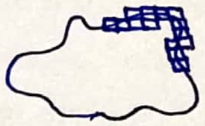
31.07.2024.

→ chain code is nothing but a shape descripter.
where the shape is obtained using the boundary information.
no information within the shape has been used.

→ Another descripter that can be used for Recognition purpose is
Polygonal Approximation.
↕
chain code is also
a type of polygonal approximation.

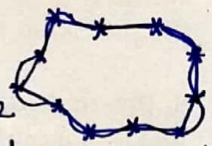
⇒ Polygonal Approximation:

→ Minimum Perimeter length polygonization.
→ splitting → criterion technique.



⇒ embed this boundary into a set of concatenated cells.
so inner and outer bothe wall canbe there.
there is tension towards outer direction, inner direction and there is some limiting point.

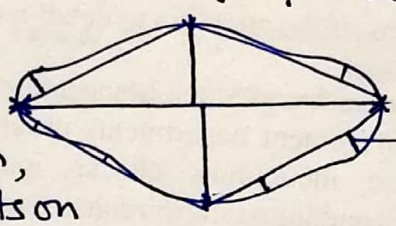
- splitting technique: consider any shape →



on this shape if we can able to point some vertices based on which the shape can be splitted then those vertices will be vertices of polygon.

criteria function helps to decide the points on the boundary which helps to create the boundary of polygon.

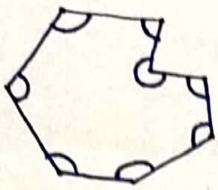
Say one shape:



→ these distances also need to be checked. if < threshold no separation of vertex.

according to a criteriafunction, initially you decide two points on boundary along which the boundary can be split. straight line split the boundary into two halves.

criteria function says that the distance from boundary to split line need to be checked. if the distance is greater than a threshold in that case the maximum distance will give one vertex of polygon



⇒ Suppose a polygon like this. - so how to generate feature vectors from this pattern.

- you can always found the angle, the inner angle of the polygon.
- if the n^{th} angle of the n^{th} vertex of the polygon is θ_n.

then this n^{th} angle can be represented by a linear combination of k nos. of previous angles.

A linear regression equation can be formed which is given by

$$\theta_n = \sum_{i=1}^{K} \alpha_i \cdot \theta_{n-i}$$

where this i will vary from 1 to K.

so any angle is represented by some previous angles to form a auto regressive model

and any angle is represented by $\alpha_1 \theta_{n-1} + \alpha_2 \theta_{n-2} + \cdots \alpha_i \theta_{n-i}$

and as the summation has been condired from 1 to K so this equation is a K^{th} order auto regressive model.

set of α is called as coefficients of auto regression.

so if any polygon has a total of p nos. of vertices then we will have p nos. of linear equation like this.

now if we want to find out p nos. of solutions then we will have exact solution if p is exactly same as n, exactly same as K.

if $p > K$ we will have less no. of equations than variables. so system will be overspecified.

if $p < K$ we will have more no. of equations than the variables. so system will be underspeified.

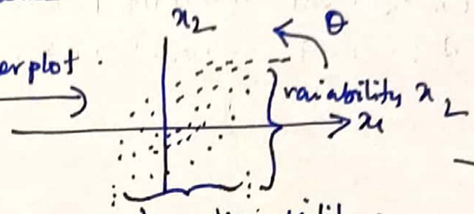
August 06, 2024

Principal Component Analysis:

- Data Reduction Technique
- Developed by Hotelling H in 1933.
- two perspective – lower dimension
  – orthogonality of new dimensions. [Principal Component].

x variable – n measurement on 2 variables.

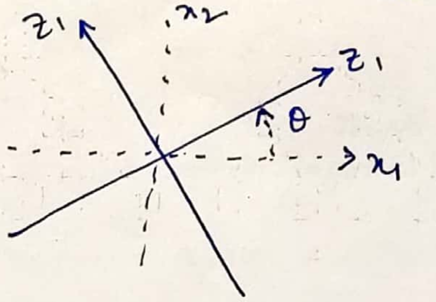
$$X_{n\times 2} = \begin{array}{c} \begin{array}{cc} x_1 & x_2 \end{array} \\ \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{bmatrix} \end{array} \xrightarrow{\text{scatterplot}}$$



after rotation by angle θ.

