If $N = 2$

$$f_{x_1, x_2}(x_1, x_2) = f_{S_1, S_2}(w_1^T \underline{x} \quad w_2^T \underline{x}) \, |W|$$

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$W = \begin{bmatrix} w_1^T \\ w_2^T \\ w_N^T \end{bmatrix} \qquad w_1^T = [w_{11}, w_{12}]$$

i.e.

$$f_{x_1, x_2}(x_1, x_2) = f_{S_1}(w_1^T \underline{x}) \; f_{S_2}(w_2^T \underline{x}) \, |W|$$

$$= f_{S_1}(s_1) \, . \, f_{S_2}(s_2) \, |W|$$

$$S_1 = w_{11} X_1 + w_{12} X_2$$
$$S_2 = w_{21} X_1 + w_{22} X_2$$
{capital S & X i.e RV's}

$$s_1 = w_{11} x_1 + w_{12} x_2$$ {value of RV}

$$\begin{bmatrix} s_1^{(1)} \\ s_2^{(1)} \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix}$$

In order to solve for W, I will try to maximize ~~[illegible]~~ $f_{x_1, x_2}(x_1, x_2)$ by choosing appropriate $\begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} = W$

$$f_{x_1, x_2}(x_1, x_2) = f_{S_1}(w_1^T \underline{x}) \; f_{S_2}(w_2^T \underline{x}) \, |W|$$

known $x_1 \; x_2 \quad x$ & assumed some W

So We have to assume some density f^n of S

papergrid

Date: / /

The log likelihood function as

$$l(W) = \sum_{i=1}^{m} \left(\sum_{j=1}^{N} \log g'(w_j^T x^{(i)}) + \log |W| \right)$$

m = no of samples.

$$X_1 = x_1^{(1)} \quad \dots \quad x_1^{(m)}$$
$$X_2 = x_2^{(1)} \quad \dots \quad x_2^{(m)}$$

In order to maximize $l(W)$ for all w's $w_1 \dots w_N$ we can use stochastic gradient descent.

No. of values of W to be determined are $N \times N$

Or we can minimize $-l(W)$

In order to use stochastic we use $l(W) = \left(\sum_{j=1}^{N} \log g'(w_j^T x^{(i)}) + \log |W| \right)$

If $N = 2$

$$W_{new} = W_{old} + \alpha \nabla_W [l(W)]$$

If we consider stochastic gradient descent one example at a time can be considered

Let us consider differentiation wrt $w_{11}, w_{12}, w_{21}, w_{22}$ where $N = 2$

$$\sum_{j=1}^{N} \log g'(w_j^T x^{(i)}) + \log |W|$$

Date: / /

$$= \log g'(\underbrace{w_1^T x^{(i)}}_{s_1^{(i)}}) + \log g'(\underbrace{w_2^T x^{(i)}}_{s_2^{(i)}}) + \log|W|$$

$$g(s) = \frac{1}{1+e^{-s}} \qquad g'(s) = g(s)(1-g(s))$$

$$= \log\left[g(w_1^T x^{(i)})\left(1 - g(w_1^T x^{(i)})\right)\right] + \log\left[g(w_2^T x^{(i)})\left(1 - g(w_2^T x^{(i)})\right)\right] + \log|W|$$

Now differentiation wrt w_{11}

$$\left(1 - 2g(s_1^{(i)})\right) x_1^{(i)}$$

papergrid

Date: / /

** The Gradient are parallel at optimum point

Unsupervised Learning.

- Group the cluster having similar property.

eg : Amazon groups people according to their likes.
Segmentation in Image processing is unsupervised.
Label of cluster is not given

04/11/19 MIDSEM PAPER.

