Principal Component Analysis - PCA

What is dimensionality reduction?

Technique which helps reduce the number of features (dimensions) while trying to keep the important information that helps understand the data.

Why dimensionality reduction?

1. The Curse of Dimensionality
    a. As the number of features increases, it can become exponentially harder for machine learning algorithms to learn effectively.
    b. Reducing dimensions helps in solving this problem.
2. Data Visualization and Interpretation
    a. Difficult to visualize high-dim data.
    b. It helps in projecting the data onto a lower-dimensional space, allowing us to visualize and interpret the data more easily.

Eigen Values and Eigen Vectors

$$x = \begin{bmatrix} -5 \\ -4 \\ 3 \end{bmatrix} \qquad A = \begin{bmatrix} 0 & 5 & -10 \\ 0 & 22 & 16 \\ 0 & -9 & -2 \end{bmatrix}$$

Multiply the matrix A with column vector A

$$Ax = \begin{bmatrix} 0 & 5 & -10 \\ 0 & 22 & 16 \\ 0 & -9 & -2 \end{bmatrix} \begin{bmatrix} -5 \\ -4 \\ 3 \end{bmatrix}$$

$$\qquad\qquad 3\times 3 \qquad 3\times 1$$

$$= \begin{bmatrix} 0 + (-20) + (-30) \\ 0 + (-88) + 48 \\ 0 + 36 + (-6) \end{bmatrix}$$

$$Ax = \begin{bmatrix} -50 \\ -40 \\ +30 \end{bmatrix} = 10 \begin{bmatrix} -5 \\ -4 \\ 3 \end{bmatrix}$$

$$Ax = 10X$$

$$\boxed{Ax = \lambda X}$$

Take 10 common from the resultant matrix,

- then A.X = 10 X

we say A.X = λX

What are eigen vector and eigen value?

For A to be a n x n non-zero vector, if A.X = λ X,

- then we say **X is the Eigenvector of A** and
- **λ is the eigen value** of the corresponding eigen vector.

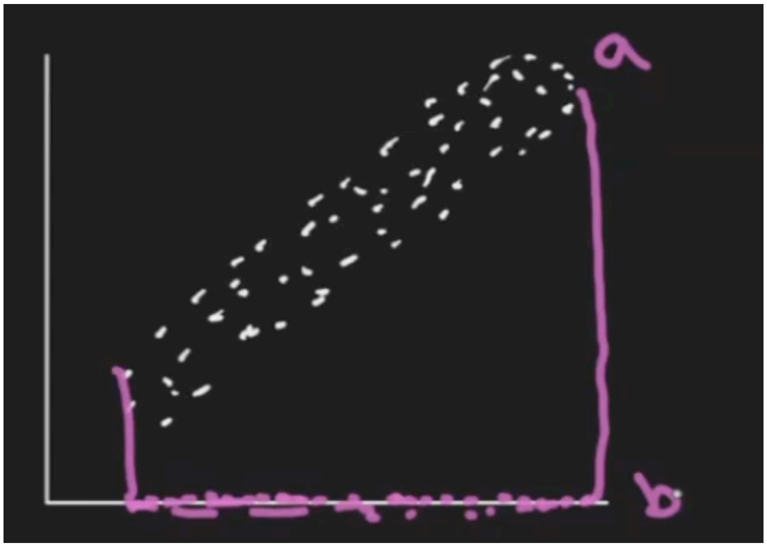
PCA: Principal component Analysis

Two methods to define PCA

- Maximum Variance formulation
- Minimum Error formulation

Maximum Variance formulation

We want to project my data points on the x-axis. Let's call the projection b.

