf{w}, n)\}$

Then the steepest descent rule becomes:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \mu \nabla E\{f(\mathbf{w}_n)\}$$

where

$$\nabla E\{f(\mathbf{w}_n)\} = \frac{\partial}{\partial \mathbf{w}} \int f(\mathbf{w}, n, x) p(x) dx = \int [\frac{\partial}{\partial \mathbf{w}} f(\mathbf{w}, n, x)] p(x) dx$$

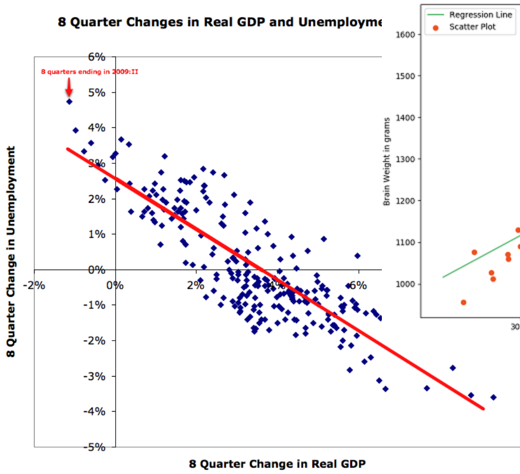
When using the <u>instantaneous gradient</u>, it becomes the *Stochastic Gradient Descent method*:

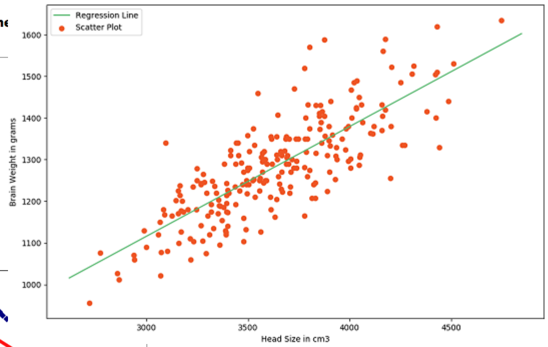
$$\mathbf{w}_{n+1} = \mathbf{w}_n - \mu \nabla \{f(\mathbf{w}_n)\}$$

22

2.3 Least Squares Method

- **Linear fitting/regression method**
  - *To fit an underlying "line" to the data*





23

2.3 Least Squares Method

- **Linear fitting/regression method**
  - *To fit an underlying "line" to the data*

For N data points, (x_i, y_i), $i=1,2,\ldots N$, to fit a function, $y=f(x,\theta)$

Sum of squares of fitting error:

$$\varepsilon = \sum (y_i - f(x_i, \theta))^2$$

$$\frac{\partial \varepsilon}{\partial \theta} = 0$$

For example, $f(x,\theta)=a+bx$

$$\frac{\partial \varepsilon}{\partial a} = -2\sum [y_i - (a + bx_i)] = 0$$

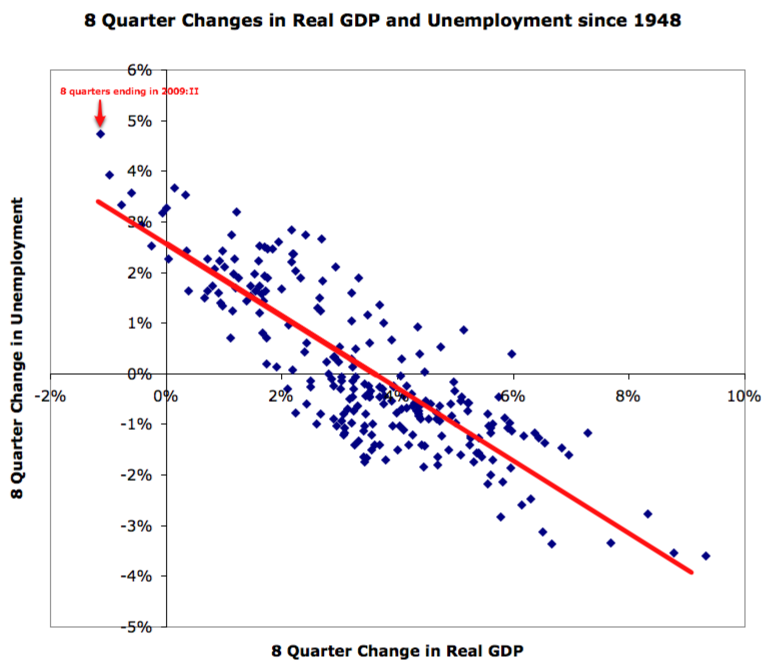
$$\frac{\partial \varepsilon}{\partial b} = -2\sum [y_i - (a + bx_i)]x_i = 0$$

