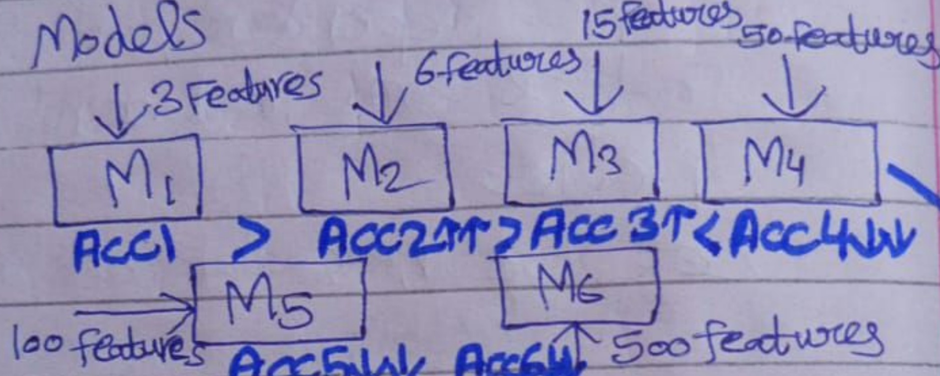
⇒ Algorithms of Unsupervised ML:

① Principal Component Analysis (PCA) [Dimension Reduction]:

① Curse of Dimensionality:

Let's suppose I have different Machine Learning Models



Let I have a dataset with 500 features (dimensions) like House size, No. of bathrooms, No of bedrooms... and I have to predict Price of house

Model is overfeeded

(we provided lot of features tha are not even important and cause confusion and model performance degrade)

This is curse of dimensionality

★ Two different ways to remove curse of dimensionality:

① Feature selection (Important features Selected)

② Dimensionality Reduction(PCA): Feature extraction (we derive feature from existing features) (we try to capture assence of these existing features).

⇒ Feature Selection vs Feature Extraction:
→ Dimensionality Reduction

Why Dimensionality Reduction?

1. To prevent curse of dimensional
2. Improve the performance of model (as more features take more time)
3. Visualize the data → understand the data (as hum understand 2D, 3D
understand

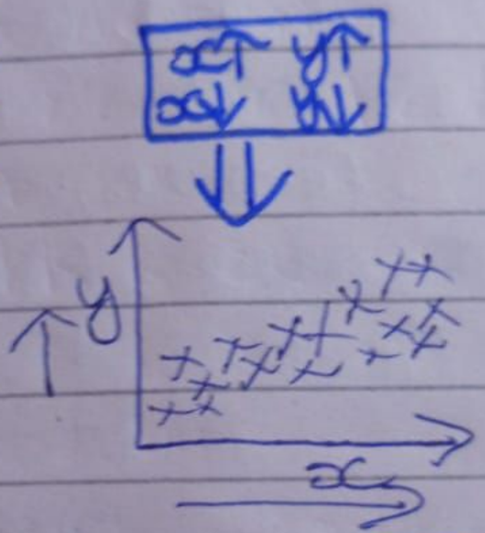
so we reduce the number of dimensions/features)

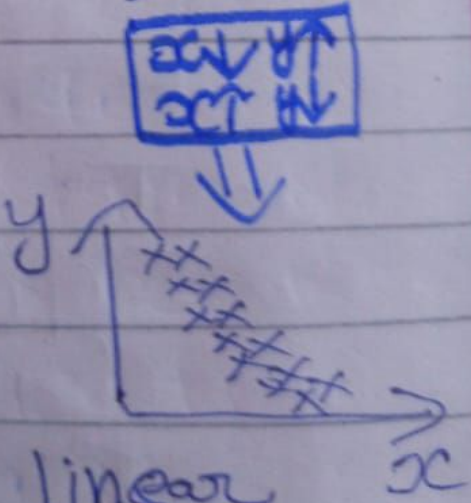
- **Feature Selection:**

Feature selection is the process which helps us to select more important features for doing prediction