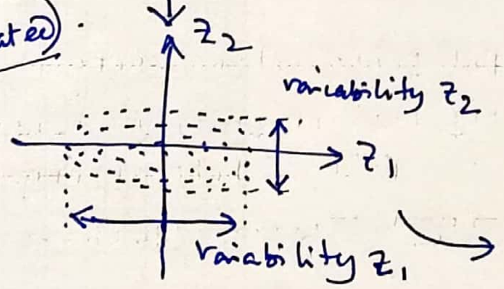
$$cov(x) = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}$$ strong correlation.

$$cor(x) = \begin{bmatrix} 1 & r_{12} \\ r_{21} & 1 \end{bmatrix}$$ → variance S_{11}, S_{22}



in this case now how you can describe about the variability along z_1 & z_2...

Transformed (rotated).



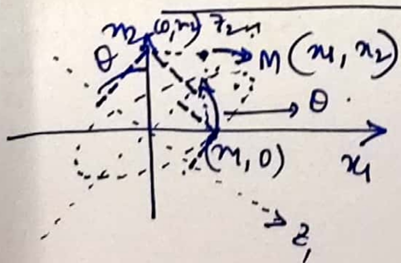
so $v(z_1) > v(z_2)$.

→ Major & minor axis of ellipse is along with z_1 & z_2 which are orthogonal.

if $v(z_1) \gg v(z_2)$ the info across z_2 is very less than z_1 so the info along with z_1 is sufficient to describe the data.

Advantage: instead of multiple co-linearity x can be transformed into z scale where one of z_1 is truely independent than the other. so linear equation can be formed.

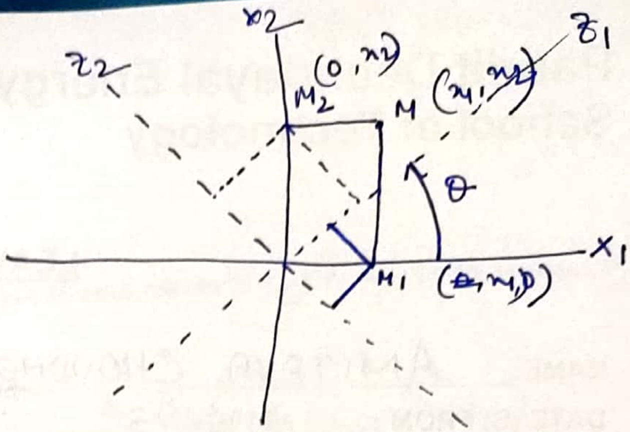
→ Now how the data reduction happens.?



what is the projection of m_1 on z_1 & z_2
similar for m_2 on z_1 & z_2
perpendicular from m_1 to z_1, z_2
, m_2 to z_1, z_2

$z_1 : x_1\cos\theta + x_2\sin\theta$
$z_2 : -x_1\sin\theta + x_2\cos\theta$

$$\Rightarrow \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



$$Z = A^T X$$

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \qquad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

so $A = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} = [a_1 \; a_2]$

$z_1 = x_1 \cos\theta + x_2 \sin\theta$

$z_2 = -x_1 \sin\theta + x_2 \cos\theta.$

then $a_1 = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$

so $a_1^T a_1 = [\cos\theta \;\; \sin\theta] \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$

$= 1.$

$a_2^T a_2 = 1.$

so $A^T A = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

$= \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\text{identity matrix}} = AA^T. = A^{-1}A$ so $\underbrace{A^{-1}.A = I}_{\text{Identity matrix}}$

so orthogonal transformation.

so from $Z = A^T X$

we can write $z_1 = a_1^T x = a_{11}x_1 + a_{12}x_2 + \dots + a_{1p}x_p$

$z_2 = a_2^T x = a_{21}x_1 + a_{22}x_2 + \dots + a_{2p}x_p$

$z_3 = a_3^T x = \dots$

$\vdots$

z_j

$\vdots$

$z_p = a_p^T x = a_{p1}x_1 + a_{p2}x_2 + \dots + a_{pp}x_p$

$a_j^T a_j = 1 \rightarrow v(z_1) \geq v(z_2) \geq \dots v(z_p)$

now what will be the principal components of a given set of data...

