Principal Component Analysis :-

MNIST → 784 dim → 2 dimensional & visualize -

d dim → d' dimensions & d' < d

→ PCA is one of the simplest dimensionality reduction technique.

* Feature Engineering → Reducing dimensionality by creating new features from given features (PCA, t SNE)
* Feature Selection → Reducing dimensionality by selecting subset of important features from given features.
(Forward feature selection, Backward elimination)

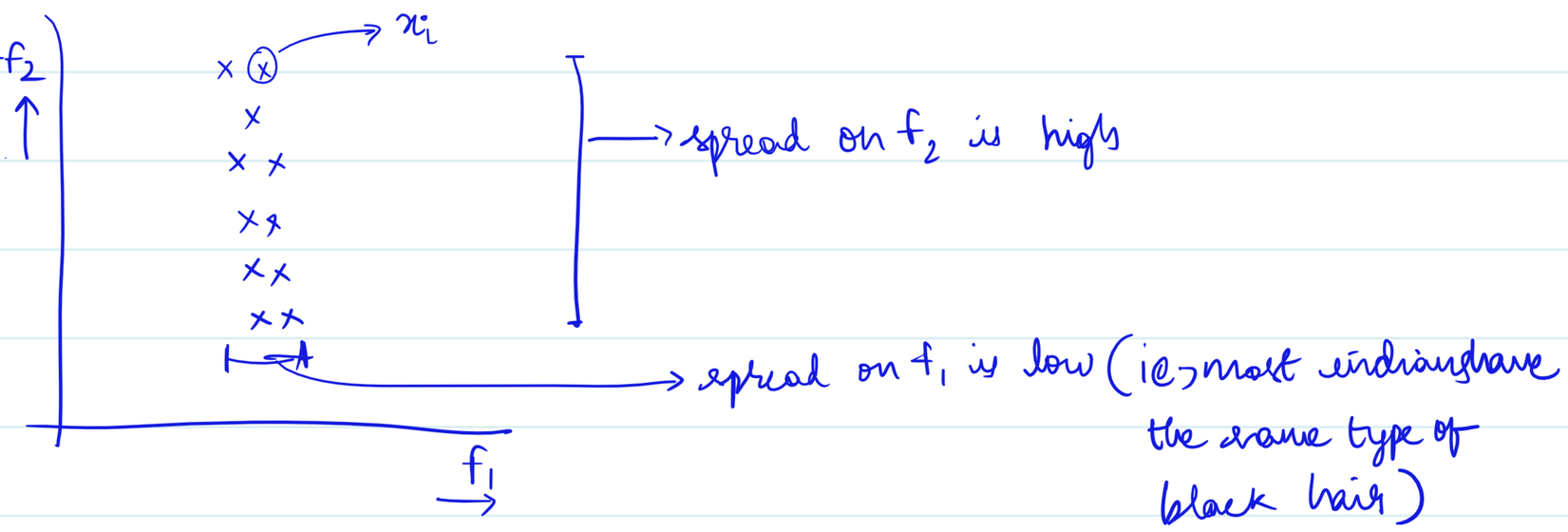
→ Categorical features :- Contain finite number of category/groups. They might not have a logical order (gender, country etc)

Continuous features :- Numerical features that have an infinite number of values b/w two values (time, price etc)

Geometric Intuition :-

Let $X = \begin{bmatrix} & f_1 & f_2 \\ 1 & & \\ 2 & & \\ 3 & & \\ \vdots & & \\ n & & \end{bmatrix}$

f_1 = blackness of hair $\in \mathbb{R}$



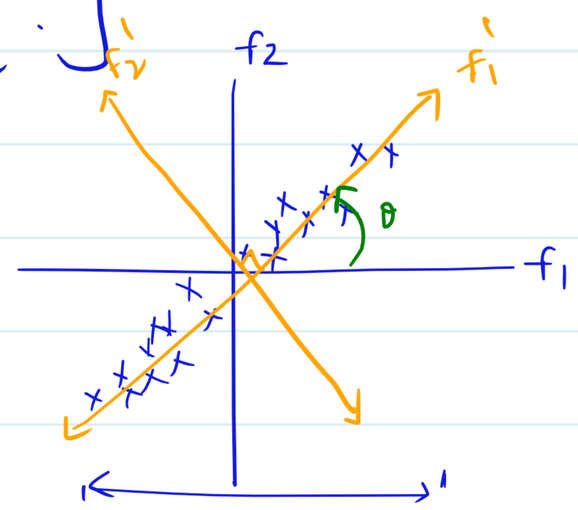
If we want to convert this 2dimensional data to 1dimension

we can skip f_2 as it has low variance & keep only f_1

$\Rightarrow$ We are preserving the direction with maximal spread/variance/information