Let we have two features

Input	Output
X	y
—	—
—	—
—	—
—	—





- Now we have linear relationship b/w input and output. We can quantify this relationship using covariance

$$\text{covariance}(x, y) = \sum_{i=1}^{n} \frac{(x_i - \bar{x}) * (y_i - \bar{y})}{N-1}$$

if $\text{cov}(x, y) \Rightarrow$ +ve then direct relation

if $\text{cov}(x, y) \Rightarrow$ -ve then inverse relation

if $\boxed{\text{cov}(x, y) = 0}$ means no relation, feature is not important

We can also use **Pearson correlation**

$$\text{Person Correlation} = \frac{\text{cov}(x, y)}{\sigma(x, y)}$$

It ranges from -1 to 1

- The more the value towards +1 the more positively correlated x and y is the more the value towards -1 the more negatively correlated x and y is and value towards 0 not correlated at all.

In this way we can understan the features important.

- **Feature Extraction:**

Let I have house dataset

Room Size No. of rooms Price

Let's we want to convert
2 Features → 1 Feature
And both features are important we cannot drop any so we can't use Feature selection.

- In this case we do **Feature Extraction (Dimensional Reduction)**

- In feature Extraction, we take two features perform transformation to extract a new feature like House Size from Room Size or No. of rooms.

⇛ PCA Geometric Intuition:

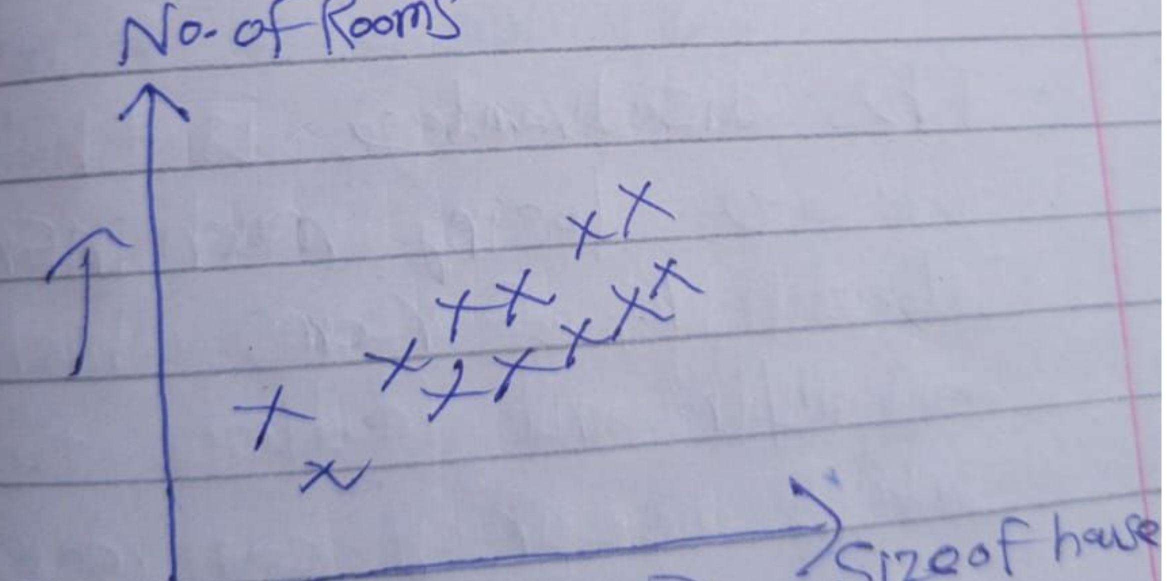
- It is used for Dimensionality Reduction (Feature Extraction)

Let we have a housing dataset

$$\overbrace{\text{Size of house} \quad \text{No. of rooms}}^{\text{Independent}} \quad \underset{\uparrow}{\text{Price}}$$