- Each PC is a linear combination of X, a $p \times 1$ variable vector, i.e $a_j^T X$.
- First PC is $a_1^T X$, subjected to $a_1^T a_1 = 1$ that maximizes $Var(a_1^T X)$.
- Second PC is $a_2^T X$ that maximizes $a_2^T X$ and subjected to a $a_2^T a_2 = 1$ and $cov(a_1^T X, a_2^T X) = 0$
- The jth PC is $a_j^T X$ that maximizes $Var(a_j^T X)$ and subjected to a $a_j^T a_j = 1$ and $cov(a_j^T X, a_K^T x) = 0$ for $K < j$.

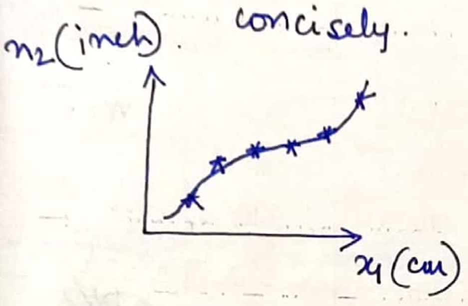
PCA's components = Eigen vectors of covariance matrix.

August 07, 2024.

⇒ PCA continues:

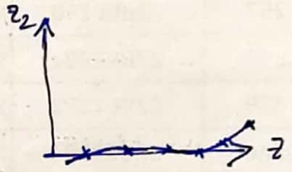
→ In pattern recognition, dimension Reduction is defined as -

- Process of converting a dataset having vast dimensions into a dataset with lesser dimensions.
- so dataset will be converted into newer dimension. PCA ensures that converted data set conveys similar information concisely.



- this graph shows two dimension x_1 & x_2
- x_1 represents measurement of several objects in cm.
- x_2 represents measurement in inches.

- through PCA these two dimensions into only 2 axis which can convert the figure according to following pattern:
- Now according to this figure the ~~figure~~ points or datas can be more easily explained.



⇒ Advantages:

- As dimensions are reduced here so storage space requirements is less.
- " " " " " " less computation time is required ~~less~~.
- Eliminates redundant features.
- Improves model performance.

⇒ Dimension Reduction Techniques:

- PCA (Principal Component Analysis)
- LDA (Linear Discriminant Analysis)

⇒ PCA:

- transforms existing variables into a new set of variables called as principal components.
- PCs are linear combination of original variables and are orthogonal.
- First PC accountable for most of the variation of original Data.

- Algorithm:
  - → Get Data
  - → Compute the mean vector (μ)
  - → Subtract mean from given data
  - → Calculate the covariance matrix
  - → Calculate the eigen vectors and eigen values of the covariance matrix
  - → Choosing components and forming a feature vector.
  - → Deriving the new data set.

August

Consider the two dimensional pattern (2, 1), (3, 5), (4, 3), (5, 6), (6, 7), (7, 8)
Compute the principal component using PCA algorithm.

Sol^n.

the given feature vectors are: $x_1 = (2, 1)$, $x_2 = (3, 5)$, $x_3 = (4, 3)$, $x_4 = (5, 6)$, $x_5 = (6, 7)$, $x_6 = (7, 8)$

$$\begin{bmatrix}2\\1\end{bmatrix}\begin{bmatrix}3\\5\end{bmatrix}\begin{bmatrix}4\\3\end{bmatrix}\begin{bmatrix}5\\6\end{bmatrix}\begin{bmatrix}6\\7\end{bmatrix}\begin{bmatrix}7\\8\end{bmatrix}$$

→ Calculate mean $\begin{bmatrix}(2+3+4+5+6+7)/6\\(1+5+3+6+7+8)/6\end{bmatrix}$

$= \begin{bmatrix}4.5\\5\end{bmatrix} \Rightarrow$ Mean feature vector.

→ Subtract the mean vector (μ) from the given feature vectors.