$\Rightarrow x' = \begin{bmatrix} & f_2 \\ 1 & \\ 2 & \\ \vdots & \\ n & \end{bmatrix}$

Let $X = \begin{bmatrix} & f_1 & f_2 \\ 1 & & \\ 2 & & \\ 3 & & \\ \vdots & & \\ n & & \end{bmatrix}$ $\Rightarrow$



spread of f_2 is also high

As a result, one variable can be eliminated

(i) find $f_1' \perp f_2'$. (ii) drop f_2' (iii) Project x_i's onto f_1', then spread on f_2' << spread on f_1' 2D → 1D

→ We want to find direction of f_1' such that the variance of x_i's projected onto f_1' is max.
Achieved by rotating axis by $\theta°$ to find f_1' with max variance.

→ Before going for PCA, column standardization needs to be performed. As it brings features into a standard metric.

→ PCA is useless for dimensionality reduction if features are completely uncorrelated.

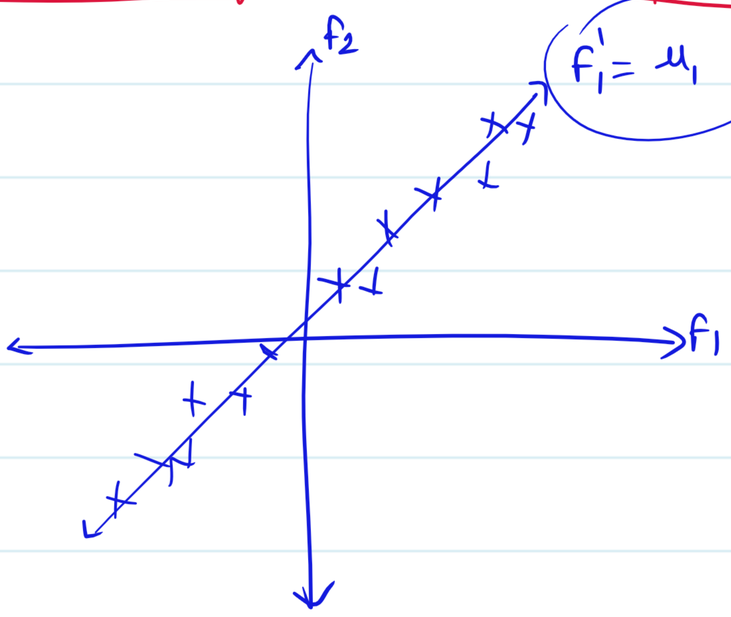
variance inflation :- When variance of dataset increases with addition of new columns that are completely correlated to already existing features of dataset

(completely correlated = repeated feature).

First principal component :- The direction in space along which projections have the largest variance.

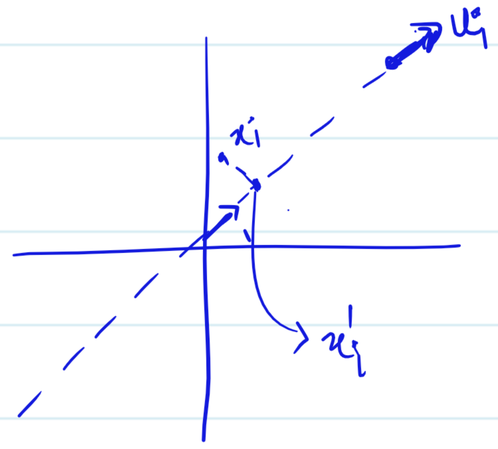
Second principal component :- Direction which maximizes variance among all directions orthogonal to the first.