Q2. $E(\hat{f}) = f$

X (RV), m (mean), σ^2 true parameters

M_n = estimated mean = $\frac{x_1 + x_2 \dots x_n}{n}$

n = no of samples

all x_i are iid R.V. (Assumed)

M_n taken over n examples & many times value of M_n varies but in comparable range ⟨M_n is a R.V⟩

$$E[M_n] = E\left[\frac{x_1 + x_2 \dots x_n}{n}\right]$$

$$= \frac{1}{n}\left[E[x_1] + E[x_2] \dots E[x_n]\right]$$

$$= \frac{1}{n}\left[m + \dots \dots m\right]$$

$$= \frac{n \cdot m}{n} = m$$

E (estimate-mean) = true mean = unbiased estimate

$$V_n = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 \qquad \bar{X} = \frac{1}{n}[x_1 + \dots + x_n]$$

$$Var(M_n) = Var\left(\frac{x_1 + x_2 \dots x_n}{n}\right)$$

X and Y $\rightarrow$ R.V

σ_1^2 σ_2^2

$var(X+Y) = \sigma_1^2 + \sigma_2^2$ (If they are uncorrelated)

$var(M_n) = var\left(\frac{X_1}{n}\right) + var\left(\frac{X_2}{n}\right) + \dots var\left(\frac{X_n}{n}\right)$

$= \frac{1}{n^2}[\sigma^2 \quad \dots \quad \sigma^2]$

$= \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n}$

large N (sample size), gives a good estimate, variance becomes close to 0.

$V_n = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2$ $\bar{X} = \frac{1}{n}[X_1 \dots X_n]$

V_n = R.V

Whether $E[V_n] = \sigma^2$?

$\Rightarrow$ Ans is no. $\rightarrow$ Biased estimate

~~In order to ma~~

If $V_n = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$

then $E[V_n] = \sigma^2$

$\rightarrow$ V_n is an unbiased estimate of true variance

PCA

$y = Ax$ $A^{-1} = A^T$

Parseval theorem $y^Ty = (Ax)^T(Ax)$

energy of transformed value $= x^TA^TAx$

= Energy of input value $= x^Tx$

Energy in y is over few values. $\|y\|^2 = \|x\|^2$

but in x it is distributed over all values.

04/11/19 Date: / /

Transformation matrix depends on N and not on input data. N×N.

- DCT / DFT

DCT has better energy compaction than DFT.

If N×N is size of A y has size of N

Ex: x: size 10,000 } → with transformation y: has 256 → values non zero. & rest values are 0.

y: size 10,000

$$\hat{x} = A^T y$$

Error: $\|\hat{x} - x\|^2$.

In DCT matrix A have real enteries.

Error DCT < Error DFT

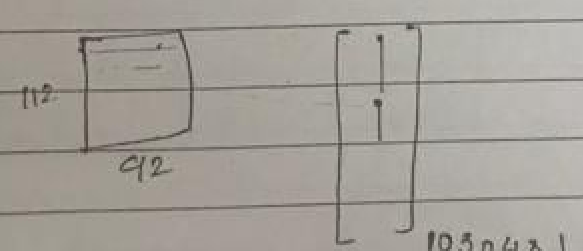
All the compressions are lossy

PCA Principal component Analysis

Transformation matrix is derived from input data

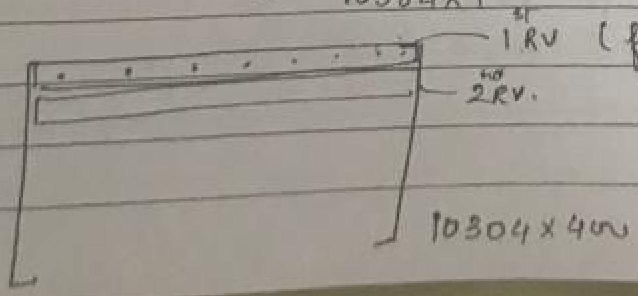
Rows of transformation matrix correspond to eigen vectors (orthogonal directions) obtained from covariance matrix of input data.

First row correspond to highest value of eigen value & second row to second highest or second Principal component



Input data matrix for 400 images.

Size 10304 × 400



So we have

10304 R.V.

cov. matrix = 10304×10304

each eigen vector size = 10304×1

no. of eigen value = 10304

Suppose we retain 100 eigen vectors.

