Principal Component Analysis (PCA) [Dimensionality Reduction Algorithm]

① Curse of Dimensionality

- Refers to problems arises when works high dimension data
- More Features ≠ Better Model

Dimensions → Dataset : 500 features

Price of the House

House size
No. of bedrooms
No. of bathrooms

3 features	6 features	15 features	50 features
M1	M2	M3	M4
Acc 1	Acc 2 ↑↑	Acc 3 ↑↑↑	Acc 4 ↓↓

Acc 1 < Acc 2 ↑↑ < Acc 3 ↑↑↑ > Acc 4 ↓↓

← Model is overfeeded

Model got confused in calculation.

100 features → M5 — Acc 5 ↓↓↓

500 features → M6 — Acc 6 ↓↓↓↓

> Acc 5 ↓↓↓ < > Acc 6 ↓↓↓↓

② Model performance Degrade

Domain

Person ↓↓↓

⇑

Confused

Accurate

f1 2BHK

Loca ← 450K - 500K

3BHK ← 500K - 600K

Beach ← ↑↑↑

Near to a celebrity house ← ↑↑↑

Grocery shop ← ↑↑

School ←

original

f1 f2 f3 o/p

Lower ⇓ Feature Extraction

D1 D2 o/p

Two different ways to remove curse of Dimensionality

① Feature Selection ⇑

② Dimensionality Reduction (PCA) ↓↓↓↓ ⇓

Take Imp feature to train model.

↓

Feature Extraction

Feature Selection Vs Feature Extraction

↳ Dimensionality Reduction

$\boxed{3d}$ $\boxed{2d}$

① Why Dimensionality Reduction?

- Prevent ⟶ Curse of Dimensionality
- Improve the performance of the model
- Visualize the data ⟶ understand the data

$\boxed{100\,d}$

⇓

$\boxed{3d}$ or $\boxed{2d}$

Feature Selection

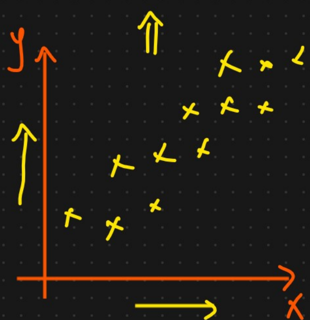
I/P	O/P
$\boxed{X}$ →	y
–	–
–	–
–	–
–	–

Linear Relationship

+ve ⇒

X↑	y↑
X↓	y↓

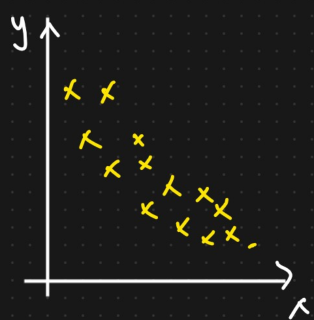
⇑



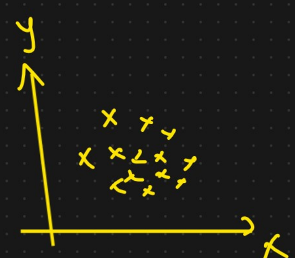
Inverse Linear Relationship

X↓	y↑
X↑	y↓

⇐ -ve



No relationship between X & Y



$$Cov(x,y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) * (y_i - \bar{y})}{N-1} = \begin{cases} +ve \\ -ve \\ \approx 0 \end{cases}$$

+ve Linear relationship, x is helpful to predict o/p.

-ve Inverse linear

≈ 0 No relationship b/w x & y.

≈0 {No Relationship}

$\boxed{-ve\ \text{correlated}}$

$$\text{Pearson Correlation} = \frac{Cov(x,y)}{\sigma_x * \sigma_y} = \boxed{-1 \text{ to } 1}$$

The more toward the value of +1 the

(more +ve correlated X & Y is

Dataset Housing

Independent feature

Not more Imp.