Mathematical objective function of PCA :-



→ we only need the direction. u_1 is the unit vector/direction

$\Rightarrow \|u_1\| = 1$



x_1' = projection of x_1 on u_1

$D = \{x_i\}_{i=1}^n$

$D' = \{x_i'\}_{i=1}^n$

$x_1' = \text{proj}_{u_1} x_1 = \dfrac{u_1 \cdot x_1}{\|u_1\|^2} = u_1^T x_1$

$x_i' = u_1^T x_i$

$\Rightarrow \bar{x_1'} = u_1^T \bar{x}$ → mean $\{x_i\}_{i=1}^n$

→ mean $\{x_i'\}_{i=1}^n$

Objective :- Find u_1 such that var $\{\text{proj}_{u_1} x_i\}_{i=1}^n$ is maximal.

$$\text{var}\{\text{proj}_{v_1} x_i\}_{i=1}^n = \frac{1}{n}\sum_{i=1}^n \left(v_1^T x_i - v_1^T \bar{x}\right)^2$$

$v_1^T \bar{x} = \bar{x_1'}$

$v_1^T x_i$: $(1 \times n)(n \times 1)$ → 1×1 → scalar

If X is column standardized, then mean becomes 0,
⇒ this value = 0

$$\Rightarrow \text{var}\{x_i'\}_{i=1}^n = \frac{1}{n}\sum_{i=1}^n (v_1^T x_i)^2$$

We want to find v_1 such that this is maximized while keeping $v_1^T v_1 = 1 = \|v\|^2$

→ Objective of an optimization problem

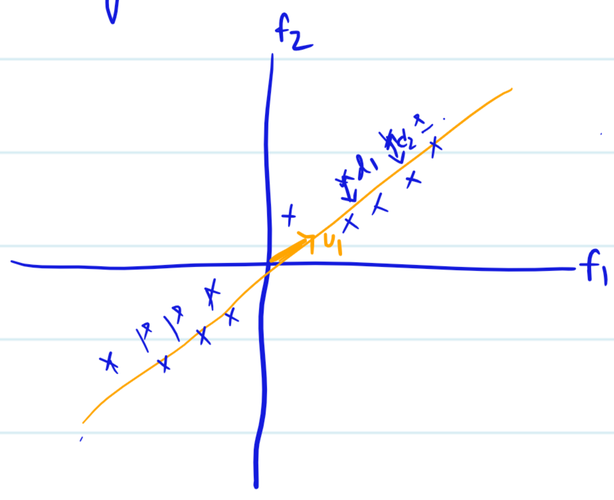
→ constraint

→ Optimization problem.

Data → Column standardization → After standardization, all features have same spread, so we don't know which one to choose → Axis transformation → Find the unit vectors in order for axis to line up with features that have the highest spread

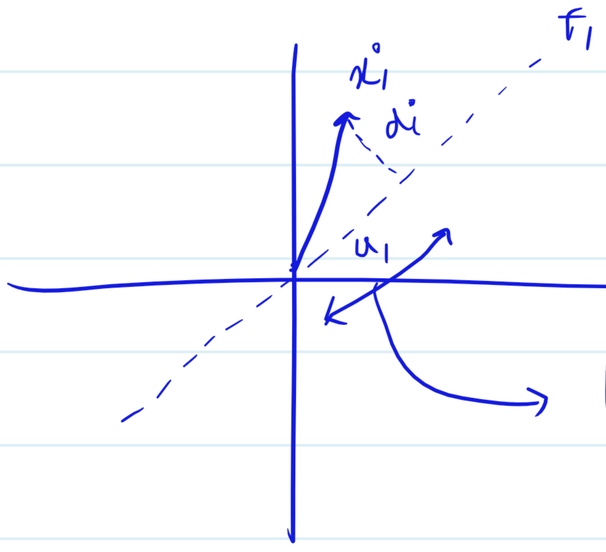
Alternative Formulazation of PCA :- Distance Minimization

previously - Find v_1 that maximizes variance -



Alternatively we want to find out v_1 that minimizes $\sum_{i=1}^{n} d_i^2$

$$\min_{v_1} \sum_{i=1}^{n} d_i^2$$



v_1 : unit vector $v_1^T v_1 = 1 = \|v_1\|^2$

projection of x_i on v_1 = $\text{proj}_{v_1} x_i = v_1^T x_i$

Pythagorean theorem: $d_i^2 = \|x_i\|^2 - (v_1^T x_i)^2$

$= x_i^T x_i - (v_1^T x_i)^2$

$$\Rightarrow \min_{v_1} \sum_{i=1}^{n} \underbrace{\left(x_i^T x_i - (v_1^T x_i)^2\right)}_{d_i^2,\ x_i \text{ is known}} \quad \text{such that } v_1^T v_1 = 1$$

→ distance minimization PCA

Previously, we had $\max_{v} \frac{1}{n} \sum_{i=1}^{n} (v_1^T x_i)^2$ such that $v_1^T v_1 = 1$ → Variance Maximization PCA.