$Y = AX$

↳ rows of A are the eigen vectors of covariance matrix of x. $= C_x$

08/11/19

$X = A^T \underline{y}$

$$X = \underbrace{\begin{bmatrix} \text{coloumns} \\ \text{of} \\ C_x \end{bmatrix}}_{10304 \times 100} \underbrace{\left[\underline{y} \right]}_{\cancel{10304} \times 400 \; 100}$$

~~100 values of~~

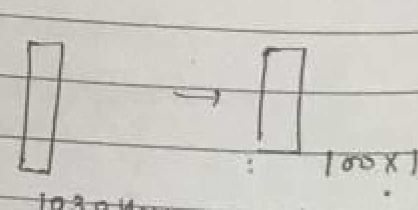
$X = 10304 \times 400$

$C_x = 10304 \times 10304$

now by retaining only 100 eigen vector from 10304 eigen vectors

As you retain more & more you get better performance (lower error)

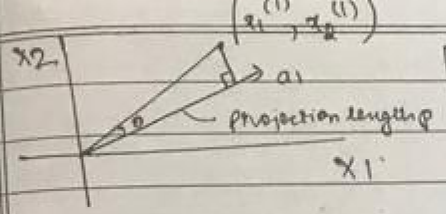
$$\underbrace{Y}_{10304 \times 400} = \underbrace{A}_{10304 \times 10304} \; X_{10304 \times 400}$$



Dimensionality reduction

↓ after reduction

$$Y_{100 \times 400} = (100 \times 10304) \cdot (10304 \times 400)$$



$$\begin{bmatrix} y_1^{(m)} & \dots & y_1^{(1)} \\ y_2^{(m)} & \dots & y_2^{(1)} \end{bmatrix} = \begin{bmatrix} A \end{bmatrix} \begin{bmatrix} x_1^{(1)} & \dots & x_1^{(m)} \\ x_2^{(1)} & \dots & x_2^{(m)} \end{bmatrix}$$

$2 \times m \qquad 2 \times 2 \qquad 2 \times m$

$$\underline{a_1} = \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix}$$

a_1 is the vector on which projected.

m: no of examples.

$x^{(1)} \dots x^{(m)}$

$x_1^{(1)}, x_2^{(1)}$ ..

$$\underline{x}^{(1)} \cdot \underline{a_1} = |\underline{x}^{(1)}| \, |\underline{a_1}| \cos\theta$$

$$\cos\theta = \frac{p}{|x^{(1)}|}$$

$$\underline{x}^{(1)} \cdot \underline{a_1} = |\underline{a_1}| \, p$$

$$p = \frac{\underline{x}^{(1)} \cdot \underline{a_1}}{|\underline{a_1}|}$$

If $a_1 = \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix}$ has unit of norm then

$$p = \underline{x}^{(1)} \cdot \underline{a_1}$$

i.e. $p = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} \end{bmatrix} \cdot \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix}$

$$p = a_{11} x_1^{(1)} + a_{12} x_2^{(1)} = y_1^{(1)}$$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

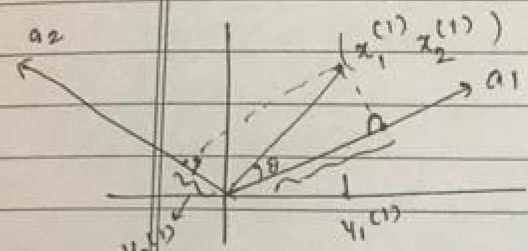
projection of $x^{(1)} \cdot a_2 = y_2^{(1)}$

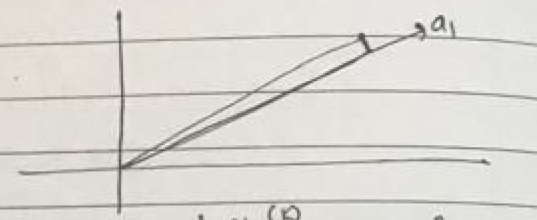
$$a_2 = \begin{bmatrix} a_{21} \\ a_{22} \end{bmatrix}$$

PCA is also called as change of basis

$$\begin{bmatrix} y_1^{(1)} \\ y_2^{(1)} \end{bmatrix} = \underbrace{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}}_{\text{Orthonormal}} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix}$$

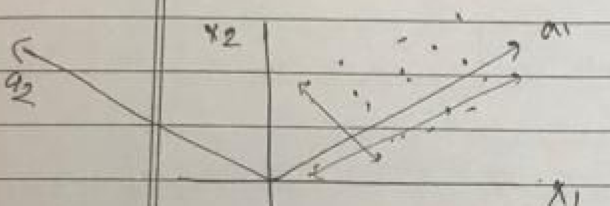
Taken the first example





∴ $y_2^{(k)}$ can be ignored but not $y_1^{(k)}$

~~i.e dr~~ i.e we are doing dimensional reduction.



a_1 is a measure of x_1 & giving more importance to 1st dimension.