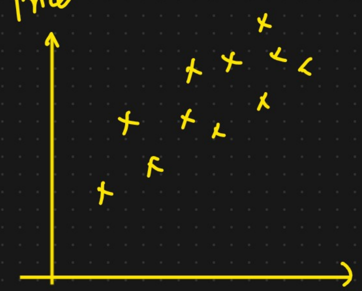
o/p feature

✓ House Size	Fountain Size X	Price
—	—	—
—	—	—

{Feature Selection}

Drop

Price

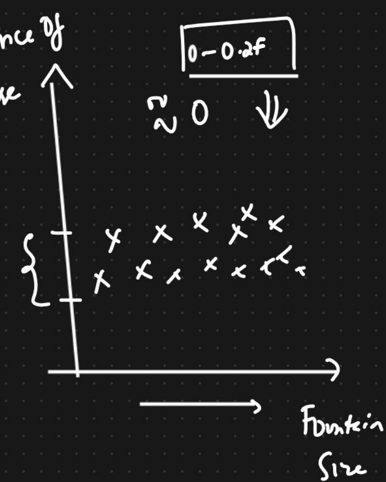


House Size

{Linear Relationship}

Covariance = +ve

-ve



Creates new features from raw data.

Domain Expertise

Feature Extraction :-

+ve

-ve

2 feature → 1 feature

Dimensionality Reduction

{ Room Size	No. of Rooms	Price	Price

Transformation To extract New feature

House Size	Price

PCA Geometric Intuition

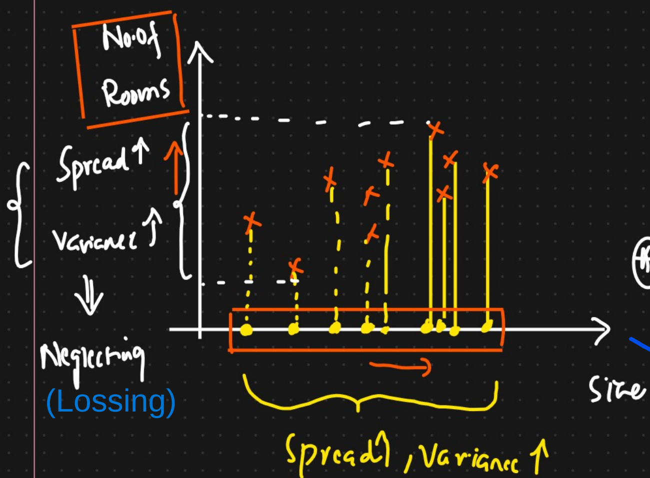
{Dimensionality Reduction}

Housing Dataset

Size House	No. of Room	Price ↑ O/P

PCA

$\boxed{2 \text{ dimension} \longrightarrow 1 \text{ dimension}}$

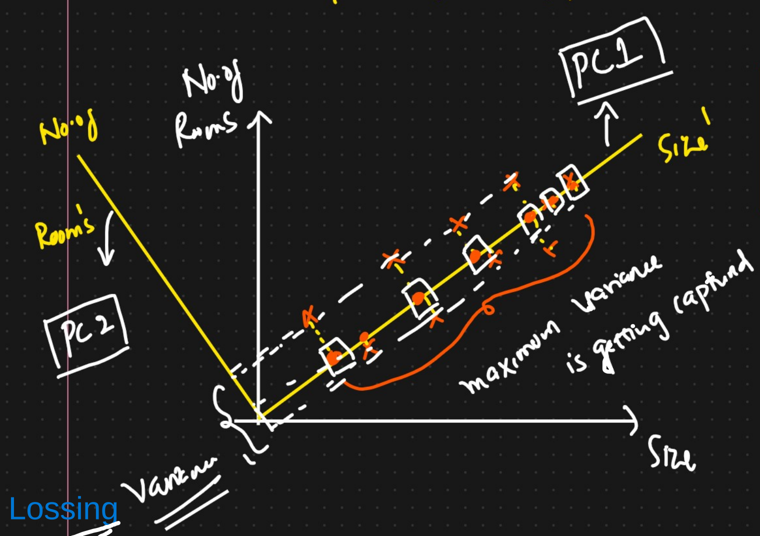


2D → 1D

(+) $\boxed{\text{Loss of information}}$ (No. of Rooms)

⇓

Feature Extraction



$\boxed{\text{Eigen decomposition on Matrix}}$

⇑

Transformation

- 2 Dimension :-

PC1 , PC2

Var(PC1) > Var(PC2)

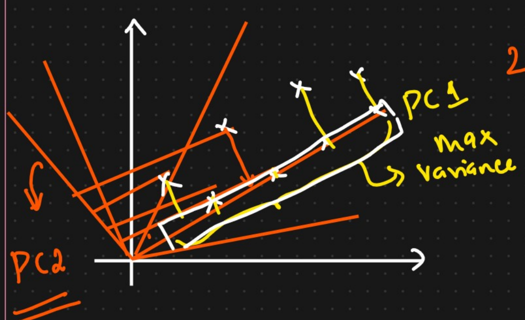
- 3 Dimension :-

PC1, PC2, PC3

Var(PC1) > Var(PC2) > Var(PC3)