The v_1 that achieves these two is the same

Eigen Values & Eigen Vectors in PCA :-

Solution to optimization problems :-

Let $X =$ (n × d matrix; columns standardized) $\Rightarrow S_{d \times d} = X^T X$ → Square Symmetric

can get these using numpy

{ eigen values :- $\lambda_1, \lambda_2, \lambda_3, \ldots$
eigen vectors :- $v_1, v_2, v_3, \ldots$

Eigen Values of $S = \lambda_1, \lambda_2, \ldots, \lambda_d$ → d values since S is a d×d matrix

Eigen Vectors $= v_1, v_2, v_3, \ldots, v_d$ → one eigen vectors for corresponding Eigen value.

Assume $\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4 \ldots > \lambda_d$

→ Maximal Eigen value.

Eigen value def :- $\lambda_1 v_1 = S_{d\times d} v_1$

λ_1 → Scalar → Eigen value

v_1 → vectors → Eigen vectors

If $\lambda_1 > \lambda_2 > \lambda_3 \cdots > \lambda_d$

$\downarrow v_1 \quad \downarrow v_2 \quad \downarrow v_3 \quad \downarrow v_d$

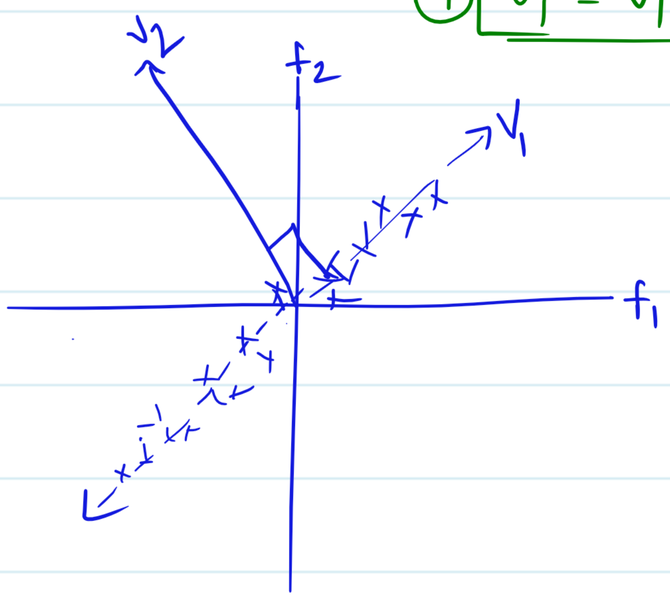
Any two v_i & $v_j \rightarrow v_i \perp v_j \Rightarrow v_i^T v_j = 0 = v_i \cdot v_j$

$U_1 = V_1$ = Eigen vectors of $S (= X^T X)$ corresponding to the largest eigen value $(= \lambda_1)$