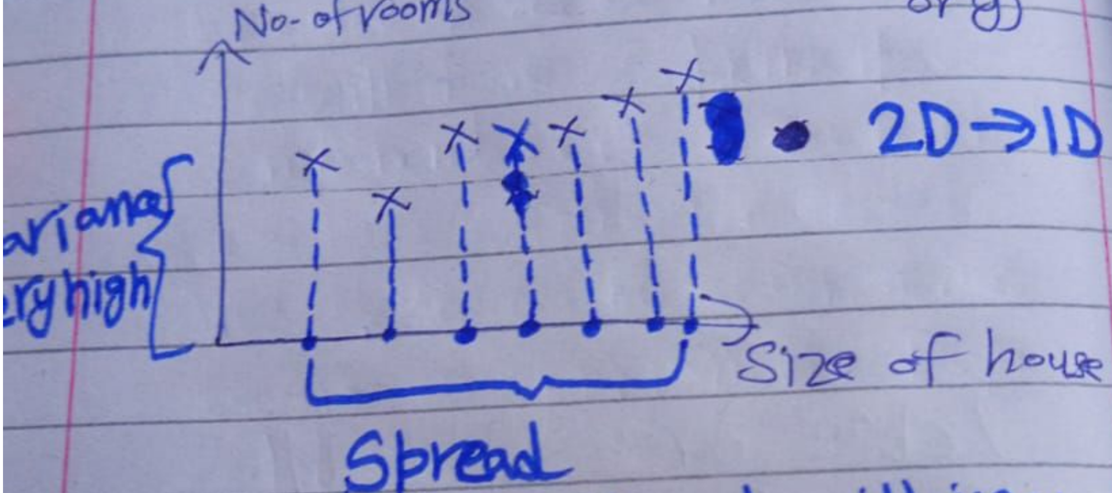
Dependent

Let's draw a plot b/w Size of house and No. of rooms



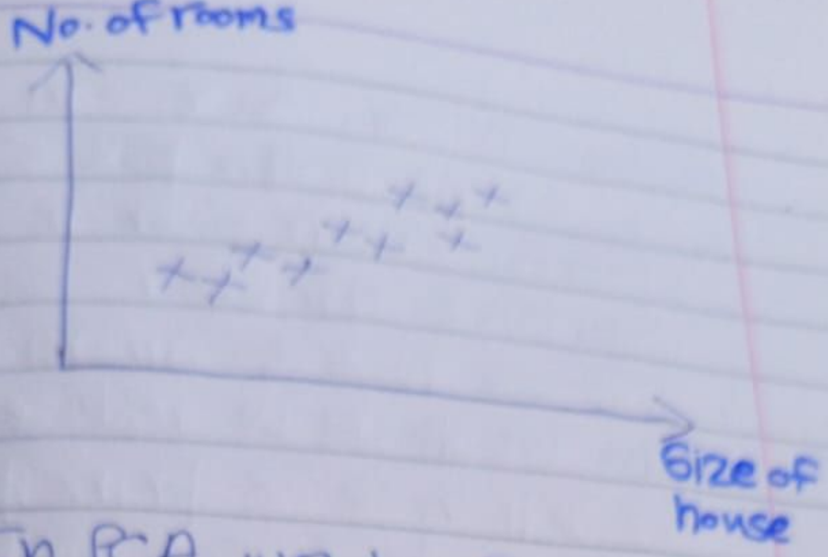
- We know as Size of house increases, No. of Rooms also increases

Now let we want to convert
2 dimension → 1 dimension
Now one simple way is project the datapoints to x-axis and take value (projecting to 1 axis x or y)