→ $x_1 - \mu = ((2-4.5), (1-5)) = (-2.5, -4)$

$x_2 - \mu = ((3-4.5), (5-5)) = (-1.5, 0)$

$x_3 - \mu = ((4-4.5), (3-5)) = (-0.5, -2)$

$x_4 - \mu = ((5-4.5), (6-5)) = (0.5, 1)$

$x_5 - \mu = ((6-4.5), (7-5)) = (1.5, 2)$

$x_6 - \mu = ((7-4.5), (8-5)) = (2.5, 3)$

→ so feature vectors after subtraction will be –

$$\begin{bmatrix}-2.5\\-4\end{bmatrix}\begin{bmatrix}-1.5\\0\end{bmatrix}\begin{bmatrix}-0.5\\-2\end{bmatrix}\begin{bmatrix}0.5\\1\end{bmatrix}\begin{bmatrix}1.5\\2\end{bmatrix}\begin{bmatrix}2.5\\3\end{bmatrix}$$

→ now it is required to calculate covariance matrix.

covariance matrix is given by $CM = \dfrac{\Sigma (x_i - \mu)(x_i - \mu)^t}{n}$

now $m_1 = (x_1 - \mu)(x_1 - \mu)^T = \begin{bmatrix}-2.5\\-4\end{bmatrix}\begin{bmatrix}-2.5 & -4\end{bmatrix} = \begin{bmatrix}6.25 & 10\\10 & 16\end{bmatrix}$

$m_2 = (x_2 - \mu)(x_2 - \mu)^T = \begin{bmatrix}-1.5\\0\end{bmatrix}\begin{bmatrix}-1.5 & 0\end{bmatrix} = \begin{bmatrix}2.25 & 0\\0 & 0\end{bmatrix}$

$m_3 = (x_3 - \mu)(x_3 - \mu)^T = \begin{bmatrix}-0.5\\-2\end{bmatrix}\begin{bmatrix}-0.5 & -2\end{bmatrix} = \begin{bmatrix}0.25 & 1\\1 & 4\end{bmatrix}$

$m_4 = (x_4 - \mu)(x_4 - \mu)^T = \begin{bmatrix}0.5\\1\end{bmatrix}\begin{bmatrix}0.5 & 1\end{bmatrix} = \begin{bmatrix}0.25 & 0.5\\0.5 & 1\end{bmatrix}$

$m_5 = (x_5 - \mu)(x_5 - \mu)^T = \begin{bmatrix}1.5\\2\end{bmatrix}\begin{bmatrix}1.5 & 2\end{bmatrix} = \begin{bmatrix}2.25 & 3\\3 & 4\end{bmatrix}$

$m_6 = (x_6 - \mu)(x_6 - \mu)^T = \begin{bmatrix}2.5\\3\end{bmatrix}\begin{bmatrix}2.5 & 3\end{bmatrix} = \begin{bmatrix}6.25 & 7.5\\7.5 & 9\end{bmatrix}$

→ now covariance matrix $= (m_1 + m_2 + m_3 + m_4 + m_5 + m_6)/6$

$= \frac{1}{6}\begin{bmatrix}17.5 & 22\\22 & 34\end{bmatrix} = \begin{bmatrix}2.92 & 3.67\\3.67 & 5.67\end{bmatrix}$

→ calculate the eigen values and eigen vectors:

$$\begin{vmatrix}2.92 & 3.67\\3.67 & 5.67\end{vmatrix} - \begin{vmatrix}\lambda & 0\\0 & \lambda\end{vmatrix} = 0 \Rightarrow \begin{vmatrix}2.92-\lambda & 3.67\\3.67 & 5.67-\lambda\end{vmatrix} = 0$$

From that by solving

$$(2.92-\lambda)(5.67-\lambda)-(3.67\times3.67)=0$$

$$\Rightarrow 16.56-2.92\lambda-5.67\lambda+\lambda^2-13.47=0$$

$$\Rightarrow \lambda^2-8.59\lambda+3.09=0$$

by solving $\lambda = 8.22, 0.38$

two eigen vectors $\rightarrow \lambda_1 = 8.22$ &

$\lambda_2 = 0.38$ so $\lambda_1 >> \lambda_2$

as λ_2 is very very less than λ_1 so the second eigen vector can be left out.

So the principal component will be the eigen vector corresponding to the ~~great~~largest eigen vector.