From Optimization problem = Maximal variance Direction

Steps to find U_1 :-

1. Column Standardize X
2. Find $S_{d\times d} = X^T X$
3. Compute 'd' Eigen Values & vectors
4. $U_1 = V_1$

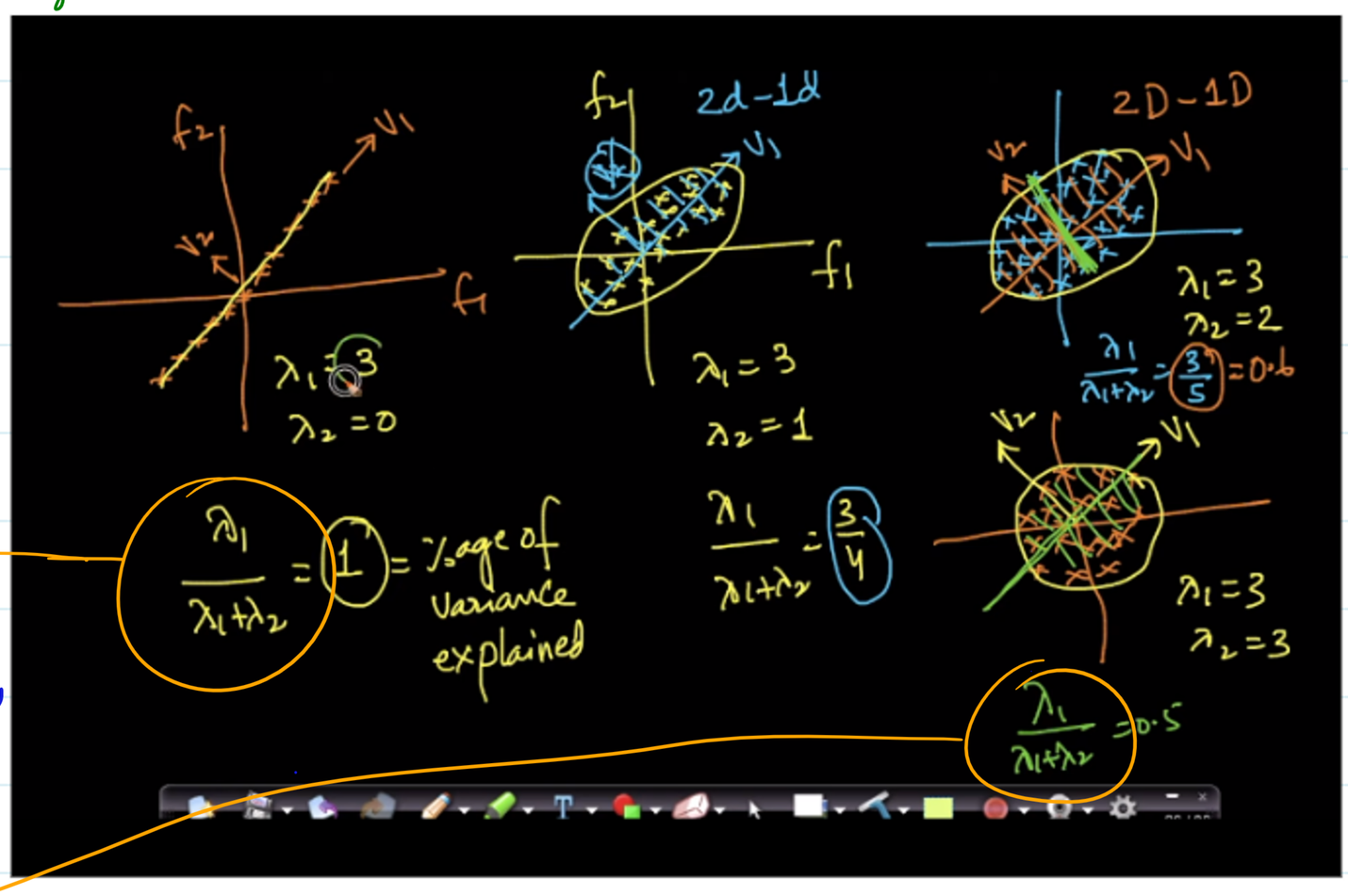


v_1 → First eigen vector with maximal variance.

v_2 → Second Eigen Vectors with maximal variance.

v_d → Least variance last eigen vector.

Significance of λ's (Eigen values) :-



How much info is present on V_1. Since the data is in a straight line, all of it on V_1, i.e. 100%.

Since data is circular, 50% of it is on V_1 & 50% is on V_2.

Eigen value :- How much data is present on each Eigen vectors.

Eigen Vectors :- Which direction the first maximal is present & the second maximal is present.

PCA for Dimensionality Reduction & Normalization :-

$$X = \begin{matrix} \\ 1 \\ 2 \\ 3 \\ \vdots \\ n \end{matrix}\begin{bmatrix} f_1 & f_2 \\ & \\ & \\ & \\ & \\ & \end{bmatrix} \xrightarrow{PCA} X' = \begin{matrix} \\ 1 \\ 2 \\ 3 \\ \vdots \\ n \end{matrix}\begin{bmatrix} V_1 \\ \\ x_i' \\ \\ \\ \\ \end{bmatrix}$$

$x_i' = x_i^T v_1$ (from max variance method PCA)

Alternative method :-

$$X' = \begin{matrix} \\ 1 \\ 2 \\ 3 \\ \vdots \\ n \end{matrix}\begin{bmatrix} V_1 & V_2 \\ & \\ & \\ & \\ & \\ & \end{bmatrix}$$

$\lambda_1 > \lambda_2 > \lambda_3 > \cdots > \lambda_d$

$x_i' = [x_i^T v_1 , x_i^T v_2]$

v_1, v_2 → Eigen vectors (top 2)

$x_i^T v_1$ → Eigen vals

→ PCA can not only be used for converting to 2D.

ex:- $X = \begin{bmatrix} & 1 & 2 & 3 & \cdots & 100 \\ 1 & & & & & \\ 2 & & & & & \\ 3 & & & & & \\ \vdots & & & & & \\ n & & & & & \end{bmatrix}$ → 100 features.