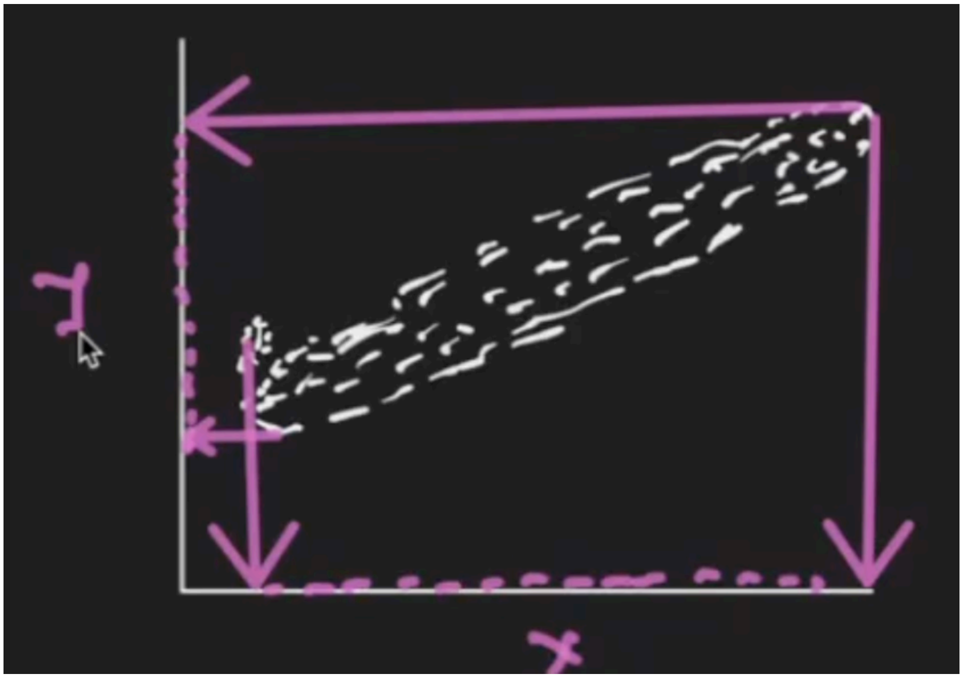


when we project them at b, data points will overlap.

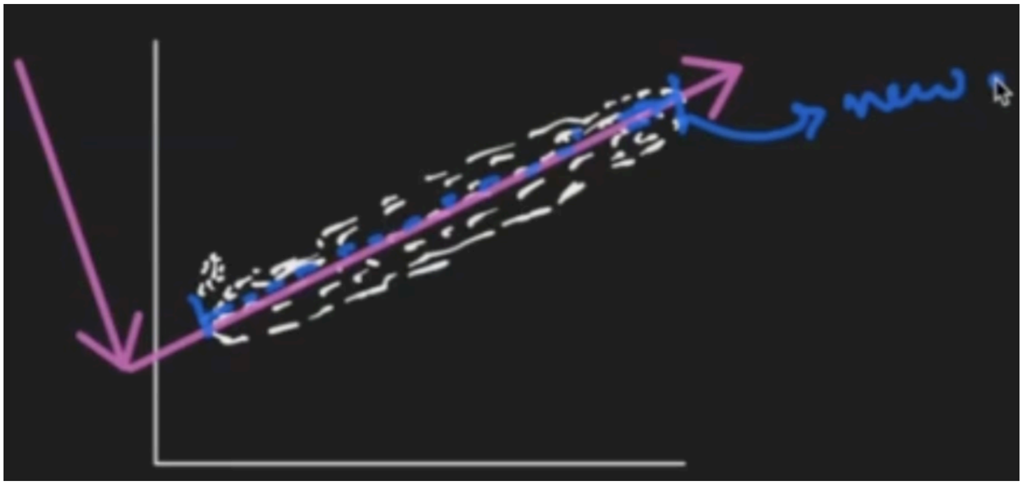
- I.e. after projection, we will see the loss of our data.

Similarly, when we project the data points to the y-axis

- the loss on the y-axis will be greater because there will be more overlapping.
- If we observe the x-axis, the data seems to be more spread(variance) than in the y-axis.



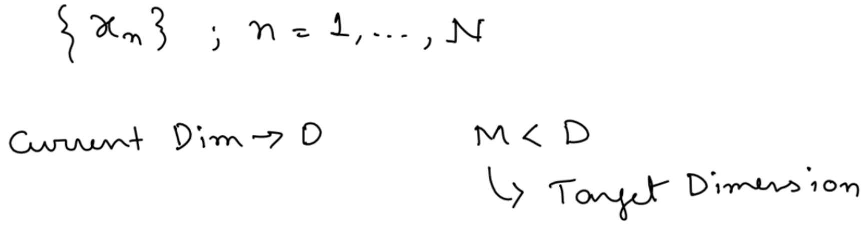
What if we rotate the axis?