$$\Rightarrow$$

$$b = \frac{N\sum(x_i y_i) - \sum x_i \sum y_i}{N\sum(x_i^2) - (\sum x_i)^2}$$

$$a = \frac{\sum y_i - b\sum x_i}{N}$$

24



25

2.4 Principal Component Analysis

- **PCA**: *A linear coordinate transformation*
  - *To find a set of "new" or "hidden" variables/directions, $\{\mathbf{q}_k\}$, which are orthogonal to each other and capture largest variances; and then project data onto them.*
  - *To (optimally) reduce data dimensionality.*

$$\max\{\mathbf{q}_i^T\mathbf{C}\mathbf{q}_i = \sigma_i^2\},\ \mathbf{q}_i \perp \mathbf{q}_j, i \neq j \qquad \min \sum_X \| \mathbf{x} - \sum_{j=1}^{m} (\mathbf{q}_j^T\mathbf{x})\mathbf{q}_j \|^2$$

- $\mathbf{x}$: n-dimensional vector, zero-mean
- $\{\mathbf{q}_j\}$: orthogonal eigenvectors of **covariance** $\mathbf{C}=E[\mathbf{x}\mathbf{x}^T]$
- $m \leq n$

Eigenvalue problem

$|\mathbf{C}-\lambda_i\mathbf{I}|=0$

$(\mathbf{C}-\lambda_i\mathbf{I})\mathbf{q}_i=0$

PCA decomposition

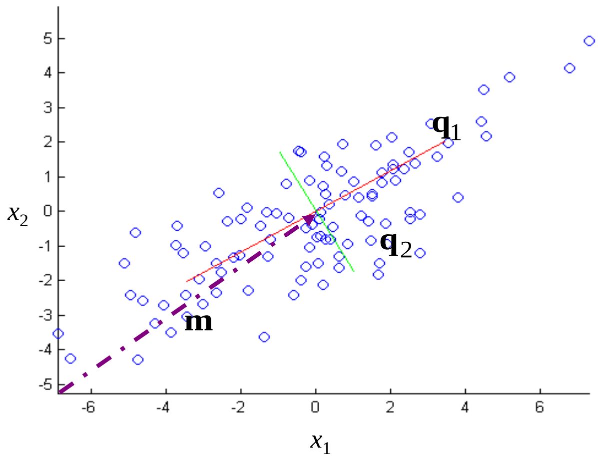
$$\mathbf{Q}^T E[\mathbf{x}\mathbf{x}^T]\mathbf{Q} = \mathbf{\Lambda}$$

- $\mathbf{Q}=[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n]$
- $\mathbf{\Lambda}=\text{diag}\,[\lambda_1, \lambda_2, \ldots \ \lambda_n]$
- $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ eigenvalues or variances

26

2.4 Principal Component Analysis

- PCA – *Optimal linear coordinate transformation & projection*



27

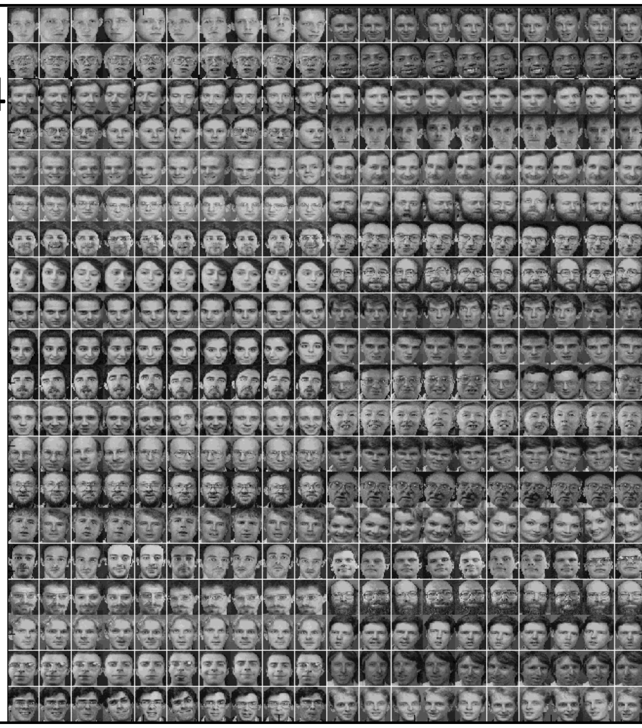
2.4 Principal Component Analysis

- PCA–example: facial images of 96x116 pixels

Examples from ORL face database (6 out of 40 subjects)