2D → 1 Dimension

In PCA, much Information is not lost



2D → 1D

- Select 2 Best Principal Component

To get the best Principal Component which captures maximum variance

$3D \rightarrow \boxed{1D}$ $\qquad$ $3D \rightarrow 2D$

$\Downarrow$

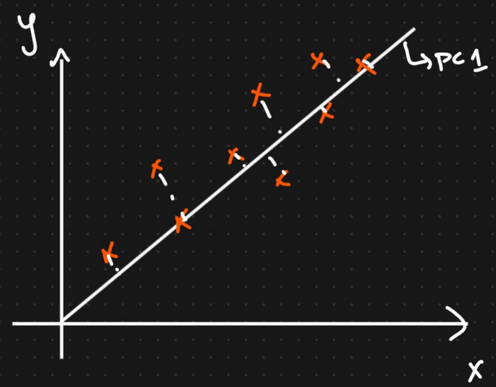
$\boxed{PC1}, \boxed{PC2}, PC3$

$PC1 \rightarrow$ 1 Dimension

$PC1, PC2 \rightarrow$ 2 Dimension.

$Var(PC1) > Var(PC2) > Var(PC3)$

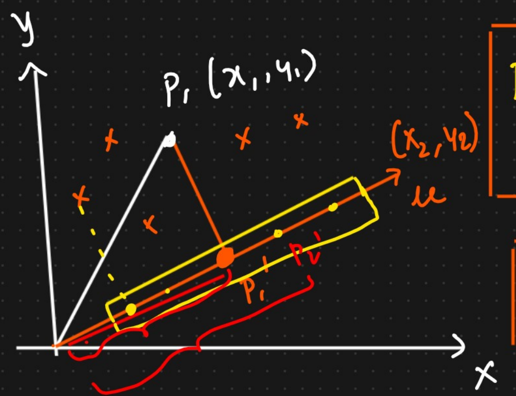
Maths Intuition behind PCA Algorithm



$2D \rightarrow 1D$

① Projection ✓

② Cost function → Variance ✓



Projection of P1 on u.

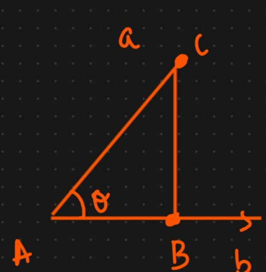
$$\text{Proj}_{P_1} u = \frac{P_1 \cdot u}{\|u\|}$$

$\|u\| = 1 \rightarrow$ unit vector

$$\boxed{\text{Proj}_{P_1} u = P_1 \cdot u}$$ ⇒ Scalar value

$\Downarrow$

P_1'



$$\boxed{P_0', P_1', P_2', P_3', P_4', \ldots P_n'}$$

$\Downarrow$

Scalar values

$\Downarrow$

Variance

$$\boxed{P_0', P_1', P_2', P_3', P_4', \ldots P_n'}$$

$$\Downarrow$$

$$x_0', x_1', x_2', x_3', x_4' \ldots x_n'$$

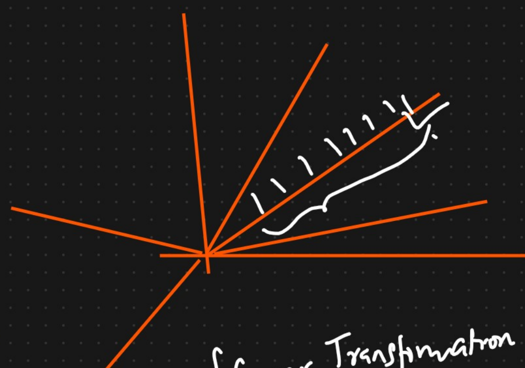
$$\text{Max Variance} = \sum_{i=1}^{n} \frac{(x_i - \bar{x})^2}{n}$$

{Goal : Find the best unit vector which captures maximum variance}.

$\Downarrow$

Cost function

$\Downarrow$ Eigen vectors And Eigen values.