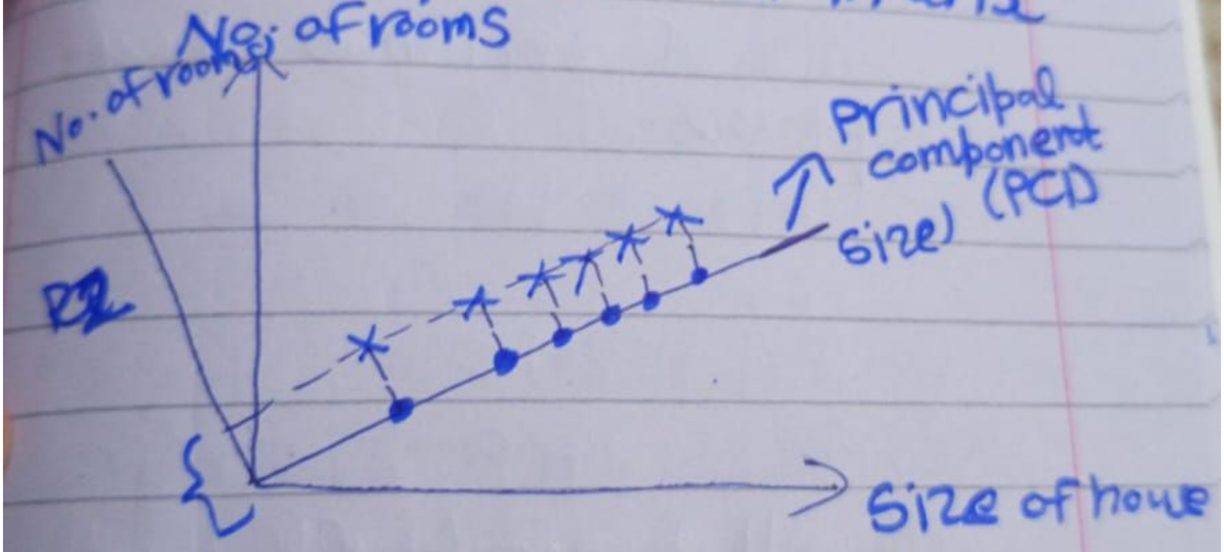


(When spread will increase variance also increase)

- The disadvantage is that we are losing much informati about No. of rooms which is an important feature so model accuracy can reduce



- In PCA we perform transformation ⇒ Eigen Decomposition on matrix



- In this case you are seeing that Maximum Variance is getting captured by introducing new axis so we are not losing much information as spread is not high
- PC1 will capture maximum information then PC2 and so on.

⇛ So in PCA our aim is to find Principal Components (axis) which cover maximum variance|spread.

- If we have 3 dimensions and we have to reduce it we will have 3 principal components (PCs) such that

Variance captured by $PC1 > PC2 > PC3$

- We select PCs on the basis of variance they capture.

$$3D \Rrightarrow 1D$$

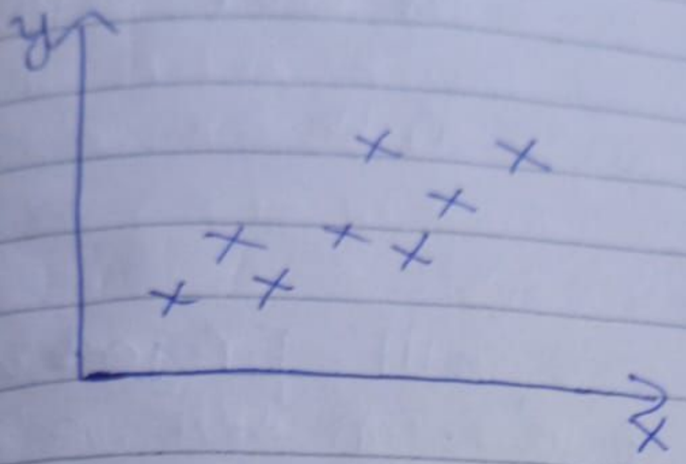
$$\Downarrow$$

$$PC1, PC2, PC3$$

$$var(PC1) > var(PC2) > var(PC3)$$

- I will project all points on PC1 and get 1 dimension
- If I want to get 2 dimension then we will project points both on PC1 and PC2.