28

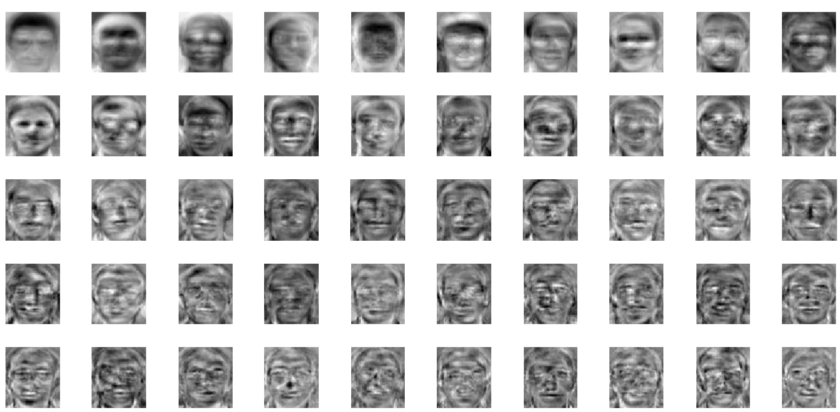
2.4 s



29

2.4 Principal Component Analysis

- PCA –example: *eigenfaces (i.e. eigenvectors)*

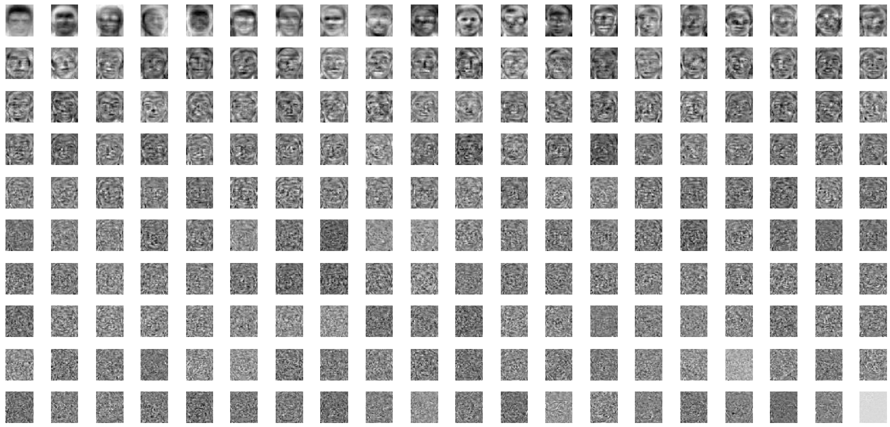


First 50 eigenfaces (from 200 training faces)

30

2.4 Principal Component Analysis

- PCA–example: *eigenfaces*



All 200 eigenfaces (of 200 training faces)

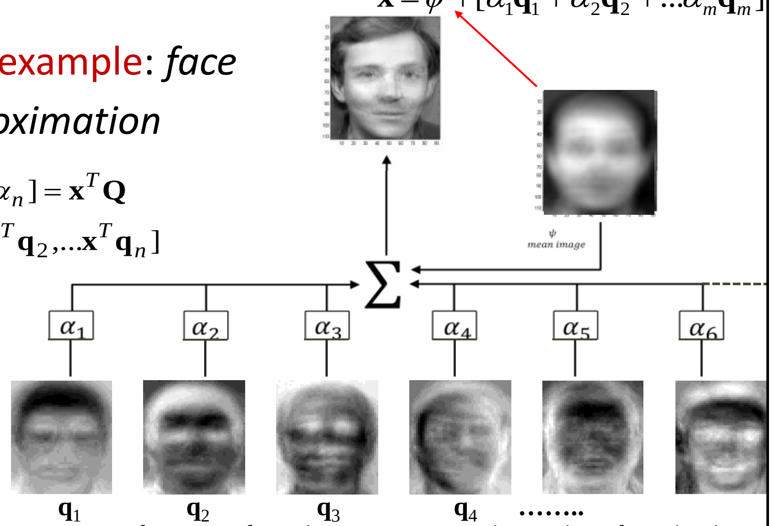
31

2.4 Principal Component Analysis

$$\mathbf{x} = \psi + [\alpha_1 \mathbf{q}_1 + \alpha_2 \mathbf{q}_2 + \ldots \alpha_m \mathbf{q}_m]$$