Retaining $y_1^{(k)}$ is not giving much error compared to $y_2^{(k)}$

The ~~dir~~ variance is high in the drn of a_1 because the points are varying high above till a_1.

= whenever you look at variance Take the high variance, & give that dimension importance.

Variance of the first row (y) has to be maximum

$a_1 = \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix}$ has unit norm

$\underline{y} = A\,\underline{x}$

$var(a_1^T \underline{x}) = var(y_1)$

This has to be maximized.

$$\begin{array}{c} y_1 \\ y_2 \end{array} \begin{bmatrix} \longleftarrow \; y \; \longrightarrow \\ \end{bmatrix} = [A] [\;] \; \underline{x}$$

$var(a_2^T \underline{x}) = var(y_2)$

$var(a_1^T \underline{x}) = var(y_1)$

This has to be maximized under the constraint that s.t $\|a_1\|^2 = 1$

PDF.

$A =$ sq. matrix $= \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$

Real enteries ; Both rows & columns are orthonormal

; rows column vector have unit norm

$R_1 \cdot R_2 = 0 \qquad C_1 \cdot C_2 = 0$

If complex enteries $A =$ unitary matrix

$y = Ax$

$x = By \qquad B = A^{-1} = A^T$

$N=2$

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$$

$$\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = y_0 \begin{bmatrix} a_{00} \\ a_{01} \end{bmatrix} + y_1 \begin{bmatrix} a_{10} \\ a_{11} \end{bmatrix}$$

$a_{00}\, a_{10} + a_{01}\, a_{11} = 0$

$\boxed{A^T A = I \text{ if } A \text{ is orthogonal}}$

$\boxed{A^{-1} = A^{*T}}$

Unitary Matrix

$y = Ax$

$y = T[x]$ $\quad T$ is linear

Analysis : $\quad y = Ax$ $\qquad A^{-1} = A^{*T}$

Synthesis $\quad x = A^T y$

Why orthogonal transforms

Tend to redistribute energy.

Most of it in few transformed values.

Orthogonality also preserves energy

Energy $y^T y$ (final) $\qquad x^T x$ (Initial)

$$y = Ax \qquad y^T = (Ax)^T = x^T A^T$$

$$y^T y = x^T A^T A x$$

$$= x^T x$$

Correlation : Gradual variations with occasional discontinuities

Correlated data has redundancy i.e more samples than what is required to represent

Define correlation : Transformation using orthogonal matrix A results in uncorrelated values of y

Discrete Cosine Transform (DCT)

$$y(k) = C(k) \cdot \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi(2n+1)k}{2N}\right]$$

$$x(n) = \sum_{k=0}^{N-1} C(k)\, y(k) \cos\left[\frac{\pi(2n+1)k}{2N}\right]$$

$$C(k) = \sqrt{\frac{1}{N}} \quad k = 0 \qquad \sqrt{\frac{2}{N}} \quad \text{for } k \neq 0$$

papergrid

Date: / /

fixed for N

N=2.

$$\begin{bmatrix} y(0) \\ y(1) \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \end{bmatrix} \qquad y = Ax$$

$$\begin{bmatrix} x(0) \\ x(1) \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \begin{bmatrix} y(0) \\ y(1) \end{bmatrix} \qquad x = A^T y$$

DFT (unitary)

$$y(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n)\, e^{-j\frac{2\pi}{N}nk} \qquad k = 0, 1 \ldots N-1$$

$$x(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} y(k)\, e^{j\frac{2\pi}{N}nk} \qquad n = 0, 1 \ldots N-1$$

N=4

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = 1/2 \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix}$$

Advantages of DCT over DFT

- Basis are real ⇒ A is real ⇒ less computation complexity
  ⇒ Better energy compaction
- less error in DCT, by taking same no of vectors

What is PCA ?:

- Optimum transform (in MSE sense), better than DFT & DCT
- The transformation matrix is derived from input data (X)
- Rows of transformation matrix correspond to eigen vectors obtained from covariance matrix of input data

In lab on PCA

$X = (112 \times 92) \times 400$

$= 10304 \times 400$

Images = 400 pixel = 10304 = Random variable

Covariance Matrix = C_x = 10304×10304

10304 eigen values (λ_i) & eigen vector of length 10304×1

⇒ Eigen vectors are used as rows of transformation matrix A (first row with highest eigen value)