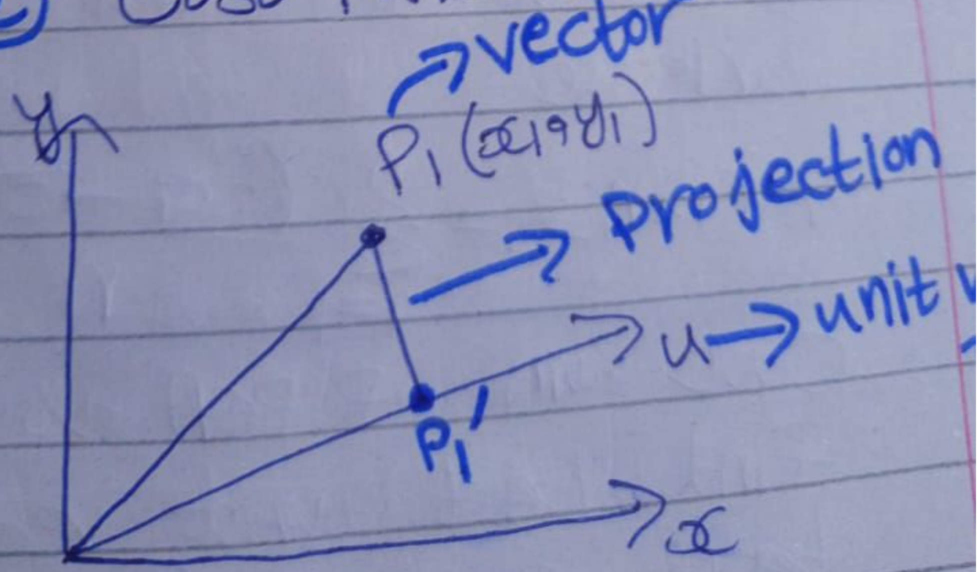
⇒ Mathematical Intuition behind PCA:



- In PCA, our aim is to find Principal component (PC1) which captures most variance.
- PCA determines PC1 is best or not it uses:

① Projection

② Cost Function ⇒ variance



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

As u ⇒ unit vector so $\|u\|=1$

$$\boxed{\text{Proj}_{P_1} u = P_1 \cdot u} \Rightarrow \text{scaler value}$$

$$\Downarrow$$

$$P_1'$$

Now we will project all the points on unit vector and get projections i.e

$$\boxed{P_0', P_1', P_2' \dots P_n'}$$

$$\Downarrow \quad \Downarrow$$

scaler values

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

Now as we get projection scaler values we can compute variance

$$\boxed{\text{Variance} = \sum_{i=1}^{n} \frac{(x_i - \bar{x})}{n}} \Rightarrow \text{Cost Function}$$

Our aim of PCA is to find best unit vector best means capture maximum variance

- Now we cannot randomly select unit vectors and check the best it is time consuming So we use:

- ⇒ **Eigen vectors and Eigen values**

Steps:

① Covariance matrix b/w features

② Eigen vectors and Eigen values will found out from this covariance matrix $\boxed{Av = \lambda v}$ → Linear transformation of matrix

③ Eigen vector has maximum eigen value (magnitude) that got selected as it captures maximum variance.

Eigen Vectors and Eigen Values[Linear Transformation] [Eigen decomposition of covariance matrix]

To find Eigen vectors and Eigen values we can use linear transformation given by:

$\boxed{AV=\lambda V}$

- And the eigen vector with maximum eigen value i.e magnitude will be our Principle Component1 (PC1) as it captures maximum variance.