So eigen vector will be found only corresponding to λ_1.

so use the following equation to search the eigen vector.— $MX = \lambda X$

M = covariance matrix, X = eigen vector, λ = eigen value

$$\begin{bmatrix} 2.92 & 3.67 \\ 3.67 & 5.67 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 8.22\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\Rightarrow \begin{array}{l} 2.92x_1 + 3.67x_2 = 8.22x_1 \\ 3.67x_1 + 5.67x_2 = 8.22x_2 \end{array}$$

By simplifying we got —

$$5.3x_1 = 3.67x_2 \quad \cdots (1).$$
$$3.67x_1 = 2.55x_2 \quad \cdots (2)$$

solving $\Rightarrow$ $\underline{x_1 = 0.69x_2}$

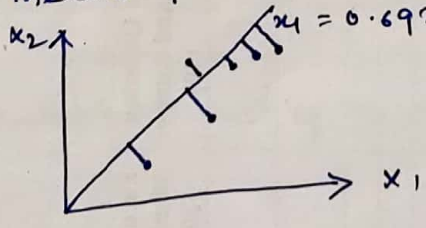
from (2) $\rightarrow$ the eigen vector is $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2.55 \\ 3.67 \end{bmatrix}$

hence the principal component of the given data set is

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2.55 \\ 3.67 \end{bmatrix}.$$

Finally project the data points onto the new subspace:

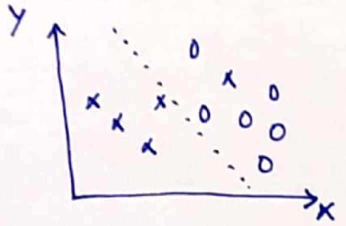
$x_1 = 0.69x_2$ (projection direction)



14.08.2024

⇒ LDA: Linear Discriminant Analysis:

- Dimensionality reduction technique
- used for supervised classification problem.
- Difference with PCA → PCA for unsupervised classification technique.
  LDA for supervised classification technique.



- two sets of data points.
- in 2D plane no straight line is possible to classify.
- no straight line can be possible to separate the two classes of the data points completely.

- In LDA, it will use both x and y axis to create a new axis in a way to maximize the separation of the two categories.

- two criteria are used by LDA
  - Maximize the distance between the means of two classes
  - Minimize the variation within each class.

- Steps:
  1. Compute the means of each class of dependent variable. (μ_1)
  2. Derive the covariance matrix of the class variable. (S_1)
  3. Compute the within class-scatter matrix $(S1 + S2) = S_W$
  4. Compute the between class scatter matrix $S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$
  5. Compute the Eigen values and vectors from the within class and between class scatter matrix. $S_W^{-1} S_B W = \lambda W$
  6. Sort the eigen values and select top K no. of values.
  7. Find the eigen vectors ~~of~~ corresponds to the top K eigen values.
  8. Obtain the LDA by taking the dot product of eigen vectors and original data.

⇒ Problem Example:

Sample for ω_1: ⇒ $X_1 = (x_1, x_2) = \{(4,2), (2,4), (2,3), (3,6), (4,4)\}$

Sample for ω_2: ⇒ $X_2 = (x_1, x_2) = \{(9,10), (6,8), (9,5), (8,7), (10,8)\}$

now according to the steps:

1. two class means are represented by μ_1 & μ_2.

so, $\mu_1 = \frac{1}{5}\left[\begin{pmatrix}4\\2\end{pmatrix} + \begin{pmatrix}2\\4\end{pmatrix} + \begin{pmatrix}2\\3\end{pmatrix} + \begin{pmatrix}3\\6\end{pmatrix} + \begin{pmatrix}4\\4\end{pmatrix}\right] = \begin{bmatrix}3\\3.8\end{bmatrix}$

$\mu_2 = \frac{1}{5}\left[\begin{pmatrix}9\\10\end{pmatrix} + \begin{pmatrix}6\\8\end{pmatrix} + \begin{pmatrix}9\\5\end{pmatrix} + \begin{pmatrix}8\\7\end{pmatrix} + \begin{pmatrix}10\\8\end{pmatrix}\right] = \begin{bmatrix}8.4\\7.6\end{bmatrix}$

T.O