First few rows represent principal components

Eigen vectors are orthonormal

$A = 10304 \times 10304$ $X = 10304 \times 400$ $Y = 10304 \times 400$

Let us retain 100 eigen vectors

$A = 100 \times 10304$

$A^T = 10304 \times 100$ $X = 10304 \times 400$ $Y = 100 \times 400$

retain only top 100 rows of Y

Here retaining 400 rows gives minimum error (Hit & Trail)

⇒ Each image which had 10309 pixels can now be represented using only 400 values

m : 400

n : 10304

Minimum the reconstruction error, the variance will increase

/11/19

P.C.A.

$$\begin{array}{c} x_1 \\ x_2 \end{array}\begin{bmatrix} x_1^{(1)} & \dots & x_1^{(m)} \\ x_2^{(1)} & \dots & x_2^{(m)} \end{bmatrix}_{2\times m} \longrightarrow \underset{2\times 2}{A} \longrightarrow \begin{array}{c} y_1 \\ y_2 \end{array}\begin{bmatrix} y_1^{(1)} & \dots & y_1^{(m)} \\ y_2^{(1)} & \dots & y_2^{(m)} \end{bmatrix}_{2\times m}$$

variance decreases as we go down the Y matrix.

m: examples $x^{(i)}$.

n: subscript(x) no of features

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$Y = AX$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$y_1^{(1)} = a_{11} x_1^{(1)} + a_{12} x_2^{(1)} = a_1^T x^{(1)}$$

$$y_2^{(1)} = a_{21} x_1^{(1)} + a_{22} x_2^{(1)} = a_2^T x^{(1)}$$

$$a_1^T = \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix} \quad x^{(1)} = \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix} \quad a_2 = \begin{bmatrix} a_{21} \\ a_{22} \end{bmatrix}$$

$$\text{var}(y_1) = \text{var}(\underline{a_1}^T \underline{x})$$

$$\begin{bmatrix} y_1^{(1)} & \dots & y_1^{(m)} \\ y_2^{(1)} & \dots & y_2^{(m)} \end{bmatrix} = \begin{bmatrix} A \end{bmatrix}\begin{bmatrix} X \end{bmatrix}$$

$$\begin{pmatrix} y_1^{(1)} \\ y_2^{(1)} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}\begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix}$$

$$a_1 = \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix}$$

working on 1 example ~~$Y = BX$ where $B = A^T$~~

$$a_1^T = \begin{bmatrix} a_{11} & a_{12} \end{bmatrix}$$

$$Y = AX$$

$\text{var}(a_1^T \underline{x})$ can be written in terms of covariance of $\underline{x}$

Date: / /

Covariance matrix of $\underline{x}$

when $n = 2$

$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ x_2 & x_1 now vector of m values.

$$C_x = \mathrm{cov}(\underline{x}) = \begin{bmatrix} \overbrace{E(x_1 - mx_1)^2}^{\sigma_{x_1}^2} & \mathrm{cov}(x_1, x_2) \\ \mathrm{cov}(x_1, x_2) & \underbrace{E(x_2 - mx_2)^2}_{\sigma_{x_2}^2} \end{bmatrix}_{2\times 2}$$

$$mx_1 = \frac{x_1^{(1)} \; \cdots \; x_1^{(m)}}{m}$$

$x_1 - mx_1$ = from each value you subtract mx_1
Sq. each of them & divide by m = sample variance

$$\mathrm{cov}(x_1, x_2) = \mathrm{cov}(x_2, x_1) = E[\underbrace{(x_1 - mx_1)}_{1\times m}\underbrace{(x_2 - mx_2)}_{1\times m}]$$

element wise multiply

C_x = semi +ve definite
$\lambda_i \geq 0$
symmetric

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} A \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$E[y_1] = a_{11}\, mx_1 + a_{12}\, mx_2$$
$$E[y_2] = a_{21}\, mx_1 + a_{22}\, mx_2$$

$$E\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = A\, \underline{mx}$$

$$mx = \begin{bmatrix} mx_1 \\ mx_2 \end{bmatrix}$$

papergrid

Date: / /

$\underline{C_Y}$ = Cov. matrix of $\underline{Y}$

$= E[(\underset{2\times1}{\underline{Y}-\underline{m_Y}})(\underset{1\times2}{Y-\underline{m_Y}})^T]$ = 2x2 matrix.