- PCA–example: *face approximation*

$$[\alpha_1, \alpha_2, \ldots \alpha_n] = \mathbf{x}^T \mathbf{Q}$$
$$= [\mathbf{x}^T \mathbf{q}_1, \mathbf{x}^T \mathbf{q}_2, \ldots \mathbf{x}^T \mathbf{q}_n]$$

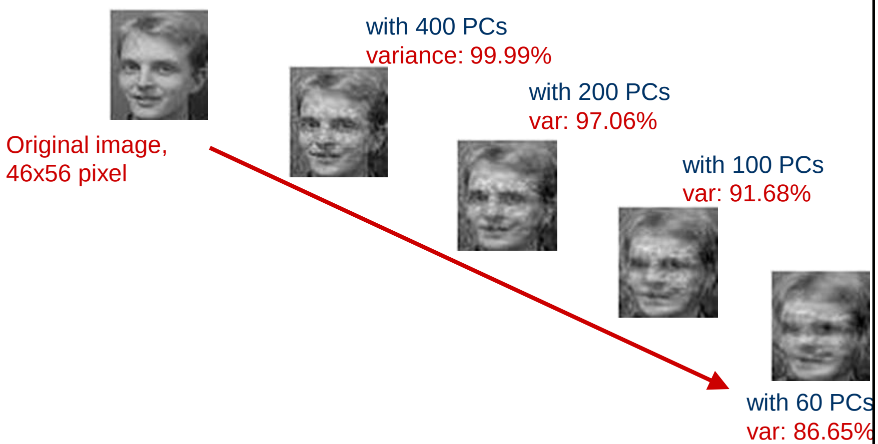


Reconstruction of an image from the mean image and a number of weighted eigenfaces, calculated from the ORL database.

32

2.4 Principal Component Analysis

- PCA–example: face approximation & compression



33

2.5 Principal Component Analysis

- Recognition with Eigenfaces

Algorithm

1. Process the image database (training set of images with labels)
   - Run PCA—compute eigenfaces
   - Calculate the *K* projection coefficients for each image
2. Given a probe image (to be recognized) **x**, calculate *K* coefficients
3. Detect if **x** is a face (Note: *better face detection methods exist*)

$$\left\| \mathbf{x} - (\bar{\mathbf{x}} + \alpha_1 \mathbf{q}_1 + \alpha_2 \mathbf{q}_2 \ldots + \alpha_K \mathbf{q}_K) \right\| < \text{threshold}$$

4. If it is a face, who is it?
   - Find the closest labelled face in training database
   - That is, the nearest-neighbour in *K*-dimensional space

34

2.4 Principal Component Analysis

(Home work)

- Recognition with Eigenfaces on ORL dataset

Detailed algorithm

0. Download ORL dataset, resize images of 96(w)x112(h) to 46x56. Split them into 5 training and 5 test images for each subject.
1. Convert each image (46x56) to column vector x of 2576x1.
2. Calculate covariance matrix C of all training images (in total 5x40=200) i.e. $C=\Sigma(x-\psi)(x-\psi)^T$, where ψ is the mean image (of training set).
   - Run PCA, e.g. using “eig” function in Matlab on C, to compute eigenfaces, i.e.
   - [V, D] =eig(C). Use the last 200 column vectors, $\{v_i\}$, in V as the eigenfaces.
3. For each training image, calculate 200 coefficients by $\alpha_i=x^T v_i$, $i=1, \ldots 200$. So, for 200 training images, each has 200 projection coefficients.
4. Given a new, probe image (to be recognized, e.g. a test image), calculate its 200 coefficients, similar to step 3, i.e. projecting it onto 200 eigenfaces.
5. Who is the probe image?
   - Compare against all training faces and find the closest training (labelled) face, in the shortest distance in terms of the 200 projection coefficients.
   - This is so-called the Nearest-Neighbour classifier.

35

Summary

- Random variables, their measures
- Concept of random processes
- Gradient descent, stochastic gradient descent
- Least squares method
- Principal component analysis, eigenface

36