By rotating the new axis we can see that the maximum variance/spread of data can be captured.

- **This becomes our direction which has the maximum variance and the goal of PCA is to find such direction.**

Maths behind Maximum Variance formulation



$\{x_n\}$; $n = 1, \dots, N$

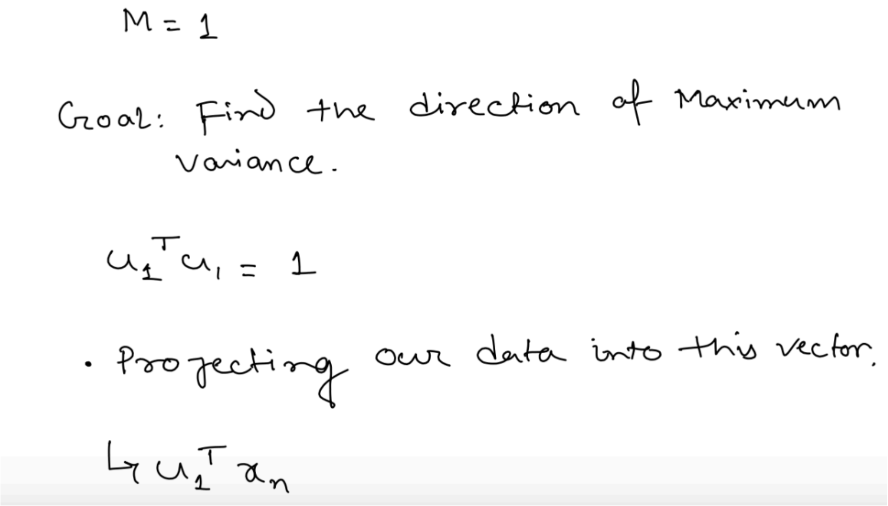
Current Dim → D

$M < D$

↳ Target Dimension

The idea is to find the direction where we can find maximum variance

- To find such a direction we are going to consider a unit vector in a random direction
- Then we are going to project our data points onto that vector



$M = 1$

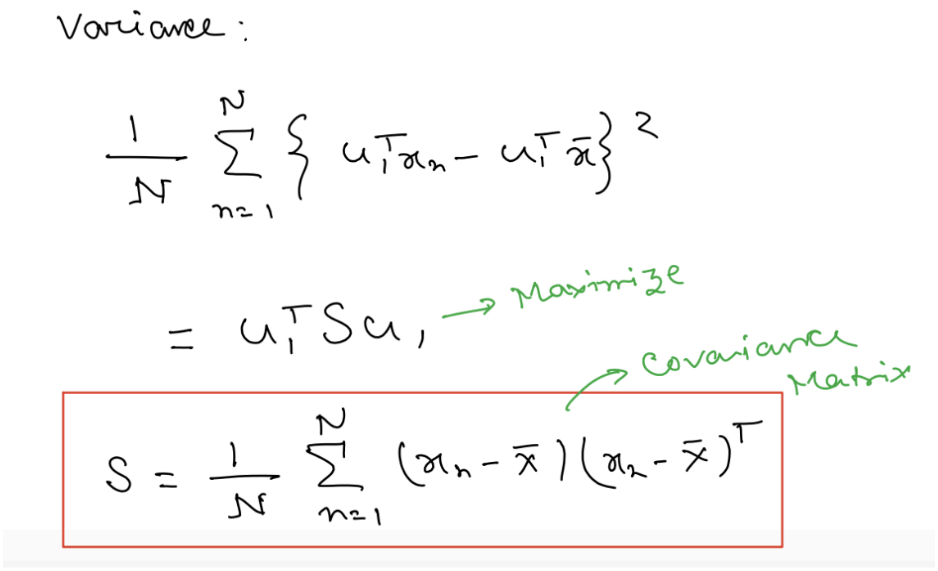
Goal: Find the direction of Maximum variance.

$u_1^T u_1 = 1$

• Projecting our data into this vector.

↳ $u_1^T x_n$

We want to find vector u s.t. It maximizes the variance.