$\underline{Y} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$ $\quad \underline{m_Y} = \begin{bmatrix} m_{Y_1} \\ m_{Y_2} \end{bmatrix}$

Y_1 & Y_2 are R.V.

$$\underline{C_X} = E\left[\underset{2\times1}{\begin{pmatrix}(x_1 - m_{x_1}) \\ (x_2 - m_{x_2})\end{pmatrix}} \underset{1\times2}{\left((x_1 - m_{x_1}) \quad (x_2 - m_{x_2})\right)}.\right]_{2\times2}$$

$$C_Y = E(YY^T)$$

$$= E[(Y - m_Y)(Y - m_Y)^T]$$

$$= E[(\underset{2\times2}{A}\underset{2\times1}{\underline{x}} - A\underline{m_x})(A\underline{x} - A\underline{m_x})^T]$$

$$= E[A(\underline{x} - \underline{m_x})(\underline{x} - \underline{m_x})^T A^T]$$

$$\underset{\text{diagonal}}{C_Y} = AC_x A^T$$

first element in $C_Y = \sum Y^{(1)\,2}$

max $\quad var(a_1^T \underline{x}) = var(Y_1)$

max

$$\begin{bmatrix} C_Y \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} C_X \end{bmatrix} \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{bmatrix}$$

$$\max (a_1^T C_x a_1) \quad \text{s.t} \quad \|\underline{a_1}^2\| = 1$$

$$L(a_1, \lambda) = a_1^T C_x a_1 - \lambda[\|a_1\|^2 - 1]$$

diff wrt a_1 & equate to 0

$$= 2C_x\underline{a_1} - 2\lambda a_1$$

$$\underline{C_x}\, \underline{a_1} = \lambda \underline{a_1}$$

$$C_y = A C_x A^T$$
$$= A(A^T \lambda A) A^T$$
$$= AA^T \lambda AA^T \qquad A^T = A^{-1}$$
$$C_y = \lambda$$

$$[\text{new data } Y]_{k \times n} = (\text{top } k)_{k \times m} (\text{orig})_{m \times n}$$

13/11/19 PCA continued.

$$Y = AX$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

↓ Output R.V. ↳ Input R.V.

x_1 takes
x_2 takes $\begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \cdots \begin{pmatrix} x_1^{(m)} \\ x_2^{(m)} \end{pmatrix}$.

↓ example

$$\begin{pmatrix} y_1^{(1)} \\ y_2^{(1)} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix}$$

If you apply each data vector we get corresponding y. (for a 2D case)

Given data set $\begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(m)} \\ x_2^{(1)} & & x_2^{(m)} \end{bmatrix}$

we arrive at $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$. It can be nD.

The problem can be solved as follows
- use constraint optimization to A.

$$a_1 = \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix} \qquad a_1^T = [a_{11} \quad a_{12}]$$

$$a_2 = \begin{bmatrix} a_{21} \\ a_{22} \end{bmatrix} \qquad a_2^T = [a_{21} \quad a_{22}]$$

Max variance of $a_1^T \underline{x}$ in order to get Y_1's variance

$var(a_1^T \underline{x}) = var(y_1)$.

$\underline{Y} = A \underline{x}$

$C_Y = A C_x A^T$

$$\begin{bmatrix} var\, y_1 & cov(y_1, y_2) \\ cov(y_1, y_2) & var\, y_2 \end{bmatrix}$$

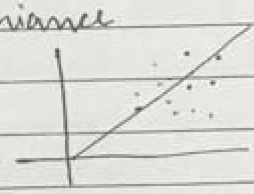
; I am interested only in diagonal elements for variance of Y_1 & Y_2.

$$A C_x A^T = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \sigma_{x_1}^2 & cov(x_1, x_2) \\ cov(x_1, x_2) & \sigma_{x_2}^2 \end{bmatrix} \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{bmatrix}$$