→ $\lambda_1 > \lambda_2 > \lambda_3 \cdots > \lambda_d$

$v_1 > v_2 > v_3 \cdots > v_d$

v_j

↓ PCA → 100 to 50d.

$X' = \begin{bmatrix} & 1 & 2 & 3 & \cdots & 50 \\ 1 & & & & & \\ 2 & & & & & \\ 3 & & & & & \\ 4 & & & & & \\ \vdots & & & & & \\ n & & & & & \end{bmatrix}$ → $x'_{ij} = x_i^T v_j$

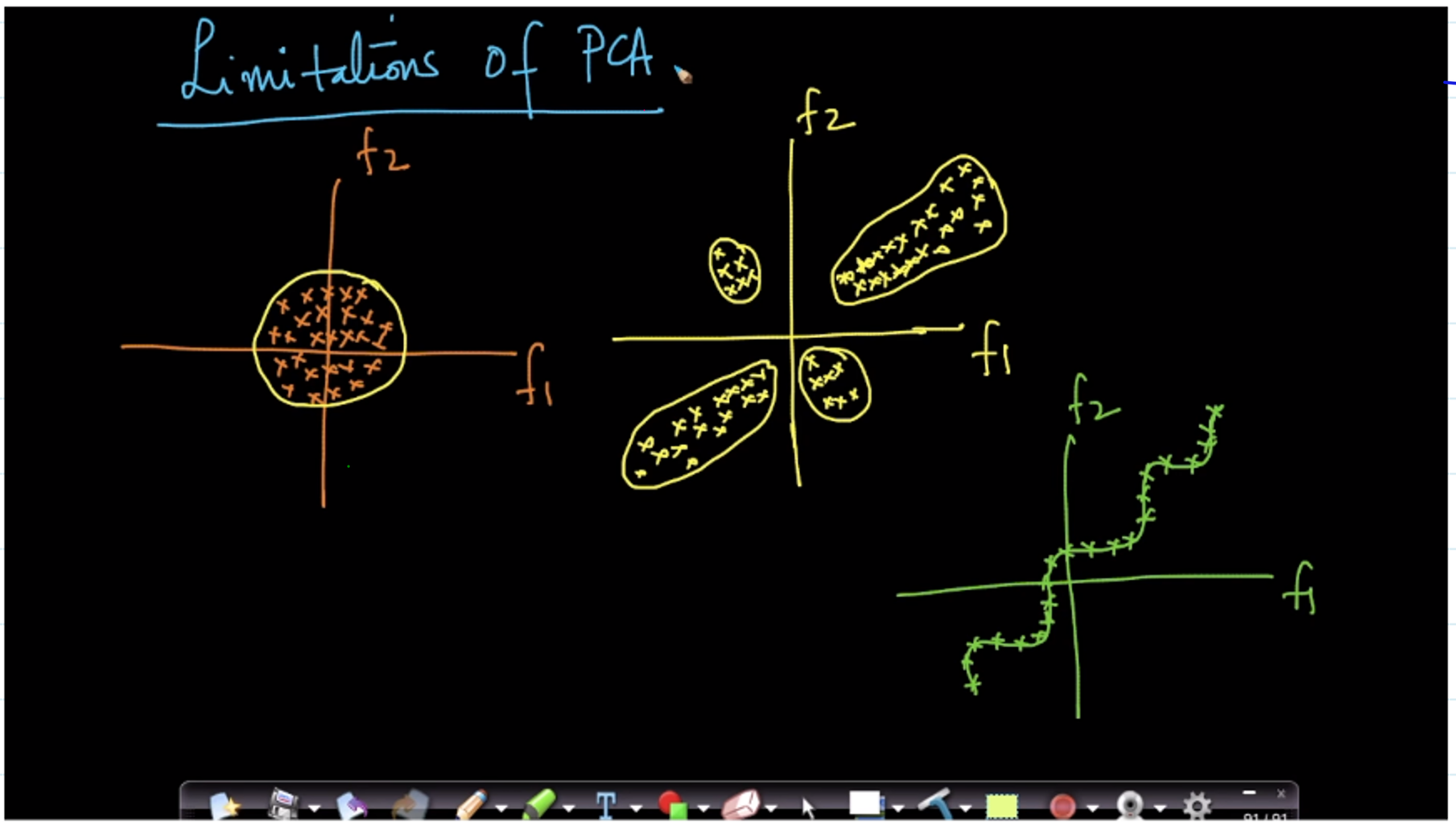
→ d feature to d' features where $d' < d$