Variance:

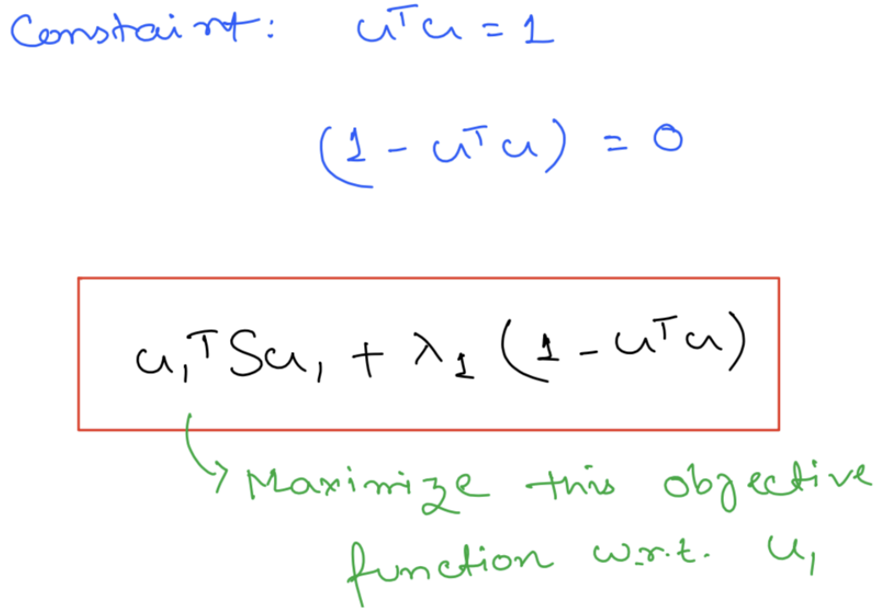
$$\frac{1}{N}\sum_{n=1}^{N}\left\{u_1^T x_n - u_1^T \bar{x}\right\}^2$$

$$= u_1^T S u_1 \rightarrow$$ Maximize

Covariance Matrix

$$\boxed{S = \frac{1}{N}\sum_{n=1}^{N}(x_n-\bar{x})(x_n-\bar{x})^T}$$

using the Lagrange multiplier, we can write objective function as:



Constraint: $u^T u = 1$

$$(1 - u^T u) = 0$$

$$\boxed{u_1^T S u_1 + \lambda_1 (1 - u^T u)}$$

Maximize this objective function w.r.t. u_1

By derivating and equating it to zero we got the eqn in A.X= λ .X format

$$2Su_1 - \lambda_1 \cdot 2u_1 = 0$$

$$\boxed{Su_1 = \lambda_1 u_1}$$

Now, we can say our direction of maximum variance can be found out by Calculating the Eigenvector of A.

Advantages

- **Easy to compute**
  - based on linear algebra, which is computationally easy to solve by computers.

- **Speeds up other machine learning algorithms**:
  - ML algorithms converge faster when trained on principal components instead of the original dataset.

- **Counteracts the issues of high-dimensional data**
  - High-dimensional data causes regression-based algorithms to overfit easily.
  - Using PCA to lower the dimensions of the training dataset -> prevent the predictive algorithms from overfitting.

Disadvantages

- **Low interpretability of principal components.**
  - Principal components are linear combinations of the features from the original data, but they are not as easy to interpret.

    - it is difficult to tell which are the most important features in the dataset after computing principal components.

Assumptions and limitations of PCA

- **Assumes a correlation between features.**
    - If the features (or dimensions or columns, in tabular data) are not correlated, PCA will be unable to determine principal components.

- **Sensitive to the scale of the features**
    - Two features - feature A takes values between 0 and 1000, while the other takes values between 0 and 1.
    - PCA will be extremely biased towards the first feature being the first principle component, regardless of the actual maximum variance within the data. This is why it's so important to standardize the values first.

- **Not robust against outliers**.
    - The algorithm will be biased in datasets with strong outliers.
    - recommended to remove outliers before performing PCA.

- **Assumes a linear relationship between features.**
    - Not well suited to capturing non-linear relationships.
    - Advised to turn non-linear features or relationships between features into linear, using standard methods such as log transforms.