$var(y_1) =$ ~~a_1~~ $a_1^T C_x a_1$ has to be maximized so that Y_1 has maximum variance

s.t $\|a_1\|^2 = 1$

Above is solved using Langrangian



$$L(\underline{a_1}, \lambda) = a_1^T C_x a_1 - \lambda (\|a_1\|^2 - 1)$$

diff. wrt a_{11} & a_{12}

$$L(\underline{a_1}, \lambda) = (a_{11} \quad a_{12}) \begin{pmatrix} \sigma_{x_1}^2 & cov(x_1, x_2) \\ cov(x_1, x_2) & \sigma_{x_2}^2 \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \end{pmatrix} - \lambda (a_{11}^2 + a_{12}^2 - 1)$$

$$= (a_{11} \quad a_{12}) \begin{pmatrix} \sigma_{x_1}^2 a_{11} + cov(x_1, x_2) a_{12} \\ cov(x_1, x_2) a_{11} + \sigma_{x_2}^2 a_{12} \end{pmatrix}_{2 \times 1} - \lambda (a_{11}^2 + a_{12}^2 - 1)$$

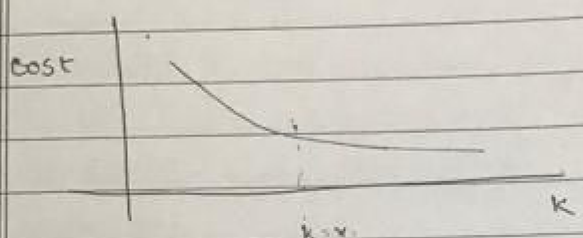
$$L(\underline{a_1}, \lambda) = \sigma_{x_1}^2 a_{11}^2 + 2 a_{11} a_{12} cov(x_1, x_2) + a_{12}^2 \sigma_{x_2}^2 - \lambda (a_{11}^2 + a_{12}^2 - 1)$$

Differentiate wrt a_{11} & equate to 0.

$$2a_{11} \sigma_{x_1}^2 + 2a_{12} cov(x_1, x_2) - 2 \lambda a_{11} = 0 \quad -(1)$$

- Application
(1) Market segmentation
(2) Social networking - Group of people using only email
Facebook
Whatsapp

Elbow Method : To get K



Algorithm:
Input K (number of clusters)

Input training set $\{ x^{(1)} \ldots \ldots x^{(m)} \}$
$x^{(i)} \in R^{d}$

Randomly initialize K cluster centroids
$\mu_1, \mu_2 \ldots \ldots \mu_K \in R^n$

Cluster assignment step → Repeat {
for $i = 1$ to m
$c^{(i)}$ = index (from 1 to K) of cluster centroid closest to $x^{(i)}$
$\min \|x^{(i)} - \mu_k\|^2 \ \forall k$

Move centroid
for $k = 1$ to K
μ_k = mean of points assigned to cluster k
}

AndrewNG.

papergrid

Date: / /

Independent component Analysis (ICA)

(Different from PCA)

: AndrewNG Notes.

PCA - dimensionality reduction

ICA is also called Blind source seperation.

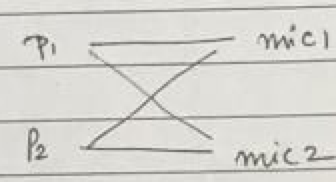
Independence is a stastical component

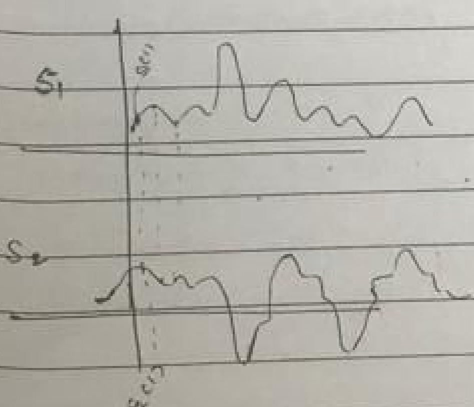
ICA works on seperation

Independence ~~vary~~ refers to the stastical component of the final (seperated) output.

$\underline{X} = A\underline{S}$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$





The sampled values of S1 take up first row of S

I will sample the 2 (@ m time instances)

$$\begin{bmatrix} s_1^{(1)} & \cdots & s_1^{(m)} \\ s_2^{(1)} & \cdots & s_2^{(m)} \end{bmatrix}_{2 \times m}$$

you will get a X of $2 \times m$

In ICA you are given x; your objective is to find S & also A

$$x = \begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(m)} \\ x_2^{(1)} & \cdots & x_2^{(m)} \end{bmatrix}$$