→ Sometimes it's asked such that PCA needs to preserve 99% of the variance & the original number of features = 100.

Then we look for a value like let $\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_{51}}{\sum_{i=1}^{100} \lambda_i} = 0.99$ → $d' = 51$. It could be anything

→ sklearn inverse_transform can be used to convert d' to d dimensions.

Limitations of PCA



→ In all these cases some information is lost.

Advantages :-

① Removes correlated features by reducing dimensions

② Improves algorithm performance.

~~③ Reduces overfitting which mainly occurs when there are too many variables~~

④ Improves visualization

Disadvantages :-

① Independent variables become less interpretable. They are converted to PCs.

② Data standardization is must before performing PCA.

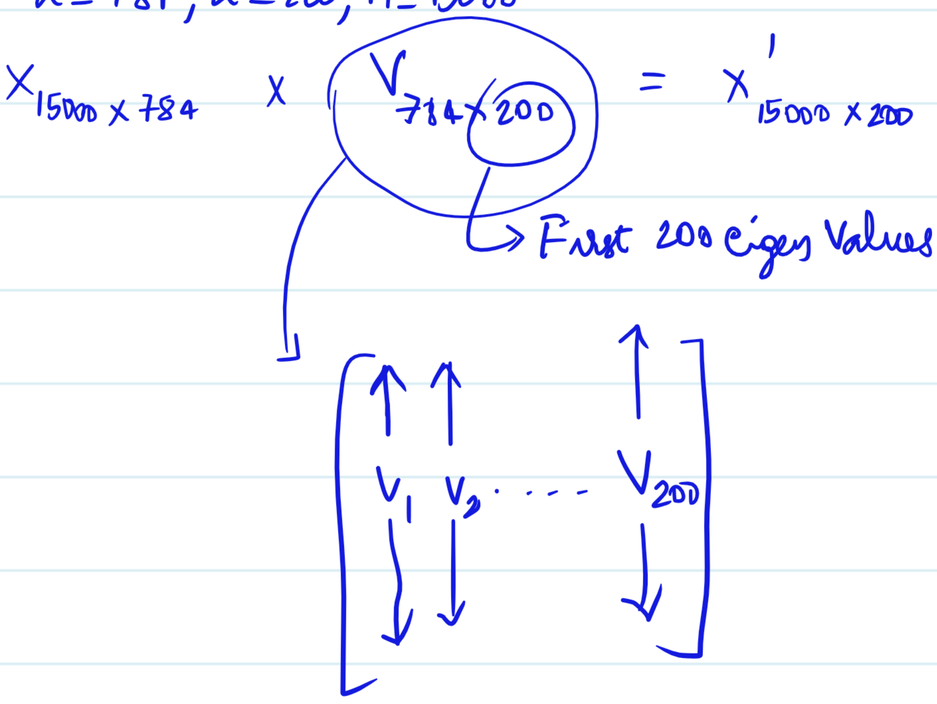
③ Information loss - If number of PCs are not selected with care.

PCA for dimensionality Reduction (Not visualization) :-

$d \xrightarrow{PCA} d'$

↳ d' when it's 2/3... it can be used for visualization sometimes, we just need to reduce the dimensions & visualization is not the main purpose.

ex:- $d = 784$, $d' = 200$, $n = 15000$



→ $\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_{d'}}{\sum_{i=1}^{d} \lambda_i}$ gives the percentage of variance explained in d' dimensions. This helps us in picking the correct value of d'.

Generally we want to find 90% retaining d'.

→ It's not always guaranteed that applying ML model after PCA shows better results than just model. So performing both & using performance metrics helps pick which one is better.