⇒ Steps to calculate Eigen values and vectors:

① Covariance of features:

Let we have independent features x,y and dependent feature Z and we want to extract x' from x,y

$$\text{cov}(x,y)=\sum_{i=1}^{n}\frac{(x_i-\overline{x})(y_i-\overline{y})}{N-1}$$

$\cdot \text{cov}(x,y) = \text{cov}(y,x)$

$\text{cov}(x,x) = \text{var}(x)$

$\text{cov}(y,y) = \text{var}(y)$

As we have two independent features our matrix will be 2×2

$$A = \left.\begin{array}{c|cc} & x & y \\ \hline x & \text{var}(x) & \text{cov}(x,y) \\ y & \text{cov}(x,y) & \text{var}(y) \end{array}\right] \longrightarrow \text{Covariance matrix}$$

Now we get A as matrix we will take vector V and apply linear transformation

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

As we have two features we will get two eigen values λ_1 and λ_2

$$\begin{tikzcd} \lambda_1 \arrow[d] & \lambda_2 \arrow[d] \\ PC1 & PC2 \end{tikzcd}$$

PCA step by step with example

X	y	Z
3	1	—
4	2	—
5	5	—
6	7	—

Step1: Centre the Data

$$\bar{X} = \frac{3+4+5+6}{4} = 4.5$$

$$\bar{Y} = \frac{1+2+5+7}{4} = 3.75$$

Sutract from each value of X and Y

X	Y	Z
3-4.5	1-3.75	—
4-4.5	2-3.75	—
5-4.5	5-3.75	—
6-4.5	7-3.75	—

X	Y	Z
-1.5	-2.75	—
-0.5	-1.75	—
0.5	1.25	—
1.5	3.25	—

Step2 : Compute covariance matrix

$$A = \begin{array}{c c} & \begin{array}{cc} x & y \end{array} \\ \begin{array}{c} x \\ y \end{array} & \begin{bmatrix} var(x) & cov(x,y) \\ cov(x,y) & var(y) \end{bmatrix} \end{array}$$

$$var(x) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})^2}{N-1} = 5$$

$$var(y) = \sum_{i=1}^{n} \frac{(y_i - \bar{y})^2}{N-1} = 21.25$$

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

$$=10$$

$$A=\begin{bmatrix}5 & 10\\ 10 & 21.5\end{bmatrix}$$

Step 3: Find Eigen value

$$\det(A-\lambda I)=0$$

$$\det\left(\begin{bmatrix}5 & 10\\ 10 & 21.5\end{bmatrix}-\lambda\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right)=0$$

$$\det\left(\begin{bmatrix}5-\lambda & 10\\ 10 & 21.5-\lambda\end{bmatrix}\right)=0$$

$$(5-\lambda)(21.5-\lambda)-100=0$$

$$\lambda^2-26.5\lambda+6.25=0$$

$$a=1 \quad b=-26.25 \quad c=6.25$$

Use quadratic formula

$$\lambda=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

$$\lambda=\frac{26.25\pm\sqrt{(-26.25)^2-4(1)(6.2}}{2(1)}$$

$$\boxed{\lambda_1=26.12} \qquad \boxed{\lambda_2=0.13}$$

$\downarrow$ For PC1 $\qquad$ $\downarrow$ For PC2 $\qquad$ Eigen v

Step4:

Now find Eigen vectors which would be PC1 and PC2

For $\lambda_1 = 26.12$

$(A-\lambda I)V=0$

$$\begin{bmatrix} 5-26.12 & 10 \\ 10 & 21.25-26.12 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

$$\begin{bmatrix} -21.12 & 10 \\ 10 & -4.87 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

$$\begin{bmatrix} -21.12x+10y \\ 10x-4.87y \end{bmatrix} = 0$$

$-21.12x+10y=0$

$y=2.112x$

Let $x=1$, $y=2.112$

$$V_1 = \begin{bmatrix} 1 \\ 2.112 \end{bmatrix}$$

Similarly go for V_2 using λ_2

$$V_2 = \begin{bmatrix} 1 \\ -0.487 \end{bmatrix}$$

So

$$\boxed{\begin{array}{l} PC1 \Rightarrow x+2.112y \\ PC2 \Rightarrow x-0.487y \end{array}}$$