① Covariance Matrix between $\boxed{\text{features}}$

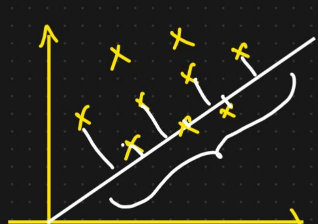
② Eigen vectors and Eigen values will found out from this covariance matrix

{Linear Transformation of matrix}

$$\boxed{Av = \lambda v}$$

③ Eigen vector $\longrightarrow$ Eigen value $\rightarrow$ magnitude of the Eigen vector $\longrightarrow$ Capture the maximum Variance

Eigen vectors And Eigen values [Linear Transformation]



[Eigen decomposition of covariance Matrix]

$\Downarrow$

Eigen vector & Eigen values

$$[\quad] * [v] = \lambda * v$$

λ ↓ Eigen Value

$$\boxed{A * v = \lambda * v}$$

⇑ ⇓

Eigen vector → Maximum magnitude
⇓
Principal Component
⇓
Max Var

Eigen vector → Max Magnitude
⇓
Max Eigen vector
⇓
Best Principal Component → PC1

★ Steps to calculate Eigen value and vectors

① Covariance of features

$\boxed{X, Y}$ Z

↓

X'

$$Cov(X,Y) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{N-1}$$

2x2

$$A = \begin{array}{c|cc} & x & y \\ \hline x & Var(x) & Cov(x,y) \\ y & Cov(y,x) & Var(y) \end{array}$$

$Cov(x,x) = Var(x)$

$Cov(y,y) = Var(y)$

	X	Y	Z
x			

y			
z			

$\boxed{\lambda_1, \lambda_2, \lambda_3}$

$\boxed{f_1 \; \& \; f_2}$

Covariance Matrix

$$\boxed{A \cdot v = \lambda \cdot v}$$

$\boxed{\lambda_1} \; \boxed{\lambda_2}$

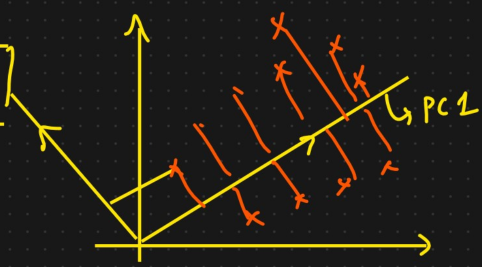
$\Downarrow \quad \Downarrow$

PC1 PC2

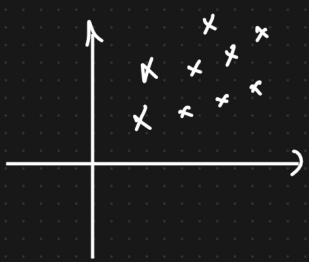
$\boxed{\lambda_1, \lambda_2} \rightarrow$ Eigen values

$\Downarrow$

$\boxed{PC1, PC2}$

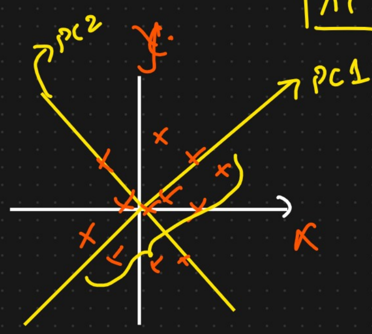


④



2D → 1D

$\boxed{\lambda_1} \Rightarrow$ magnitude of the Eigen vector



① Standardize the data

② Covariance Matrix of x & y

$$A = \begin{array}{c|cc} & x & y \\ \hline x & \mathrm{Var}(x) & \mathrm{Cov}(x,y) \\ y & \mathrm{Cov}(y,x) & \mathrm{Var}(y) \end{array} \qquad 2 \times 2$$

③ Find out Eigen vectors And value

$$A v = \lambda v$$

$\boxed{\lambda_1, \lambda_2} \Rightarrow$ Eigen values.

$\downarrow \quad \downarrow$

PC1 PC2

$3D \rightarrow 2D$

$\lambda_1, \lambda_2, \lambda_3$

$\downarrow \quad \downarrow \quad \downarrow$

PC1 PC2 PC3

$\boxed{1D}$ $\boxed{1D}$ $\Rightarrow 2D$

$3D \rightarrow 1D$

$\lambda_1, \lambda_2, \lambda_3$

$\boxed{1D}$

$2D \rightarrow 1D$

λ_1, λ_2

$\downarrow \quad \downarrow$

PC1 PC2

$\boxed{1D}$ $\rightarrow$ $\boxed{\text{Projection}}$